JEPA-ANYTHING: LEARNING PREDICTIVE MODELS ACROSS DIFFERENT WORLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models learn internal states for predicting how an underlying system changes. Cells, molecules, physical fields, control environments, and clinical trajectories differ in observation geometry, yet require predictions of hidden, intervened, or future states from available context. Joint-embedding predictive architectures (JEPAs) provide a natural framework for this problem, but their single target-prediction pathway does not explicitly organize predictive target coordinates. We therefore introduce JEPA-Anything, a framework that separates domain-specific observations and context–target construction from a shared predictive-state interface. Within this interface, we further propose *orthogonal predictive factorization* (OPF), which assigns target coordinates to separately parameterized factor-specific output heads and synthesizes them into a complete state for readout, intervention, planning, and multi-step rollout. Across a broad set of visual, biological, clinical, control, physical-field, weather, and molecular settings, domains retain their own encoders and observation geometries while using the same predictive core contract. Matched experiments show improvements in terminal readout, intervention prediction, and out-of-distribution or recursive forecasting; geometric analyses support well-conditioned synthesis, and factor interventions indicate functional use of the learned coordinate blocks. These results support a shared factorized predictive principle for heterogeneous worlds while preserving the structures needed for prediction, interaction, and analysis.

1 INTRODUCTION

World models turn partial observations into states that anticipate how an underlying system can change. They support latent planning and control (Ha & Schmidhuber, 2018; Hafner et al., 2025; Hansen et al., 2024), and the same capability is important for scientific systems such as cells, patients, molecules, and physical fields, where observation is incomplete and intervention can be expensive. These systems use different descriptors and encoders, making a single universal input representation unrealistic. They nevertheless share a learning problem: predict a hidden, intervened, or future state from available context. We therefore ask:

> *Can different worlds be modeled through a common predictive learning principle while retaining their own observation geometry?*

We use *world model* operationally: context is encoded into a latent state used to predict another state of the same system. The target may be a masked region, cellular response, future record or molecule, or the next environment or field configuration. The prediction may terminate at a readout or be reused for intervention, rollout, and planning. The shared object is the interface connecting context, predictive state, target, and use across domain-specific tokenizers and checkpoints.

JEPAs (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024) offer a natural foundation: a context encoder summarizes observations, a target encoder defines the requested state, and a predictor maps between them in representation space. Latent prediction abstracts away difficult or irrelevant raw detail (Balestriero & LeCun, 2024) and is effective for masked images and video (Assran et al., 2023; Bardes et al., 2024). Once domain inputs are converted into context and target states, the same operation can apply broadly.

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.

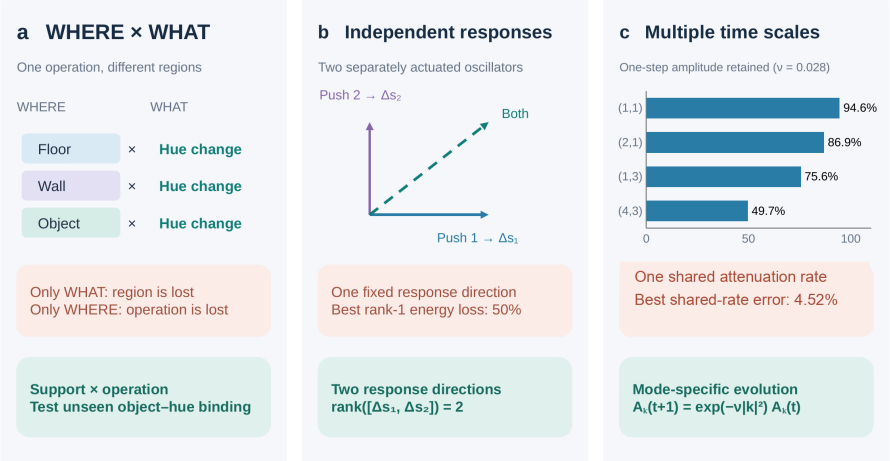

Figure 1: Motivating predictive structures. (a) Retaining only location or operation loses their binding. (b) Two equally weighted orthogonal responses incur 50% relative energy loss under their best rank-one approximation. (c) Fitting the displayed one-step amplitudes with one shared attenuation rate leaves residual error. These constructions motivate coordinate groups that keep complementary predictive information accessible while forming a complete state.

A common interface also requires a way to organize predictive capacity. Standard JEPAs (Assran et al., 2023; Bardes et al., 2024) use one target embedding and prediction pathway. This is a strong baseline, but provides no explicit allocation among local and global structure, independently changing entities, or modes evolving at different rates. Prior analysis has found that JEPA representations can favor slow features (Sobal et al., 2022); weaker or faster predictive signals may then compete within the same pathway. Figure 1 gives three motivating constructions.

The constructions cover where–what binding, composition of separately actuated responses, and modes with different rates. Together they expose a common capacity-allocation requirement: complementary predictive information should remain accessible in separate coordinate groups while preserving its joint content. The matched experiments below test prediction, conditioning, and functional use of these groups across domains.

We introduce JEPA-Anything to provide this framework. Each domain retains its raw representation, descriptors, context–target sampler, and encoder. What is shared is the algorithmic contract—context and target states enter the same analysis–prediction–synthesis core and expose the same state-reuse modes. *Anything* denotes reuse of this contract across independently parameterized domain models (Figure 2).

The shared core uses *orthogonal predictive factorization* (OPF) to analyze a target into factor coordinates, predict them through factor-specific output heads, and synthesize a complete state. Learned factor coordinates supply a block-structured optimization bias, while orthogonality discourages repeated directions and controls synthesis conditioning. OPF provides the concrete coordinate-organizing mechanism inside the broader interface.

We ask whether the interface produces reusable states, supports intervention and recursive dynamics, and yields coordinate blocks that are well-conditioned and functionally used. Experiments cover terminal readout, paired intervention and PDE prediction, OOD rollout, and control, including a significant HalfCheetah planning improvement.

Our contributions are:

- a cross-domain predictive-state interface that separates domain-specific observation geometry from a common latent prediction and state-reuse contract;
- a factorized shared core whose factor-specific output heads organize predictive coordinates and whose orthogonality-regularized analysis supports well-conditioned complete-state synthesis; and

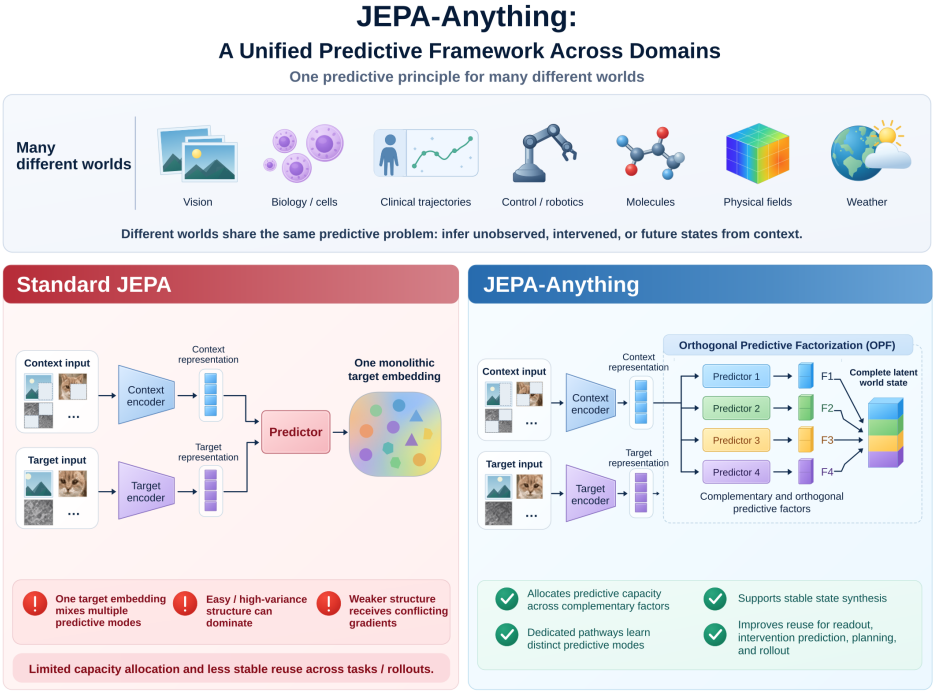

Figure 2: Overview of JEPA-Anything as a shared predictive world-modeling framework. Domain-specific observations and structures are expressed through a common context–target interface. Within the shared predictive core, OPF organizes separately accessible target coordinates through factor-specific output heads and synthesizes their outputs into a complete latent world state.

- evaluations across six primary domain families spanning terminal readout, paired intervention, OOD prediction, rollout, geometry, and functional-factor tests.

2 Related Work

Latent prediction and world models. Latent world models learn states that can be advanced for control and planning (Ha & Schmidhuber, 2018; Hafner et al., 2025; Hansen et al., 2024). DINO-WM plans toward visual goals (Zhou et al., 2025), while V-JEPA 2 adds action-conditioned planning (Assran et al., 2025). I-JEPA, V-JEPA, and Cell-JEPA apply latent prediction to images, video, and transcriptomics (Assran et al., 2023; Bardes et al., 2024; ElSheikh et al., 2026). The JEPA concept was proposed as part of a predictive world-model architecture (LeCun, 2022). JEPA-Anything instead asks which algorithmic contract can be reused across observation families while their tokenizers, encoders, targets, and parameters remain domain-specific.

Representation-space learning. JEPA is related to energy-based and metric-learning approaches that compare learned representations rather than reconstructing every observation detail (LeCun et al., 2006; Chopra et al., 2005; Hadsell et al., 2006). Redundancy- reduction objectives such as Barlow Twins and VICReg shape embedding statistics without explicit target prediction (Zbontar et al., 2021; Bardes et al., 2022), whereas analyses of reconstruction and JEPA objectives motivate emphasizing predictive content while monitoring feature selectivity (Balestriero & LeCun, 2024; Sobal et al., 2022). OPF retains the JEPA target-prediction objective but partitions its output coordinates into separately predicted blocks.

Structured representations. Prior work seeks independent generative factors (Higgins et al., 2017), entity slots (Locatello et al., 2020), object states (Kipf et al., 2020), or identifiable causal factors (Lippe et al., 2022); unsupervised semantic disentanglement is not generally identifiable

(Locatello et al., 2019). VICReg/VJ-VCR, SparseJEPA, and CausalVAE instead use statistical regularization, sparse grouping, or causal structure (Bardes et al., 2022; Drozdov et al., 2024; Hartman & Varshney, 2025; Yang et al., 2020). These methods target non-redundancy, grouping, or semantic structure in different ways. OPF attaches factor-specific predictors to target subspaces and retains an explicit complete-state synthesis map, with semantics supplied by domain structure or downstream analysis.

