YOUR PROBABILISTIC JEPA IS SECRETLY A HIDDEN MARKOV MODEL: A STATE-SPACE INTERPRETATION OF JOINT-EMBEDDING PREDICTIVE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

A hidden Markov model (HMM) combines three roles: inference of a hidden-state belief from observations, propagation through a Markov transition, and emission back to observation space. We show that full, time-indexed Predictive Information Bottleneck VJEPA (PIB-VJEPA) exposes the same computational structure: a stochastic context encoder plays the role of an amortized filtering distribution, a probabilistic predictor defines latent-state dynamics, and a decoder, inverse target encoder, or induced implicit conditional supplies the emission direction. We formalize four progressively stronger levels of correspondence, from computational analogy to exact sequence-level and model-and-objective equivalence, and give sufficient conditions under which a probabilistic temporal JEPA admits an exact HMM representation. We instantiate this view with Markov-Chain JEPA (MCJEPA), which replaces the latent predictor by a learned transition matrix; in the finite time-homogeneous case, matrix powers guarantee exact multi-horizon Chapman–Kolmogorov consistency. The framework extends naturally to conditioned discrete-state transitions, continuous-state Markov kernels, continuous-time dynamics, and deterministic Dirac-kernel dynamics. We further reinterpret predictive information-bottleneck learning as Markov-state construction: compression promotes *minimality*, while residual predictability diagnoses *insufficiency*. Controlled experiments support transition recovery and composition, the filtering interpretation of the context encoder, recovery of a known minimal predictive state, and HMM-style sequence training of probabilistic JEPA, including hybrid JEPA–HMM objectives. Together, these results give probabilistic temporal JEPA a principled state-space interpretation and connect joint-embedding predictive learning with probabilistic dynamical modeling.

1 INTRODUCTION

Joint-Embedding Predictive Architectures (JEPAs) learn by predicting a target *representation* from an observed context rather than reconstructing the target observation (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024). Variational JEPA (VJEPA) makes this prediction probabilistic, replacing a point predictor with a conditional distribution over future latent states (Huang, 2026c), and the full Predictive Information Bottleneck (PIB) extension additionally makes the current representation stochastic and controls how much information it retains about the observed history (Huang, 2026b). This progression raises a natural question: *what familiar probabilistic model is hidden inside a fully stochastic temporal JEPA?*

The basic state-space analogy is immediate. An HMM and a full PIB-VJEPA equipped with an observation model follow the pipelines

$$\begin{aligned} &\text{HMM:} && X_{\le t} \to p(S_t \mid X_{\le t}) \to p(S_{t+1} \mid S_t) \to p(X_{t+1} \mid S_{t+1}), \\ &\text{PIB-VJEPA:} && X_{\le t} \to q_\theta(Z_t \mid X_{\le t}) \to p_\phi(Z_{t+1} \mid Z_t, \xi_t) \to p_\psi(X_{t+1} \mid Z_{t+1}), \end{aligned} \tag{1}$$

where X_t are observations (frames, pixels, measurements), Z_t is a stochastic predictive representation, and ξ_t is side information such as an action or elapsed time (Fig. 1). The subtlety is not the architecture but its precise probabilistic and training interpretation. The stochastic encoder maps observations to

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.

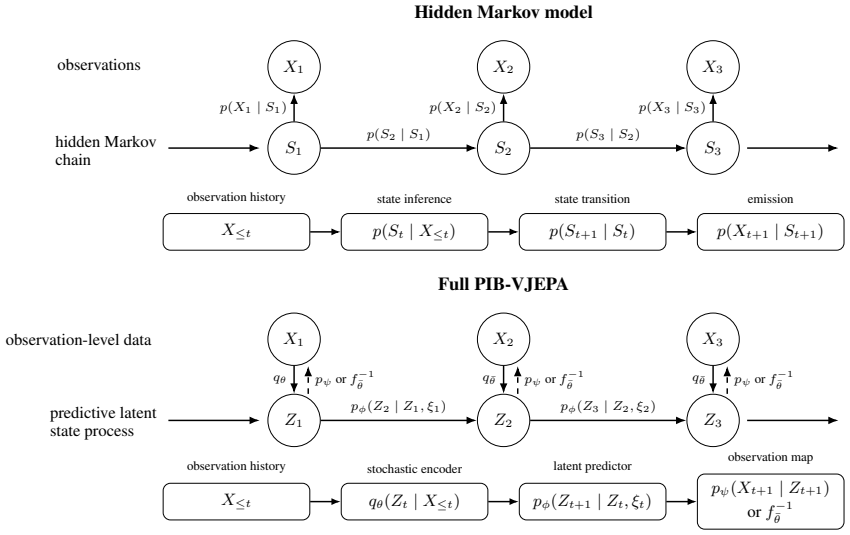


Figure 1: Graphical and computational comparison of an HMM (top) and full PIB-VJEPA (bottom). In PIB-VJEPA, solid arrows from observations to states denote stochastic encoding (the state-inference role), whereas dashed arrows denote the optional state-to-observation map given by an explicit decoder p_ψ or an inverse target encoder $f_{\bar{\theta}}^{-1}$ (the emission role). The boxed rows reproduce the aligned pipelines in Eq. (1).

states: it plays the *filtering* role and is *not* an emission model. The emission direction must instead be supplied by a decoder, by the inverse of an invertible target encoder, or by an induced implicit conditional. Moreover, standard JEPA training matches latent targets and need not optimize any observation likelihood, so sharing HMM components does not imply defining the same sequence distribution, let alone being trained by the same objective.

We therefore make the claim "a probabilistic JEPA is secretly an HMM" precise rather than all-or-nothing. Our main object of study is the full, time-indexed PIB-VJEPA (Huang, 2026b), whose current representation, future target, and transition are all probabilistic; other probabilistic temporal JEPAs inherit the same encode–transition structure, and classical deterministic JEPA is recovered as a degenerate point-mass case. Our contributions are:

1. **Markov-Chain JEPA (MCJEPA).** We replace the latent predictor by a learned row-stochastic transition matrix, so that all horizons are generated by matrix powers and direct and composed multi-step predictions satisfy exact Chapman–Kolmogorov consistency (Section 3).

2. **A hierarchy of latent Markov transitions.** We extend the construction to conditioned discrete transitions, continuous-state kernels, and continuous-time dynamics, with deterministic temporal JEPA as a degenerate Dirac-kernel case (Section 4).

3. **Four levels of HMM correspondence.** We distinguish computational correspondence, emission-complete latent-state representation, sequence-level HMM equivalence, and model-and-objective equivalence; realize the emission direction via explicit, inverse, or implicit constructions; and give sufficient conditions for an exact HMM representation (Section 5).

4. **Information bottleneck as Markovization.** We