Predictive modeling across domains. Domain-specific systems pair specialized representations with specialized prediction targets, including single-cell models based on scGPT (Cui et al., 2024; ElSheikh et al., 2026), clinical event forecasting (Shmatko et al., 2025), visual-control benchmarks and simulators (Ahmed et al., 2021; Tassa et al., 2018; Ortiz et al., 2024), physical-field forecasting (Takamoto et al., 2022; Koehler et al., 2024), weather prediction (Rasp et al., 2024), and molecular trajectory forecasting (Thiemann et al., 2026). JEPA-Anything supplies a common predictive-state contract above these observation and encoder choices.

3 JEPA-ANYTHING: A SHARED PREDICTIVE FRAMEWORK

3.1 DOMAIN-SPECIFIC OBSERVATIONS, SHARED PREDICTIVE STATES

JEPA-Anything specifies the boundary between domain-specific and shared components. Images, cells, trajectories, and fields retain suitable tokenizers and encoders; each domain then exposes an available context and requested target through a common predictive-state interface.

Let δ index a domain and $x \sim \mathcal{D}_\delta$. A domain adapter $\mathcal{A}_\delta$ maps x to content tokens $H = \{h_i\}_{i\in\Omega}$ and optional structural descriptors $S = \{s_i\}_{i\in\Omega}$. Descriptors may encode patch coordinates, time, graph position, entity identity, or known forcing. A view sampler $\mathcal{V}_\delta$ selects context indices C and targets T:

$$x \xrightarrow{\mathcal{A}_\delta} (H, S) \xrightarrow{\mathcal{V}_\delta} (H_C, S_C, T, S_T). \tag{1}$$

The adapter, view semantics, and encoder family are domain-specific; the context–target relation and latent predictive interface are shared.

Following the online/EMA target-encoder design of I-JEPA (Assran et al., 2023), an online encoder gives $z_c = f_\theta(H_C, S_C)$, while an exponential-moving average (EMA) target encoder gives $z_t = f_{\bar\theta}(H, S)_t \in \mathbb{R}^d$ for each $t \in T$. Target parameters receive no gradients and are updated by $\bar\theta \leftarrow m\bar\theta + (1-m)\theta$. This includes masked spatial targets, future sequence states, and single-vector transitions as special cases. The interface assumes that context and target describe statistically related states of the same underlying system. This separation allows the framework to change observation geometry without changing the role of the learned world state.

Every instantiation follows the same predictive loop: adapt observations into tokens and descriptors, sample context and targets, encode them with online and EMA networks, predict the target state in latent space, update the trainable components, and then update the target encoder by EMA. The next subsection specifies how JEPA-Anything organizes capacity inside this otherwise common loop.

3.2 ORGANIZING THE SHARED PREDICTIVE STATE

The interface above could use a monolithic predictor, but the diagnostics in Figure 1 motivate an explicit allocation of predictive capacity. We instantiate the shared core with orthogonal predictive factorization. Choose K factor blocks of width r with $Kr = d$, and learn projectors $P_k \in \mathbb{R}^{d\times r}$. OPF analyzes the stopped target state and predicts each coordinate block separately:

$$\widetilde{z}_t = \mathrm{sg}(z_t), \qquad z_t^{(k)} = P_k^\top \widetilde{z}_t, \qquad \widehat{z}_t^{(k)} = q_k(z_c, s_t). \tag{2}$$

The projectors remain trainable. Concatenating the predictions as $\widehat{u}_t = [\widehat{z}_t^{(1)}; \ldots; \widehat{z}_t^{(K)}]$ and defining $P = [P_1, \ldots, P_K]$, the complete target state is synthesized through the Moore–Penrose pseudoinverse of the analysis map:

$$\widehat{z}_t = (P^\top)^\dagger \widehat{u}_t. \tag{3}$$

For an exactly orthogonal P, this reduces to $\widehat{z}_t = \sum_k P_k \widehat{z}_t^{(k)}$. Thus factors are explicit coordinates for prediction, but downstream components still receive a complete d-dimensional state. In the

matched predictor used below, a shared trunk produces h and the factor-specific linear heads concatenate into one dense linear output, $\widehat{u} = Wh+b$. For full-rank square P, $\widehat{z} = (P^\top)^{-1}(Wh+b)$, so concatenating factor heads preserves the dense predictor's function class. Moreover, $\|\widehat{u} - P^\top z\|_2^2 = \|P^\top(\widehat{z}-z)\|_2^2$, which equals $\|\widehat{z}-z\|_2^2$ when P is orthogonal. OPF changes the training parameterization and activity constraints; a softly constrained P can adapt the residual metric, while orthogonality controls synthesis conditioning.

Each factor is trained by magnitude-preserving regression,

$$\mathcal{L}_{\text{pred}} = \frac{1}{K|T|r} \sum_{t \in T} \sum_{k=1}^{K} \|\widehat{z}_t^{(k)} - z_t^{(k)}\|_2^2. \tag{4}$$

To prevent several branches from learning the same target directions, we use

$$\mathcal{L}_{\text{orth}} = \sum_k \|P_k^\top P_k - I_r\|_F^2 + \sum_{i<j} \|P_i^\top P_j\|_F^2. \tag{5}$$

The first term encourages orthonormal columns within each factor; the second penalizes overlap between factor subspaces. This differs from redundancy-reduction losses on embedding statistics (Zbontar et al., 2021; Bardes et al., 2022): OPF constrains the analysis–synthesis geometry of learned predictive target subspaces. OPF also admits an exactly orthogonal QR parameterization (Appendix D), while our task evaluations use the soft penalty above.

Proposition 1 (Complete and stable decomposition)**.** *If $P_k^\top P_k = I_r$, $P_i^\top P_j = 0$ for $i \neq j$, and $Kr = d$, then P is orthogonal. Consequently, for every $z \in \mathbb{R}^d$,*

$$\|P^\top z\|_2^2 = \sum_k \|P_k^\top z\|_2^2 = \|z\|_2^2, \qquad z = \sum_k P_k P_k^\top z, \tag{6}$$

and factor-coordinate error is not amplified by pseudoinverse synthesis: $\|(P^\top)^\dagger(P^\top z + e) - z\|_2 = \|e\|_2$.

The result follows from $P^\top P = PP^\top = I_d$. Without cross-factor orthogonality, repeated directions can make P rank deficient, and nearly repeated directions can make its condition number arbitrarily large. Appendix Proposition 2 extends the result to the approximately orthogonal case. Pseudoinverse synthesis reconstructs exact coordinates $P^\top z$ for any full-rank square P. Orthogonality adds the self-dual relation $(P^\top)^\dagger = P$, unit conditioning, and preservation of coordinate-error magnitude. The analysis below measures these geometric properties.

Orthogonality alone does not ensure that batchwise target variation occupies every projected coordinate. Let $\sigma_{k,j}^{\text{fac}}$ be the empirical standard deviation of factor coordinate (k, j) and σ_j^{enc} that of online-encoder coordinate j. Following the variance-hinge form in VICReg (Bardes et al., 2022), we use

$$\begin{aligned} \mathcal{L}_{\text{fac}} &= \frac{1}{Kr} \sum_{k,j} [\gamma_{\text{fac}} - \sigma_{k,j}^{\text{fac}}]_+, \\ \mathcal{L}_{\text{enc}} &= \frac{1}{d} \sum_j [\gamma_{\text{enc}} - \sigma_j^{\text{enc}}]_+, \end{aligned} \tag{7}$$

where $[a]_+ = \max(0, a)$. Standard deviations are evaluated as $\sqrt{\text{Var} + \varepsilon_{\text{num}}}$, with $\varepsilon_{\text{num}} > 0$, over the batch and sampled target indices; for token-valued contexts, σ_j^{enc} is computed over all valid context tokens. These terms define an additive OPF loss. Let $\mathcal{L}_{\text{base}}^{(\delta)}$ denote the loss already used by the original implementation for domain δ. The optimized training loss is

$$\begin{aligned} \mathcal{L}_{\text{OPF}} &= \mathcal{L}_{\text{pred}} + \lambda_{\text{orth}}\mathcal{L}_{\text{orth}} + \lambda_{\text{fac}}\mathcal{L}_{\text{fac}} + \lambda_{\text{enc}}\mathcal{L}_{\text{enc}}, \\ \mathcal{L}_{\text{train}}^{(\delta)} &= \mathcal{L}_{\text{base}}^{(\delta)} + \mathcal{L}_{\text{OPF}}. \end{aligned} \tag{8}$$

The OPF loss is added to the original objective, preserving domain losses and observation-specific inductive biases while organizing predictive capacity.

The stop-gradient in Equation (2) is applied to the target embedding before projection, while the projectors remain differentiable. Thus, $\mathcal{L}_{\text{pred}}$ backpropagates through the online encoder, the factor-specific predictors, and the projectors; $\mathcal{L}_{\text{orth}}$ and $\mathcal{L}_{\text{fac}}$ update the projectors; and $\mathcal{L}_{\text{enc}}$ updates the online encoder. None of these losses backpropagates into the target encoder, whose parameters are updated only by the EMA rule above. In particular, $\mathcal{L}_{\text{fac}}$ constrains the variation of the projected target coordinates through P, rather than directly updating the predictor heads or the target encoder.

3.3 USING THE SHARED WORLD STATE

For representation tasks, pretraining shapes the online encoder and a task-specific readout consumes its tokens or pooled state; the EMA encoder and predictive heads are discarded. For operational world models, the projectors and predictors are retained. An action, intervention, or forcing ξ_t is placed in the context or target descriptor, and the transition is

$$\widehat{u}_{t+1} = [q_1(z_t, \xi_t, s_{t+1}); \ldots; q_K(z_t, \xi_t, s_{t+1})], \quad \widehat{z}_{t+1} = (P^\top)^\dagger \widehat{u}_{t+1}. \tag{9}$$

Repeated application defines a rollout; a decoder or planner operates on the synthesized state. Within an instantiation, these two modes use the same predictive-state framework and differ only in which learned components are retained.

4 EXPERIMENTS

The experiments test the shared framework at three levels. First, we ask whether the same context–target interface can produce reusable predictive states across different observation geometries. Second, we test operational world modeling, where predicted states are used for interventions, OOD forecasting, recursive dynamics, or planning. Third, geometry and functional interventions examine the organization and use of predictive capacity. Each domain retains its established task-specific metrics.

Across all direct JEPA-versus-JEPA-Anything comparisons, the two models use the same adapter, encoder, context–target sampler, data split, optimization budget, and downstream readout. The inherited base loss is also held fixed. The standard baseline adds a monolithic predictive term, whereas JEPA-Anything adds $\mathcal{L}_{\mathrm{OPF}}$ and its regularizers. Domain baselines establish the performance scale. Unless noted, each domain uses the default architecture and optimization settings of its referenced backbone implementation; matched standard JEPA and OPF runs share all non-OPF hyperparameters. Results in the main text average five training seeds except for visual binding, which uses ten. Appendix Table 3 summarizes the context–target construction, state dimensions, factor widths, seed counts, and evaluation splits used by each reported instantiation. Values written as mean $\pm$ standard deviation use the sample standard deviation across training seeds; reported confidence intervals are computed from paired seed-level differences. We report paired intervals for matched comparisons and seed means otherwise.

4.1 GROUP I: SHARED PREDICTIVE STATES AT TERMINAL READOUT

Controlled visual binding. DINOv3 ViT-S and SigLIP2 Base encoders (Siméoni et al., 2025; Tschannen et al., 2025) are trained by I-JEPA-style block-masked patch prediction (Assran et al., 2023), then frozen. A structured readout must identify both the location and type of a controlled change in MuJoCo scenes (Todorov et al., 2012), including held-out support–operation combinations. Evaluation uses image-disjoint splits, leave-one-cell-out readout, injective Hungarian alignment (Kuhn, 1955), and ten seeds. DINOv3 uses $d = 384$, $K = 4$, $r = 96$; SigLIP2 uses $d = 768$, $K = 4$, $r = 192$. For each source–intervention pair, patch-token differences define an innovation field. INJ is injective held-out-cell accuracy, collapse is the fraction of held-out cells below a 0.5 matching threshold, and recovery measures reconstruction of the learned support–operation grid.

Single-cell state. Following Cell-JEPA (ElSheikh et al., 2026), an scGPT backbone (Cui et al., 2024) is used with two target constructions. For representation learning, masked gene expression predicts the complete cell embedding; this path is pretrained on about 800,000 kidney cells and evaluated on the PBMC-10K clustering task. Perturbation response is evaluated on Adamson and Norman Perturb-seq (Adamson et al., 2016; Norman et al., 2019). For Norman, the predicted target is $z_{\mathrm{response}} = z_{\mathrm{observed}} - z_{\mathrm{control}}$. The Norman projector tensor has shape $[4, 512, 128]$, corresponding to $d = 512$, $K = 4$, and $r = 128$ for this response representation. AvgBIO summarizes cell-type clustering quality; Pearson correlation is computed on absolute post-perturbation expression. Perturbation evaluation follows the encoder $\rightarrow$ factor prediction heads $\rightarrow$ gating $\rightarrow$ latent response residual $\rightarrow$ expression decoder path.

Longitudinal disease state. We evaluate UK Biobank (Sudlow et al., 2015) disease forecasting under the longitudinal protocol of Delphi-2M (Shmatko et al., 2025). Events are dated first occur-

Table 1: Group I terminal-readout results. Standard JEPA and JEPA-Anything columns report seed means. Δ is the difference between the displayed means (JEPA $-$ *Anything* minus Standard JEPA); confidence intervals use paired seed-level differences. Visual binding uses ten seeds and the remaining tasks use five. Clinical AP is macro averaged over outcomes.

Domain	Metric	Reference	Standard JEPA	JEPA-Anything	Δ [95%] paired CI
DINOv3 binding	INJ	Frozen ckpt. (.569)	.572	**.581**	+.009 [.004,.016]
SigLIP2 binding	INJ	Frozen ckpt. (.484)	.483	**.490**	+.007 [.004,.010]
PBMC fine-tuned	AvgBIO	scGPT (.7531)	.7830	**.8301**	+.0471 [.0238,.0553]
PBMC zero-shot	AvgBIO	scGPT (.5288)	.7194	**.7752**	+.0558 [.0297,.0875]
Norman perturbation	Pearson	scGPT (.631)	.787	**.814**	+.027 [.010,.032]
Adamson perturbation	Pearson	scGPT (.905)	.937	**.942**	+.005 [.001,.009]
Clinical forecasting	AP	Cross-attn. fusion (.702)	.711	**.720**	+.009 [.003,.011]

rences of three-character ICD-10 diagnoses and death. Records through 30 June 2020 are split by participant into 402,799 training and 100,639 validation individuals. For longitudinal evaluation, histories are fixed at this cutoff; a one-year gap precedes the outcome window from 1 July 2021 to 1 July 2022. Evaluation covers participants aged 50–80 who are alive at the cutoff, with at least 25 cases required per evaluated outcome. A GPT-2 small online encoder and its EMA target copy (Radford et al., 2019) represent patient history and future health state. OPF predicts four factors of width $r = 192$ and synthesizes a $d = 768$ state for a shared event-score decoder. Cross-attention fusion, standard JEPA, and OPF use the same evaluation cohort, outcome window, and event set. We report macro-averaged event-level average precision (macro AP), computed with scikit-learn (Pedregosa et al., 2011) and averaged over five training seeds. Appendix B.2 specifies the temporal split, label construction, and metric aggregation.

Table 1 answers the first question: the same framework can be instantiated across visual, cellular, and clinical state spaces, and its factorized shared core reports a higher seed mean than the matched monolithic JEPA for every listed terminal-readout metric. The largest mean differences occur on cellular tasks. On DINOv3, collapse decreases from .426 to .417 and grid recovery increases from .645 to .659; on SigLIP2, the corresponding values improve from .514 to .503 and from .679 to .688. Appendix Table 4 reports the complete visual diagnostic. The larger mean difference in zero-shot cellular clustering is consistent with improved reuse beyond the pretraining population.

4.2 Group II: intervention and recursive world dynamics

Intervention-conditioned composition. CITRIS Interventional Pong (Lippe et al., 2022) supplies a pre-intervention observation and labels the externally changed state factors. We train on single-factor changes and test single- and combined-intervention one-step prediction as well as six-step free rollout. In a matched-budget comparison, standard JEPA and soft-orthogonality OPF use the same encoder, trunk, decoder, latent width, training protocol, and exactly 4,657,856 trainable parameters. Across five paired seeds $\{11, 23, 37, 53, 71\}$, mean four-channel MSE decreases from .00952107 to **.006982131** for single interventions (26.67% reduction) and from .00942106 to **.00876528** for combined interventions (6.96% reduction). Thus, the soft-orthogonality OPF model improves intervention prediction under a matched parameter budget. The combined-intervention condition tests whether separately observed changes compose in a joint intervention, while the six-step free rollout remains a separate recursive evaluation reported in Figure 3. Appendix B.6 reports a separate same-architecture control with all three OPF regularizers disabled. Appendix Figure 7 examines the sensitivity to the orthogonality coefficient and factor count.

Table 2: Geometric calibration on CITRIS Interventional Pong. Entries are mean ± sample standard deviation over five seeds. The trained soft-OPF column is computed directly from the raw learned P of the same checkpoints used for the CITRIS task evaluation, before any QR or post-hoc transformation. The exact-QR column is an ideal orthogonal reference.

Metric	Unconstrained P	Trained soft-OPF P	Exact-QR reference
Subspace overlap ↓	$.4550 \pm .0420$	$.00218 \pm .00160$	$(5.18 \pm .34) \times 10^{-16}$
$\sigma_{\min}(P)$ ↑	$.00513 \pm .00233$	$.90222 \pm .03882$	$.999989 \pm .000003$
$\kappa_2(P)$ ↓	438.52 ± 170.00	$1.11077 \pm .04951$	$1.00005 \pm .000004$
Transpose-synthesis NMSE ↓	$.7886 \pm .0299$	$.00113 \pm .00060$	$(2.98 \pm .11) \times 10^{-14}$

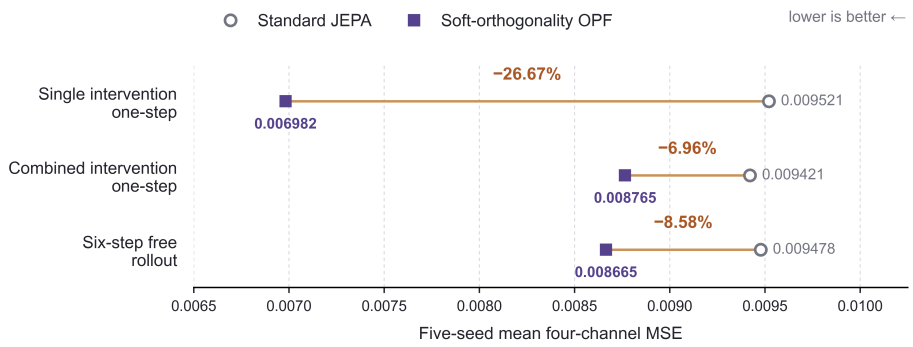

Figure 3: Soft-orthogonality OPF for intervention-conditioned prediction and free rollout on CITRIS Interventional Pong. Lower four-channel MSE is better. Points show means over the five paired seeds; the six-step row is a recursive evaluation.

Orthogonal geometry analysis. Table 2 compares the raw P of the soft-OPF task checkpoints with unconstrained and exact-QR references under the same data, five seeds, and 2,500-step budget. It measures overlap, weakest-direction coverage, conditioning, and transpose-synthesis error.

Transpose-synthesis NMSE evaluates $P(P^\top z)$ to assess self-duality. A post-hoc QR audit changes conditioning while preserving one-step prediction and six-step rollout to numerical precision (Appendix B.8). On the same five soft-OPF CITRIS checkpoints, equal-energy masking yields larger six-step MSE changes for learned factor blocks than for either PCA grouping (Appendix B.7). The random-rotation contrast has a positive mean with an interval spanning zero.

OOD prediction and rollout. We next compare matched four-factor and standard JEPAs on CausalWorld (Ahmed et al., 2021), DeepMind Control and DMC-VB (Tassa et al., 2018; Ortiz et al., 2024), PDEBench (Takamoto et al., 2022), and WeatherBench2 (Rasp et al., 2024). Tests use unseen initial conditions, held-out episodes, or later time periods and six-step autoregressive rollout. Both models receive the same true action or forcing sequence. Figure 4 reports relative changes from five-seed means. OPF lowers mean MSE on all nine plotted tasks; step-1 reductions are 39.8% for Burgers and 43.6% for shallow water, while DMC pixels improve by 0.5%. Closed-loop CausalWorld also has a higher mean return, with variability reported in Figure 4. The APEBench evaluation below reports seed-level results on its tasks. Reaction–diffusion's 18.2% bar uses the OOD MSE in Appendix Table 6; its separate six-step reduction is 22.6%.

For four representative tasks, Appendix Table 5 reports the first and sixth steps to make error accumulation explicit. The APEBench protocol (Koehler et al., 2024) confirms improvements on late-state Burgers (49.5%), six-step Burgers rollout (44.7%), and Kuramoto–Sivashinsky late-state prediction (13.2%), in all paired seeds. Absolute APEBench results are reported in Appendix Table 11.

The OPF ablation (Appendix Table 6) attributes most task-level gain to learned factorized analysis and prediction; orthogonality further reduces error and improves synthesis conditioning.

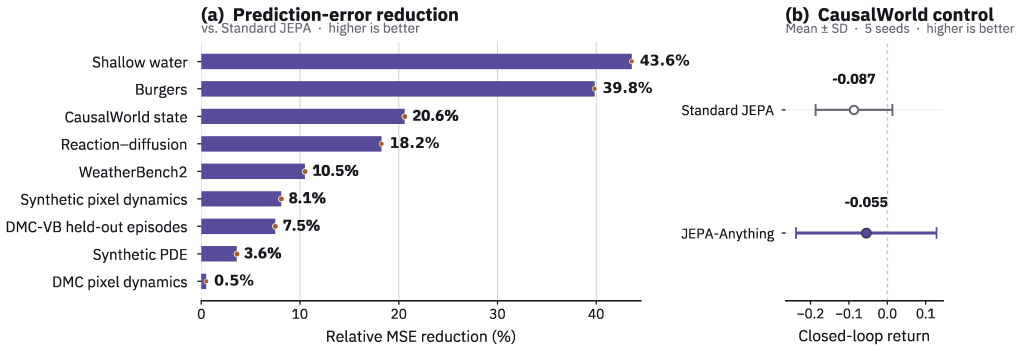

Figure 4: Matched dynamics. (a) Five-seed MSE reductions; rows in Appendix Table 5 use step 1, reaction–diffusion uses OOD MSE from Appendix Table 6, and the rest use their benchmark endpoints. (b) CausalWorld return (mean ± SD, five seeds).

On five-seed residual Burgers, orthogonality reduces the basis condition number from 1.0973 to 1.0048 and worst-direction synthesis amplification from about 1.09 to 1.004 (Appendix Table 10).

Continuous-control planning and factor use. For the Gymnasium MuJoCo environments Hopper-v5, Walker2d-v5, and HalfCheetah-v5 (Towers et al., 2024; Todorov et al., 2012), a cross-entropy method (CEM) planner (de Boer et al., 2005) rolls candidate action sequences through learned latent dynamics and executes the first action of the best sequence. Dense standard JEPA and four-factor OPF are matched within 0.13% training parameters and 1.0% predictor FLOPs. Five training seeds are evaluated on five matched resets with shared CEM randomness. Mean returns change from 2.43 to 12.69 on HalfCheetah. The paired change is $+10.2628$ (95% CI $[5.7244, 13.7619]$), with improvement in all five training seeds. This return comparison uses 80,000 training transitions; Hopper-v5 and Walker2d-v5 provide the corresponding factor-use prediction diagnostics.

Replacing one factor by its training-set mean throughout a 20-step rollout increases MSE for every factor; factor-wise standard deviations span 0.587–0.715. Learned-factor masking exceeds equal-size random or contiguous coordinate masking in all environment–seed pairs ($15/15$). In HalfCheetah, every factor lowers mean CEM return when masked; F1/F3 do so in $5/5$ seeds and F2/F4 in $4/5$. These tests support functional channel use (Figure 5).

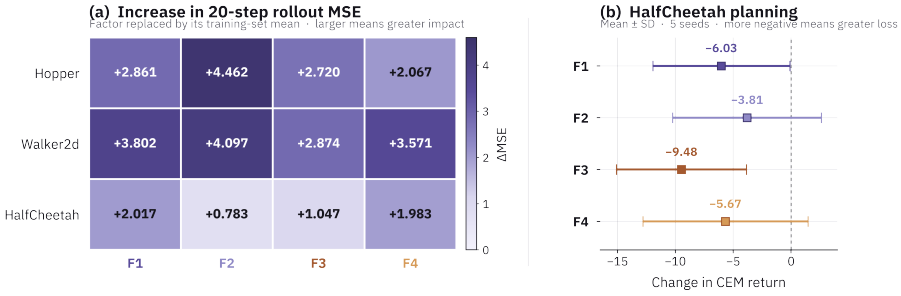

Figure 5: Factor mean-masking effects on 20-step rollout MSE and HalfCheetah CEM return.

5 Discussion

JEPA-Anything provides a shared latent prediction and reuse contract across six domain families and a supplementary molecular study. Factorized heads improve intervention prediction, rollout, and HalfCheetah planning; orthogonality improves synthesis conditioning, and factor masking indicates functional use. Domains retain their adapters, encoders, targets, losses, and parameters. Factor count depends on state dimension, while semantic or causal interpretation requires domain structure: the blocks are predictive coordinates, not guaranteed semantic or causal factors.

AI USE STATEMENT

OpenAI Codex was used for language editing and translation; LaTeX restructuring; consistency, bibliographic, and mathematical checks; cropping and layout of a manuscript figure; and feedback on experimental design, statistical reporting, and the interpretation and presentation of results. The authors reviewed all proposed changes and checked cited sources, derivations, and experimental claims against the original materials. The authors made the scientific decisions and take responsibility for the final manuscript.

ETHICS STATEMENT

The clinical experiment uses de-identified UK Biobank participant data accessed through its controlled research and data-governance process. Analyses were conducted within the scope of an approved research application and applicable institutional ethics and data-use requirements. No participant-level data are redistributed.

REPRODUCIBILITY STATEMENT

Section 3 specifies the additive OPF loss, its combination with the original implementation loss, the activity statistics, and the synthesis map. Section 4 identifies the data, matched-baseline controls, and metrics, while Appendix Table 3 records the context–target construction, factor dimensions, evaluation splits, and seed counts of each instantiation; Appendix Table 13 reports the OPF regularization coefficients used by the listed experiments. Unless otherwise stated, experiments retain the default architecture and optimization settings of the cited backbone implementations, with all non-OPF settings shared by each matched comparison. Tables report absolute metrics where scales are comparable, and figures state the aggregation used for cross-task summaries. The CITRIS matched-budget projection comparison and its soft-orthogonality evaluation are specified in Appendix C.8.

REFERENCES

Britt Adamson, Thomas M. Norman, Marco Jost, Min Y. Cho, James K. Nuñez, Yuwen Chen, Jacqueline E. Villalta, Luke A. Gilbert, Max A. Horlbeck, Marco Y. Hein, Ryan A. Pak, Andrew N. Gray, Carol A. Gross, Atray Dixit, Ophir Parnas, Aviv Regev, and Jonathan S. Weissman. A multiplexed single-cell CRISPR screening platform enables systematic dissection of the unfolded protein response. *Cell*, 167(7):1867–1882.e21, 2016. doi: 10.1016/j.cell.2016.11.048. URL https://doi.org/10.1016/j.cell.2016.11.048.

Ossama Ahmed, Frederik Träuble, Anirudh Goyal, Alexander Neitz, Yoshua Bengio, Bernhard Schölkopf, Manuel Wüthrich, and Stefan Bauer. CausalWorld: A robotic manipulation benchmark for causal structure and transfer learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=SK7Wm7_dHu.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023. doi: 10.1109/CVPR52729.2023.01501. URL https://openaccess.thecvf.com/content/CVPR2023/html/Assran_Self-Supervised_Learning_From_Images_With_a_Joint-Embedding_Predictive_Architecture_CVPR_2023_paper.html.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, François Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. doi: 10.48550/arXiv.2506.09985. URL https://arxiv.org/abs/2506.09985.

Randall Balestriero and Yann LeCun. How learning by reconstruction produces uninformative features for perception. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 2566–2585. PMLR, 2024. URL https://proceedings.mlr.press/v235/balestriero24a.html.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=xm6YD62D1Ub.

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. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=WFYbBOEOtv.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pp. 539–546. IEEE, 2005. doi: 10.1109/CVPR.2005.202. URL https://doi.org/10.1109/CVPR.2005.202.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scGPT: Toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21(8):1470–1480, 2024. doi: 10.1038/s41592-024-02201-0. URL https://doi.org/10.1038/s41592-024-02201-0.

Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, and Reuven Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of Operations Research*, 134(1):19–67, 2005. doi: 10.1007/s10479-005-5724-z. URL https://doi.org/10.1007/s10479-005-5724-z.

Katrina Drozdov, Ravid Shwartz-Ziv, and Yann LeCun. Video representation learning with joint-embedding predictive architectures. *arXiv preprint arXiv:2412.10925*, 2024. URL https://arxiv.org/abs/2412.10925.

Ali ElSheikh, Rui-Xi Wang, Weimin Wu, Yibo Wen, Payam Dibaeinia, Jennifer Yuntong Zhang, Jerry Yao-Chieh Hu, Mei Knudson, Sudarshan Babu, Shao-Hua Sun, Aly A. Khan, and Han Liu. Cell-JEPA: Latent representation learning for single-cell transcriptomics. *arXiv preprint arXiv:2602.02093*, 2026. doi: 10.48550/arXiv.2602.02093. URL https://arxiv.org/abs/2602.02093.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, volume 31, pp. 2451–2463. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/hash/2de5d16682c3c35007e4e92982f1a2ba-Abstract.html.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 1735–1742. IEEE, 2006. doi: 10.1109/CVPR.2006.100. URL https://doi.org/10.1109/CVPR.2006.100.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/s41586-025-08744-2. URL https://doi.org/10.1038/s41586-025-08744-2.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Oxh5CstDJU.

Max Hartman and Lav R. Varshney. SparseJEPA: Sparse representation learning of joint embedding predictive architectures. *arXiv preprint arXiv:2504.16140*, 2025. URL https://arxiv.org/abs/2504.16140.

Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Thomas Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1gax6VtDB.

Felix Koehler, Simon Niedermayr, Rüdiger Westermann, and Nils Thuerey. APEBench: A benchmark for autoregressive neural emulators of PDEs. In *Advances in Neural Information Processing Systems*, volume 37. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/d9875ebcf74bccdc5076acab0dbee62c-Abstract-Datasets_and_Benchmarks_Track.html.

Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955. doi: 10.1002/nav.3800020109. URL https://doi.org/10.1002/nav.3800020109.

Yann LeCun. A path towards autonomous machine intelligence, 2022. URL https://openreview.net/forum?id=BZ5a1r-kVsf. Version 0.9.2, 27 June 2022.

Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu Jie Huang. A tutorial on energy-based learning. In Gökhan Bakir, Thomas Hofmann, Bernhard Schölkopf, Alexander J. Smola, and Ben Taskar (eds.), *Predicting Structured Data*, pp. 191–246. MIT Press, Cambridge, MA, 2006. ISBN 978-0-262-02582-8.

Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M. Asano, Taco Cohen, and Stratis Gavves. CITRIS: Causal identifiability from temporal intervened sequences. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 13557–13603. PMLR, 2022. URL https://proceedings.mlr.press/v162/lippe22a.html.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124. PMLR, 2019. URL https://proceedings.mlr.press/v97/locatello19a.html.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11525–11538. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/8511df98c02ab60aea1b2356c013bc0f-Abstract.html.

Thomas M. Norman, Max A. Horlbeck, Joseph M. Replogle, Alex Y. Ge, Albert Xu, Marco Jost, Luke A. Gilbert, and Jonathan S. Weissman. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793, 2019. doi: 10.1126/science.aax4438. URL https://doi.org/10.1126/science.aax4438.

Joseph Ortiz, Antoine Dedieu, Wolfgang Lehrach, J. Swaroop Guntupalli, Carter Wendelken, Ahmad Humayun, Guangyao Zhou, Sivaramakrishnan Swaminathan, Miguel Lázaro-Gredilla, and Kevin Murphy. DMC-VB: A benchmark for representation learning for control with visual distractors. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0211. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0c946accd3ccc88c09dfae7e1cd40ffe-Abstract-Datasets_and_Benchmarks_Track.html.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12 (85):2825–2830, 2011. URL https://www.jmlr.org/papers/v12/pedregosa11a.html.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, Matthew Chantry, Zied Ben Bouallegue, Peter Dueben, Carla Bromberg, Jared Sisk, Luke Barrington, Aaron Bell, and Fei Sha. WeatherBench 2: A benchmark for the next generation of data-driven global weather models. *Journal of Advances in Modeling Earth Systems*, 16(6):e2023MS004019, 2024. doi: 10.1029/2023MS004019. URL https://doi.org/10.1029/2023MS004019.

Artem Shmatko, Alexander Wolfgang Jung, Kumar Gaurav, Søren Brunak, Laust Hvas Mortensen, Ewan Birney, Tom Fitzgerald, and Moritz Gerstung. Learning the natural history of human disease with generative transformers. *Nature*, 647(8088):248–256, 2025. doi: 10.1038/s41586-025-09529-3. URL https://doi.org/10.1038/s41586-025-09529-3.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025. doi: 10.48550/arXiv.2508.10104. URL https://arxiv.org/abs/2508.10104.

Vlad Sobal, Jyothir S V, Siddhartha Jalagam, Nicolas Carion, Kyunghyun Cho, and Yann LeCun. Joint embedding predictive architectures focus on slow features. *arXiv preprint arXiv:2211.10831*, 2022. doi: 10.48550/arXiv.2211.10831. URL https://arxiv.org/abs/2211.10831.

Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, Bette Liu, Paul Matthews, Giok Ong, Jill Pell, Alan Silman, Alan Young, Tim Sprosen, Tim Peakman, and Rory Collins. UK Biobank: An open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLOS Medicine*, 12(3):e1001779, 2015. doi: 10.1371/journal.pmed.1001779. URL https://doi.org/10.1371/journal.pmed.1001779.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1596–1611. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/0a9747136d411fb83f0cf81820d44afb-Abstract-Datasets_and_Benchmarks.html.

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018. doi: 10.48550/arXiv.1801.00690. URL https://arxiv.org/abs/1801.00690.

Fabian L. Thiemann, Thiago Reschützegger, Massimiliano Esposito, Tseden Taddese, Juan D. Olarte-Plata, and Fausto Martelli. Force-free molecular dynamics through autoregressive equivariant networks. *Nature Machine Intelligence*, 8:764–776, 2026. doi: 10.1038/s42256-026-01227-7. URL https://doi.org/10.1038/s42256-026-01227-7.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109. URL https://doi.org/10.1109/IROS.2012.6386109.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U. Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A

standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*, 2024. doi: 10.48550/arXiv.2407.17032. URL https://arxiv.org/abs/2407.17032.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. doi: 10.48550/arXiv.2502.14786. URL https://arxiv.org/abs/2502.14786.

Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. CausalVAE: Structured causal disentanglement in variational autoencoder. *arXiv preprint arXiv:2004.08697*, 2020. URL https://arxiv.org/abs/2004.08697.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12310–12320. PMLR, 2021. URL https://proceedings.mlr.press/v139/zbontar21a.html.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhou25t.html.

A Approximate Pseudoinverse-Synthesis Stability

The exact decomposition result has a continuous counterpart that applies when the learned analysis matrix is close to orthogonal and synthesis uses the pseudoinverse in Equation (3).

Proposition 2 (Approximate stability). *Suppose $P \in \mathbb{R}^{d\times d}$ satisfies $\|P^\top P - I_d\|_2 \le \epsilon < 1$. Then*

$$\sqrt{1-\epsilon} \le \sigma_i(P) \le \sqrt{1+\epsilon}, \qquad \kappa_2(P) \le \sqrt{\frac{1+\epsilon}{1-\epsilon}}. \tag{10}$$

In particular, P is invertible. For exact factor coordinates $u = P^\top z$ and predicted coordinates $\widehat{u} = u + e$, pseudoinverse synthesis obeys

$$\widehat{z} = (P^\top)^\dagger \widehat{u} = z + (P^\top)^{-1} e, \qquad \|\widehat{z} - z\|_2 \le \frac{1}{\sqrt{1-\epsilon}} \|e\|_2. \tag{11}$$

The eigenvalues of $P^\top P$ lie in $[1-\epsilon, 1+\epsilon]$, which gives Equation (10) and implies $\sigma_{\min}(P) \ge \sqrt{1-\epsilon} > 0$. Hence P is invertible and $(P^\top)^\dagger = (P^\top)^{-1}$. Substituting $\widehat{u} = P^\top z + e$ gives the identity in Equation (11); its bound follows from $\|(P^\top)^{-1}\|_2 = 1/\sigma_{\min}(P)$. Thus pseudoinverse synthesis has no reconstruction bias for exact factor coordinates, while the measured orthogonality residual controls the amplification of factor-prediction error.

B Additional Empirical Evidence

This appendix reports the existing diagnostics and absolute results summarized in the main text. All comparisons use the same matched protocols described in Section 4.

B.1 Cross-domain protocol summary

The main evidence spans six domain families: visual, biological, clinical, control, physical-field, and weather systems. CITRIS and APEBench provide intervention and physical-field evaluation protocols, respectively; molecular forecasting provides an additional instantiation.

Table 3: Instantiations of the shared predictive interface. OPF setup is shown as $d; K \times r$, where $Kr = d$.

Domain	Context → target	OPF setup	Seeds and evaluation
DINOv3 / SigLIP2	Visible patches → masked patch states Masked expression → cell state (representation)	$384; 4\times96$ / $768; 4\times192$	10; image-disjoint, leave-one-cell-out
Single cell	Norman target: $z_{\text{response}} := z_{\text{observed}} - z_{\text{control}}$	Norman: $512; 4\times128$	5; PBMC and perturbation prediction
Clinical	Patient history → future health state	$768; 4\times192$	5; longitudinal evaluation, macro AP
CITRIS	Observation + intervention → next state	$160; 5\times32$	5 paired; single/combined one-step and six-step rollout
Matched dynamics	State, frame, action, or forcing → next state	$128; 4\times32$	5; held-out/OOD and six-step rollout
APEBench	Current field → next field	$128; 4\times32$	5 paired; late holdout and six-step rollout
Residual Burgers	Current field → 50-step free rollout	$128; 4\times32$	5; H50 synthesis stability
Locomotion	State + action → future control state	$32; 4\times8$	5; matched resets and 20-step factor interventions
Molecules	Atomic state → future configuration	$64; 4\times16$	auxiliary aggregate; one-step and 100-step errors

Individual benchmark protocols are specified in Section C.

The matched-dynamics benchmark uses seeds $\{11, 23, 37, 53, 71\}$. Each six-step rollout conditions both models on the same contiguous action or forcing sequence and scores every predicted step against its corresponding target; WeatherBench2 normalization uses training-period statistics only. Locomotion uses seeds $\{42, 123, 456, 789, 2026\}$, five matched environment resets per checkpoint, and paired CEM randomness.

B.2 Clinical cohort and longitudinal prediction protocol

Cohort and event representation. We follow the UK Biobank cohort and longitudinal evaluation protocol of Delphi-2M (Shmatko et al., 2025). Clinical events are first occurrences of three-character ICD-10 diagnoses assembled from self-report, primary-care, hospital-admission, cancer-registry, and death-registry records. Death is an additional outcome. Pre-cutoff records, from birth through 30 June 2020, are partitioned by participant into 402,799 training and 100,639 validation individuals. Longitudinal evaluation draws from the 471,057 participants alive at the cutoff and restricts prediction to ages 50–80. The training and validation split separates participants; longitudinal evaluation separates history and outcomes in calendar time.

Prediction time and outcome window. For every evaluation participant i, the input $\mathcal{H}_i(c)$ contains information available through the common cutoff c, 30 June 2020. Predictions are made from this history without updating it using subsequent records. The following year forms a prediction gap. Outcomes are evaluated from 1 July 2021 to 1 July 2022. For outcome j, the binary label y_{ij} indicates its recorded occurrence during this evaluation window. Thus the task predicts event occurrence in a fixed future window rather than the identity of the immediately next event. All compared methods use the same participant eligibility rules, outcome labels, and evaluation window.

Predictive state and compared models. A GPT-2 small online encoder and EMA target encoder have hidden width $d = 768$ (Radford et al., 2019). The online encoder represents the available history, and the target encoder supplies the future health-state representation during training. Each training pair combines an earlier history with a later health-state target from the same training participant, with both drawn from records on or before 30 June 2020. Standard JEPA predicts that representation through a monolithic pathway. OPF predicts $K = 4$ blocks of width $r = 192$ and synthesizes the state using Equation (3). A shared decoder produces event scores s_{ij}. The JEPA comparison retains the same encoder, data split, and readout design; cross-attention fusion provides the additional fusion baseline in Table 1.

Evaluation and aggregation. Let $\mathcal{E}$ be the common set of evaluated outcomes with at least 25 cases in the evaluation window. For training seed b, we compute

$$\mathrm{AP}^{(b)}_{\text{macro}} = \frac{1}{|\mathcal{E}|} \sum_{j \in \mathcal{E}} \mathrm{AP}\Big(\{y_{ij}\}_i, \{s^{(b)}_{ij}\}_i \Big), \tag{12}$$

where AP is scikit-learn's `average_precision_score`. Each eligible outcome receives equal weight. Table 1 reports the mean of this metric over five training seeds: .702 for cross-attention fusion, .711 for standard JEPA, and .720 for OPF.

B.3 CONTROLLED VISUAL BINDING DIAGNOSTICS

Table 4 expands the visual rows of Table 1. Collapse is the fraction of held-out cells below the matching threshold, and recovery measures reconstruction of the learned support–operation grid.

Table 4: Controlled visual binding with the shared learned-grid readout. The frozen-checkpoint rows require no predictive pretraining; Standard JEPA and JEPA-Anything start from the same checkpoints and use the same readout. Values are averaged over ten seeds.

Backbone	Pretraining	INJ ↑	Collapse ↓	Recovery ↑
DINOv3	Frozen checkpoint	.569	.433	.643
DINOv3	Standard JEPA	.572	.426	.645
DINOv3	JEPA-Anything	**.581**	**.417**	**.659**
SigLIP2	Frozen checkpoint	.484	.511	.676
SigLIP2	Standard JEPA	.483	.514	.679
SigLIP2	JEPA-Anything	**.490**	**.503**	**.688**

B.4 ABSOLUTE MATCHED-DYNAMICS ROLLOUT RESULTS

Figure 4 summarizes relative changes across nine prediction tasks. For four representative tasks, Table 5 reports the first and sixth autoregressive steps. The values are five-seed mean MSE; lower is better.

Table 5: Absolute six-step rollout diagnostics for matched dynamics (five-seed mean MSE; lower is better).

	Standard JEPA		JEPA-Anything	
Task	Step 1	Step 6	Step 1	Step 6
CausalWorld state	.01306	.01307	**.01037**	**.01079**
DMC pixels	.008539	.008594	**.008498**	**.008526**
PDEBench Burgers	.001830	.006369	**.001101**	**.004014**
Shallow water	.007090	.010510	**.003999**	**.006522**

B.5 OPF COMPONENT ABLATION ON PDEBENCH REACTION–DIFFUSION

To attribute the task-level improvement, we ablate encoder variance, factor variance, the analysis map, and orthogonality on PDEBench reaction–diffusion under the matched out-of-distribution and six-step rollout protocol. The baseline and full-OPF endpoints are the same five-seed comparison used for the reaction–diffusion bar in Figure 4. All variants use the same data split, architecture, training budget, and evaluation procedure; only the indicated predictive and regularization components differ. The reported values are five-seed means with sample standard deviations. Here, "Encoder variance" and "Encoder + factor variance" denote the corresponding activity terms in the objective; "Identity block" fixes $P = I$ while retaining factorized heads; "Learned P, no orthogonality" trains P without $\mathcal{L}_{\mathrm{orth}}$; and Full OPF uses the complete objective in Equation (8).

Table 6: Ablation of OPF components on PDEBench reaction–diffusion under the matched OOD and six-step rollout protocol. Lower is better. The baseline is the monolithic $K = 1$ predictive pathway.

Variant	OOD MSE	Six-step rollout MSE
Baseline ($K = 1$)	$.0003935 \pm .0000505$	$.003639 \pm .000365$
Encoder variance	$.0003935 \pm .0000482$	$.003640 \pm .000343$
Encoder + factor variance	$.0003955 \pm .0000621$	$.003504 \pm .000651$
Identity block	$.0003611 \pm .0000478$	$.003359 \pm .000425$
Learned P, no orthogonality	$.0003425 \pm .0000097$	$.002981 \pm .000322$
Full OPF	$\mathbf{.0003217 \pm .0000087}$	$\mathbf{.002818 \pm .000264}$

Encoder variance alone leaves performance essentially unchanged relative to the baseline. The identity-block condition lowers both mean errors. Learning the analysis matrix without orthogonality reduces OOD and six-step rollout error by 13.0% and 18.1%, respectively, relative to the baseline. This is consistent with the adaptive residual geometry induced by $PP^\top$. Adding orthogonality further reduces mean error from .0003425 to .0003217 on OOD MSE and from .002981 to .002818 on six-step rollout MSE (6.1% and 5.5% relative to the no-orthogonality variant). Thus, learned factorized analysis accounts for most of the task-level gain in this ablation, while orthogonality also improves the task means and is directly supported by the geometric and long-horizon synthesis diagnostics.

B.6 CITRIS SAME-ARCHITECTURE REGULARIZATION AUDIT

We additionally compare full OPF with an all-off coordinate predictor on CITRIS Interventional Pong. Both use the same shared trunk, five linear output heads, trainable P, coordinate regression, parameter count, initialization, training-example order, and update count. The all-off control disables $\mathcal{L}_{\text{orth}}$, $\mathcal{L}_{\text{fac}}$, and $\mathcal{L}_{\text{enc}}$. Validation selection fixed the full-OPF coefficients at $(.10, .05, .02)$ and the learning rate for P at 10^{-5} (0.05 times the main learning rate) before five paired-seed runs.

Table 7: CITRIS same-architecture regularization audit. Entries are mean four-channel MSE over five paired seeds; lower is better. Full OPF has lower MSE in all five pairs for each endpoint.

Endpoint	All-off coordinate predictor	Full OPF
Single-intervention one-step	.01161346	**.00701495**
Combined-intervention one-step	.01108196	**.00960596**
Six-step free rollout	.01122661	**.01060005**

This audit estimates the joint contribution of the three regularizers within a fixed coordinate predictor. Its validation-selected projector learning rate and comparator differ from the standard-JEPA comparison in Figure 3, so their absolute MSE values are reported separately.

B.7 CITRIS COORDINATE-BASIS MASKING

We applied coordinate-basis masking to the same five trained soft-OPF CITRIS checkpoints, keeping the predictive model fixed. The latent state has $d = 160$ coordinates partitioned into $K = 5$ blocks of $r = 32$. We compare masking in its learned factor basis with three alternative coordinate bases. PCA directions are fitted to training-set latent states. Sorted PCA groups consecutive directions in descending eigenvalue order into blocks of 32. Balanced PCA assigns those same ordered directions to five blocks in round-robin order, distributing high-variance directions across blocks. For the random control, QR decomposition of a Gaussian matrix gives an orthogonal basis; 20 rotations are generated per training seed and their results are averaged within that seed.

Block means and the perturbation energy induced by mean masking are estimated from training data. Learned and control blocks are paired by this training-set energy. At test time, each paired block is moved toward its own mean. For each sample and rollout step, the larger of the two latent perturbations is scaled down so that their Euclidean norms match immediately before decoding. We measure the increase in six-step prediction MSE relative to the unmasked rollout, then subtract the control-basis increase from the learned-basis increase. Positive differences therefore indicate greater sensitivity to masking the learned coordinates at matched perturbation energy.

Figure 6 shows five seed-level differences and their means with 95% confidence intervals. The mean differences (in units of 10^{-4}) are 5.54 $[2.05, 9.02]$ against balanced PCA, 6.95 $[3.64, 10.25]$ against sorted PCA, and 3.60 $[-1.24, 8.44]$ against random rotations. Both PCA comparisons favor the learned factor basis under matched perturbation energy; the random-rotation interval includes zero.

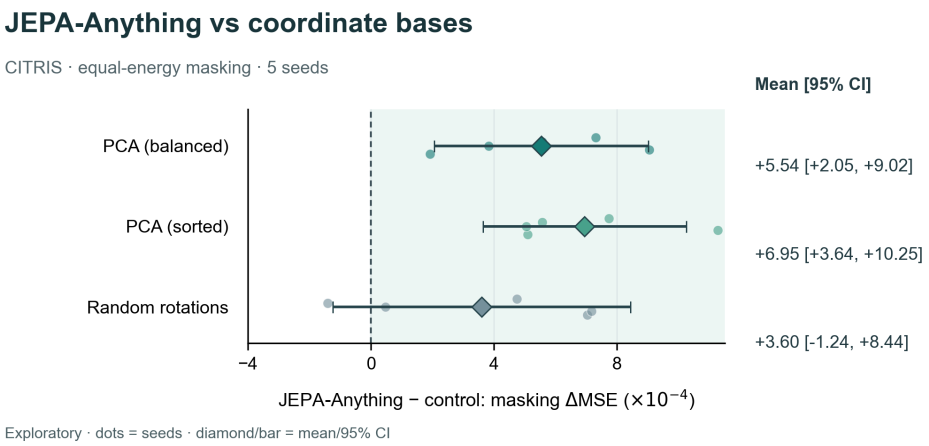

Figure 6: Equal-energy coordinate-basis masking on CITRIS Interventional Pong using the same trained soft-OPF checkpoints as the task comparison. The horizontal axis is the learned-basis minus control-basis increase in six-step prediction MSE, in units of 10^{-4}, after matching latent perturbation norms before decoding. Positive values indicate greater sensitivity to masking learned factors. Dots are five training-seed values; each random-rotation value first averages 20 rotations within its seed. Diamonds are means, horizontal bars are 95% confidence intervals across seeds, and the dashed vertical line marks zero.

B.8 Post-hoc coordinate-preserving QR audit

We perform a post-hoc audit to separate coordinate geometry from predictive behavior. For an invertible analysis matrix $P = QR$, with Q orthogonal and R upper triangular, a prediction $\widehat{y} = (P^\top)^{-1} f$ can be written as $\widehat{y} = Q f_{\text{new}}$ with $f_{\text{new}} = R^{-\mathsf{T}} f$. For a linear coordinate head $f = Wh + b$, we apply the corresponding transformation $W_{\text{new}} = R^{-\mathsf{T}} W$ and $b_{\text{new}} = R^{-\mathsf{T}} b$. This changes the coordinate basis while preserving the complete prediction function. We audit Full OPF and an all-off model with all OPF regularizers disabled, using five checkpoints ($\{11, 23, 37, 53, 71\}$) in both FP32 and FP64. No model is retrained or selected after the audit, and original and transformed models receive the same inputs.

Table 8: Conditioning before and after the function-preserving QR transform. Entries are means over five checkpoints; the pre-QR range is reported across those checkpoints.

Configuration	dtype	$\kappa_2(P)$ before	$\kappa_2(Q)$ after	Pre-QR range
Full OPF	FP32	1.11077	1.0000508	1.0426–1.18725
Full OPF	FP64	1.11076	1.0000000	1.0426–1.18725
All-off	FP32	3.40261	1.0000226	3.06147–3.94601
All-off	FP64	3.40260	1.0000000	3.06147–3.94601

Table 9: Function-preservation errors under the post-hoc QR transform. Values are maxima across the audited inputs and checkpoints.

Configuration	dtype	Max one-step difference	Max six-step difference	Max compensated coordinate residual
Full OPF	FP32	6.94×10^{-6}	1.27×10^{-7}	2.04×10^{-5}
Full OPF	FP64	2.59×10^{-14}	1.79×10^{-16}	6.33×10^{-14}
All-off	FP32	5.07×10^{-6}	7.45×10^{-9}	1.87×10^{-5}
All-off	FP64	1.51×10^{-14}	2.43×10^{-17}	5.86×10^{-14}

The FP64 discrepancies are at numerical precision, while the larger FP32 differences are consistent with finite-precision QR and matrix operations. Uncompensated coordinate MSE, Gram penalties,

and activity hinges are not invariant to this basis change; after applying the corresponding $R^\top$ residual compensation, the maximum FP64 coordinate-MSE difference is 7.63×10^{-17}. The audit establishes function-preserving geometric reparameterization of the complete state; a global QR transform can mix factor blocks.

B.9 Residual long-horizon Burgers stability

We evaluate synthesis stability in a five-seed residual Burgers setting.

Table 10: Residual Burgers synthesis stability over five seeds. Lower is better.

Metric	Without orthogonality	With orthogonality
Mean condition number	1.0973	$\mathbf{1.0048}$
Worst-direction amplification	≈ 1.09	$\mathbf{1.004}$

B.10 APEBench evaluation

Table 11 reports the absolute metrics for the APEBench comparison. The evaluated models use an MLP whose first linear-layer weight has shape $[256, 64]$; the stored `opf.raw` tensor has shape $[4, 128, 32]$, corresponding to $d = 128$, $K = 4$, and $r = 32$.

Table 11: Stable prediction improvements in the APEBench evaluation. Lower MSE is better; metric entries are mean $\pm$ sample standard deviation.

System	Metric	Standard JEPA	JEPA-Anything	Reduction	Wins
Burgers	Held-out late MSE	$.16621 \pm .01740$	$\mathbf{.08389 \pm .01770}$	49.5%	5/5
Burgers	Six-step rollout MSE	$.29186 \pm .00682$	$\mathbf{.16153 \pm .03500}$	44.7%	5/5
Kuramoto–Sivashinsky	Held-out late MSE	$1.1012 \pm .0763$	$\mathbf{.9558 \pm .0340}$	13.2%	5/5

B.11 Supplementary molecular forecasting

We additionally apply OPF to the 64-channel $l = 1$ representation of a TrajCast-style equivariant molecular forecaster (Thiemann et al., 2026), using $K = 4$ factors of width $r = 16$. Table 12 reports aggregate one-step displacement MAE and median final-position RMSD after 100 force-free autoregressive steps.

Table 12: Supplementary molecular forecasting results in Å. Lower is better; entries are aggregate one-step and long-rollout summaries.

System	Metric	Scratch	TrajCast-JEPA	JEPA-Anything
Water	MAE	.00387	.00452	**.00376**
Water	RMSD	3.331	2.536	**2.459**
Quartz	MAE	.01080	.01043	**.01011**
Quartz	RMSD	2.089	1.912	**1.877**
Paracetamol	MAE	.00901	.00777	**.00765**
Paracetamol	RMSD	3.155	1.868	**1.846**
Benzene	MAE	2.59×10^{-5}	2.10×10^{-5}	$\mathbf{2.05 \times 10^{-5}}$
Benzene	RMSD	.0958	.0701	**.0699**

B.12 Additive OPF coefficients

Table 13 reports the dimensions and the three additive OPF coefficients in Equation (8) used in the listed experiments. Domain-specific terms already contained in $\mathcal{L}_{\text{base}}^{(\delta)}$ retain their native definitions. Dimensions agree with Table 3.

Table 13: OPF dimensions and coefficients of the three additive OPF regularizers in Equation (8). Domain-specific base-loss terms are excluded.

Experiment	d	K	r	λ_{orth}	λ_{fac}	λ_{enc}
CITRIS Interventional Pong	160	5	32	.10	.05	.02
CausalWorld state/control	128	4	32	.10	.05	.02
DMC pixel dynamics	128	4	32	.10	.05	.02
DMC-VB held-out episodes	128	4	32	.10	.05	.02
PDEBench Burgers	128	4	32	.10	.05	.02
PDEBench reaction–diffusion	128	4	32	.10	.05	.02
PDEBench shallow water	128	4	32	.10	.05	.02
Synthetic PDE	128	4	32	.10	.05	.02
Synthetic pixel dynamics	128	4	32	.10	.05	.02
WeatherBench2	128	4	32	.10	.05	.02
APEBench Burgers / Kuramoto–Sivashinsky	128	4	32	.10	.05	.02
Hopper-v5, 20-step	32	4	8	.02	.10	.02
Walker2d-v5, 20-step	32	4	8	.02	.10	.02
HalfCheetah-v5, 20-step	32	4	8	.02	.10	.02

C BENCHMARK-SPECIFIC EXPERIMENTAL PROTOCOLS

This section specifies the experimental interfaces and evaluation conditions for each benchmark. Unless stated otherwise, the direct comparison changes the monolithic prediction pathway to OPF while retaining the domain adapter, encoder, data sampler, split, optimization budget, readout, and base loss. Dimensions and additive regularizer coefficients are collected in Table 13. Seeds index training runs; resets and rollout starts are evaluation replicates.

C.1 DINOV3 CONTROLLED VISUAL BINDING

The DINOv3 ViT-S experiment uses $d = 384$, $K = 4$, and $r = 96$. Standard JEPA and OPF start from the same checkpoint and learn block-masked patch prediction. Evaluation uses controlled source/intervened image pairs, image-disjoint splits, and leave-one-cell-out support–operation combinations. Both trained encoders use the same learned-grid readout; the frozen-checkpoint reference receives no predictive pretraining. Injective matching accuracy (INJ) uses Hungarian alignment. Collapse is the fraction of held-out cells below the .5 matching threshold; recovery measures reconstruction of the learned support–operation grid. Table 4 averages ten seeds.

C.2 SIGLIP2 CONTROLLED VISUAL BINDING

The SigLIP2 Base experiment uses $d = 768$, $K = 4$, and $r = 192$. Its standard-JEPA and OPF arms start from the same SigLIP2 checkpoint and use block-masked patch prediction. The image-disjoint, leave-one-cell-out support–operation evaluation and learned-grid readout follow the visual protocol in Section C.1. INJ, collapse, and grid recovery are averaged over ten seeds in Table 4. The SigLIP2 frozen-checkpoint row is an inference reference.

C.3 PBMC FINE-TUNED REPRESENTATION EVALUATION

The cellular representation experiment uses a scGPT-based encoder and predicts a full-cell EMA representation from masked expression input. Pretraining uses approximately 800,000 kidney cells, and the downstream population is PBMC-10K. This endpoint evaluates the fine-tuned representation with AvgBIO, an aggregate of biological clustering scores. Table 1 compares the scGPT reference, standard JEPA, and OPF and reports five-seed aggregates under the cellular evaluation convention. The zero-shot endpoint transfers the kidney-pretrained encoder directly to PBMC-10K.

C.4 PBMC ZERO-SHOT REPRESENTATION EVALUATION

The zero-shot endpoint transfers the kidney-pretrained cellular representation directly to PBMC-10K. It evaluates AvgBIO and compares the scGPT reference, standard JEPA, and OPF in Table 1. Results are averaged over five seeds. This endpoint assesses transfer of the kidney-pretrained representation to the PBMC population.

C.5 NORMAN PERTURBATION PREDICTION

The target is a perturbation-induced latent response, $z_{\text{observed}} - z_{\text{control}}$, which is decoded into expression. The encoder feeds factor prediction heads and a gating module before response synthesis and expression decoding. The projector shape is $[4, 512, 128]$, corresponding to $d = 512$, $K = 4$, and $r = 128$. The endpoint in Table 1 is Pearson correlation on decoded absolute post-perturbation expression. The reported comparison includes scGPT, standard JEPA, and OPF and uses the manuscript's five-seed cellular convention.

C.6 ADAMSON PERTURBATION PREDICTION

Adamson supplies a second perturbation-prediction dataset. Table 1 compares scGPT, standard JEPA, and OPF using Pearson correlation on absolute post-perturbation expression, with five seeds reported for the cellular comparison.

C.7 Clinical event prediction

Section B.2 specifies the UK Biobank longitudinal protocol. Histories end on 30 June 2020; after a one-year gap, event occurrence is evaluated from 1 July 2021 to 1 July 2022 in participants aged 50–80 who are alive at the cutoff. OPF uses a GPT-2 small online/EMA encoder with $d = 768$, $K = 4$, and $r = 192$. Outcomes with at least 25 cases enter the macro-averaged average precision in Equation (12). Table 1 reports means over five training seeds under the same cohort, window, and event set for all three clinical comparators.

C.8 CITRIS Interventional Pong prediction

The model receives a pre-intervention observation and labels identifying the externally changed factors, and predicts the next state. Training uses single-factor changes. Evaluation covers single-intervention one-step prediction, combined-intervention one-step prediction, and six-step free rollout. The matched one-step comparison uses the same encoder, trunk, decoder, latent width, training protocol, and exactly 4,657,856 trainable parameters for standard JEPA and soft-orthogonality OPF. Five paired seeds $\{11, 23, 37, 53, 71\}$ are used. The projector has $d = 160$, $K = 5$, and $r = 32$, and OPF uses the soft orthogonality penalty in Equation (8). The one-step means are .00952107 versus **.006982131** for single interventions and .00942106 versus **.00876528** for combined interventions. The six-step free-rollout evaluation is reported separately in Figure 3.

C.9 CITRIS orthogonality and factor-count sensitivity

Figure 7 varies the orthogonality coefficient with $K = 5$ and separately compares factor counts at $d = 160$. Increasing λ_{orth} reduces the condition number of P; the plotted single-intervention MSE is lowest near the selected value of .10, while the combined-intervention MSE is lowest without orthogonality. The factor-count panel reports percentage changes relative to $K = 4$ and does not include the $K = 5$ configuration used in the coefficient sweep.

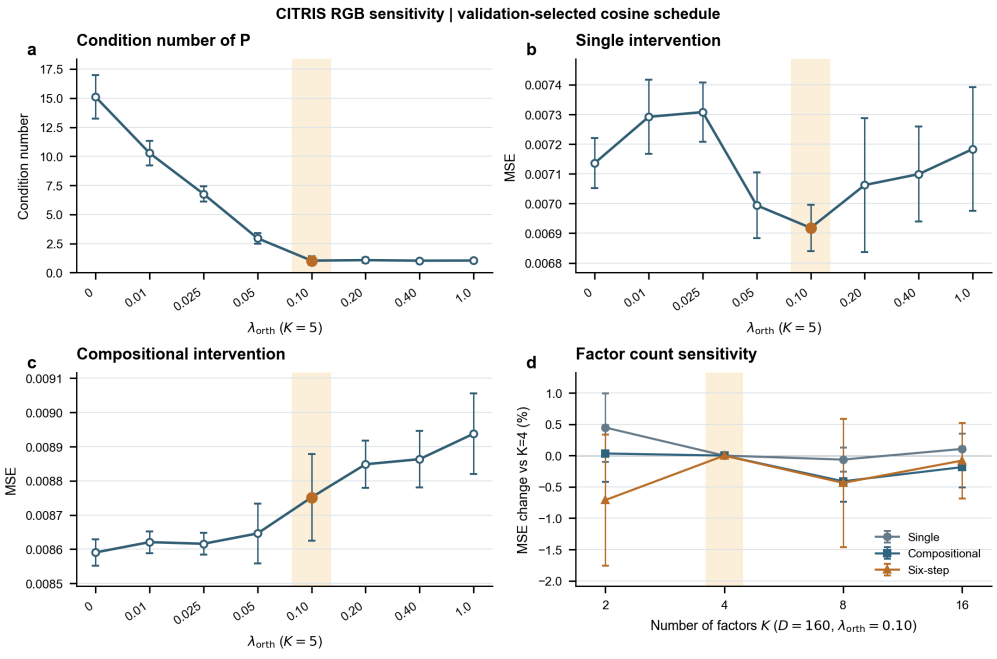

Figure 7: CITRIS RGB sensitivity under the validation-selected cosine schedule. (a)–(c) Condition number of P and one-step four-channel MSE versus λ_{orth} with $K = 5$; shading marks the selected $\lambda_{\text{orth}} = .10$. (d) MSE change for different factor counts at $d = 160$ and $\lambda_{\text{orth}} = .10$, relative to $K = 4$ (shaded).

C.10 CITRIS GEOMETRY DIAGNOSTIC

This diagnostic compares an unconstrained learned projection, the raw learned projection of the trained soft-orthogonality OPF checkpoints, and an exact-QR orthogonal basis under the same data, five seeds, and 2,500 training steps. Table 2 reports mean and sample standard deviation of subspace overlap, $\sigma_{\min}(P)$, $\kappa_2(P)$, and transpose-synthesis normalized MSE. The reconstruction uses $u = P^\top z$ followed by Pu. The soft-OPF statistics are measured before any QR or post-hoc transformation, while the exact-QR column is an ideal geometric reference.

C.11 SHARED SETTINGS FOR MATCHED DYNAMICS

The following ten endpoints share the matched-dynamics comparison: $d = 128$, $K = 4$, $r = 32$, coefficients $(.10, .05, .02)$ in the order $(\lambda_{\text{orth}}, \lambda_{\text{fac}}, \lambda_{\text{enc}})$, and training seeds $\{11, 23, 37, 53, 71\}$. Standard JEPA and OPF use the same adapter, encoder, sampler, split, optimization budget, and readout. For six-step prediction, both models receive the same contiguous true action or forcing sequence where applicable and recursively predict the state; each predicted step is scored against its corresponding target. Figure 4(a) reports relative MSE changes from five-seed means.

C.12 CAUSALWORLD STATE PREDICTION

This endpoint predicts the next CausalWorld state from the current state and action and evaluates six-step recursive prediction under the shared matched-dynamics settings above. Table 5 reports five-seed mean MSE at steps 1 and 6. Closed-loop return is reported in the control endpoint below.

C.13 CAUSALWORLD CLOSED-LOOP CONTROL

This endpoint evaluates the learned model through closed-loop control in CausalWorld. Figure 4(b) reports return as mean and standard deviation over the five matched-dynamics training seeds. It uses the CausalWorld state/control dimensions and regularizer coefficients in Table 13.

C.14 DEEPMIND CONTROL PIXEL DYNAMICS

This endpoint predicts future visual observations conditioned on actions. It uses the shared matched-dynamics model and five seeds, with six-step recursive prediction under matched action sequences. Table 5 reports pixel-dynamics MSE at steps 1 and 6. DMC-VB uses held-out episodes.

C.15 DMC-VB HELD-OUT EPISODES

The DMC-VB endpoint evaluates prediction on held-out episodes using the shared matched-dynamics dimensions, regularizers, and five seeds. Figure 4(a) reports its relative MSE change.

C.16 PDEBENCH BURGERS

The Burgers endpoint predicts the next field from the current field and evaluates a six-step autoregressive field trajectory under the matched-dynamics settings. Table 5 reports MSE at steps 1 and 6. APEBench and residual Burgers use their own reported protocols and metric scales.

C.17 PDEBENCH REACTION–DIFFUSION

This endpoint predicts the next reaction–diffusion field and uses the shared matched-dynamics dimensions, regularizers, and five seeds. The OOD endpoint uses held-out initial conditions, and six-step prediction is recursive. Figure 4(a) reports the OOD MSE reduction, while Table 6 reports the same Standard JEPA and full-OPF endpoints together with the component ablations and six-step rollout MSE.

C.18 PDEBENCH SHALLOW WATER

The shallow-water endpoint predicts successive fields with the shared matched-dynamics model and five seeds. It evaluates six-step recursive prediction. Table 5 supplies absolute mean MSE for steps 1 and 6, in addition to the relative comparison in Figure 4(a).

C.19 SYNTHETIC PDE DYNAMICS

The synthetic PDE endpoint is a generated-data prediction task in Figure 4(a). It uses the shared matched-dynamics configuration and five-seed comparison with six-step recursive prediction.

C.20 SYNTHETIC PIXEL DYNAMICS

This generated visual-dynamics endpoint uses the shared matched-dynamics configuration and five seeds; its six-step comparison is included in Figure 4(a).

C.21 WEATHERBENCH2

The weather endpoint predicts future states in the matched-dynamics comparison with $d = 128$, four factors, and five seeds. Evaluation includes six-step recursive prediction. Normalization uses training-period statistics only. Figure 4(a) reports the relative MSE change.

C.22 APEBENCH BURGERS

The APEBench Burgers evaluation uses $d = 128$, $K = 4$, $r = 32$ and coefficients $(.10, .05, .02)$. The MLP first-layer weight has shape $[256, 64]$, and the raw projector tensor has shape $[4, 128, 32]$. Five paired seeds compare standard JEPA and OPF on normalized held-out late-state MSE and six-step rollout MSE. Table 11 reports mean $\pm$ sample standard deviation and the number of paired seed wins. These normalized errors use the APEBench metric scale; PDEBench Burgers uses the matched-dynamics metric scale.

C.23 APEBENCH KURAMOTO–SIVASHINSKY

The Kuramoto–Sivashinsky endpoint uses the APEBench configuration, $d = 128$, $K = 4$, $r = 32$, and coefficients $(.10, .05, .02)$. Table 11 reports normalized held-out late-state MSE as mean $\pm$ sample standard deviation over five paired seeds, together with paired wins.

C.24 RESIDUAL BURGERS SYNTHESIS STABILITY

This five-seed residual Burgers evaluation uses $d = 128$, $K = 4$, and $r = 32$ in a 50-step free-rollout setting. The comparison removes or includes orthogonality and reports the learned basis condition number and worst-direction synthesis amplification in Table 10. These metrics characterize the conditioning and error amplification of the synthesis operator.

C.25 HOPPER-V5 CONTROL AND FACTOR USE

The Hopper-v5 experiment uses state/action-conditioned latent dynamics with $d = 32$, $K = 4$, $r = 8$ and coefficients $(.02, .10, .02)$. Standard JEPA and OPF have matched data and optimization budgets, with training parameters within 0.13% and predictor FLOPs within 1.0%. Training seeds are $\{42, 123, 456, 789, 2026\}$; evaluation uses five matched resets per checkpoint and shared CEM randomness. The planner rolls out candidate action sequences and executes the first action of the best sequence. Factor-use analysis applies factor mean-masking repeatedly during a 20-step rollout and measures prediction MSE.

C.26 WALKER2D-V5 CONTROL AND FACTOR USE

The Walker2d-v5 experiment uses state/action-conditioned latent dynamics with $d = 32$, $K = 4$, $r = 8$ and coefficients $(.02, .10, .02)$. Standard JEPA and OPF have matched data and optimization

budgets, with training parameters within 0.13% and predictor FLOPs within 1.0%. Training seeds are $\{42, 123, 456, 789, 2026\}$; evaluation uses five matched resets per checkpoint and shared CEM randomness. The planner rolls out candidate action sequences and executes the first action of the best sequence. Factor-use analysis applies factor mean-masking repeatedly during a 20-step rollout and measures prediction MSE.

C.27 HalfCheetah-v5 control and factor use

The HalfCheetah-v5 experiment uses state/action-conditioned latent dynamics with $d = 32$, $K = 4$, $r = 8$ and coefficients $(.02, .10, .02)$. Standard JEPA and OPF have matched data and optimization budgets, with training parameters within 0.13% and predictor FLOPs within 1.0%. Training seeds are $\{42, 123, 456, 789, 2026\}$; evaluation uses five matched resets per checkpoint and shared CEM randomness. The planner rolls out candidate action sequences and executes the first action of the best sequence. Factor-use analysis applies factor mean-masking repeatedly during a 20-step rollout and measures prediction MSE. The reported mean return changes from 2.43 to 12.69; the paired difference is 10.2628 with 95% interval $[5.7244, 13.7619]$. This comparison uses 80,000 training transitions. The HalfCheetah analysis also evaluates factor masking within CEM control.

C.28 Water molecular forecasting

The water comparison evaluates scratch training, TrajCast-JEPA, and OPF with a TrajCast-style equivariant forecaster. OPF factorizes the 64-channel $l = 1$ representation with $K = 4$ and $r = 16$. Table 12 reports one-step displacement MAE and median final-position RMSD after 100 force-free autoregressive steps, in Å. The rollout endpoint measures the final-position error at step 100.

C.29 Quartz molecular forecasting

The quartz comparison evaluates scratch training, TrajCast-JEPA, and OPF with a TrajCast-style equivariant forecaster. OPF factorizes the 64-channel $l = 1$ representation with $K = 4$ and $r = 16$. Table 12 reports one-step displacement MAE and median final-position RMSD after 100 force-free autoregressive steps, in Å. The rollout endpoint measures the final-position error at step 100.

C.30 Paracetamol molecular forecasting

The paracetamol comparison evaluates scratch training, TrajCast-JEPA, and OPF with a TrajCast-style equivariant forecaster. OPF factorizes the 64-channel $l = 1$ representation with $K = 4$ and $r = 16$. Table 12 reports one-step displacement MAE and median final-position RMSD after 100 force-free autoregressive steps, in Å. The rollout endpoint measures the final-position error at step 100.

C.31 Benzene molecular forecasting

The benzene comparison evaluates scratch training, TrajCast-JEPA, and OPF with a TrajCast-style equivariant forecaster. OPF factorizes the 64-channel $l = 1$ representation with $K = 4$ and $r = 16$. Table 12 reports one-step displacement MAE and median final-position RMSD after 100 force-free autoregressive steps, in Å. The rollout endpoint measures the final-position error at step 100.

D Exactly Orthogonal OPF via QR Parameterization

OPF can enforce orthogonality by construction while retaining the same factor-specific prediction and synthesis interface. Let $A \in \mathbb{R}^{d\times d}$ be a trainable full-rank matrix. At each forward pass, compute $A = QR$ and set $P = Q = [Q_1, \ldots, Q_K]$, with $Q_k \in \mathbb{R}^{d\times r}$ and $Kr = d$. Fix the QR sign ambiguity by requiring $R_{ii} > 0$, using matched sign flips of columns of Q and rows of R. A single QR decomposition of the full matrix ensures both within-factor and cross-factor orthogonality; separate decompositions of individual blocks would not ensure the latter. The OPF analysis, prediction, and synthesis become

$$z_t^{(k)} = Q_k^\top \operatorname{sg}(z_t), \qquad \widehat{z}_t^{(k)} = q_k(z_c, s_t), \qquad \widehat{z}_t = \sum_k Q_k \widehat{z}_t^{(k)}.$$

Gradients pass through Q to A, with stop-gradient applied only to the target embedding. Equation (8) retains the base loss, prediction loss, and activity terms, while the orthogonality penalty is omitted because it is identically zero in exact arithmetic. An implementation can initialize A orthogonally and monitor its smallest singular value to avoid ill-conditioned QR gradients; finite-precision orthogonality holds up to numerical error.

This parameterization gives $\kappa_2(Q) = 1$ and self-dual synthesis in exact arithmetic. It also gives $\|\widehat{u} - Q^\top z\|_2^2 = \|\widehat{z} - z\|_2^2$, so the prediction loss uses the Euclidean state-space metric; learning the orthogonal basis changes the coordinate parameterization rather than this metric. Task results above use soft OPF; the exact-QR reference in Table 2 and the post-hoc transformation in Appendix B.8 are geometric diagnostics.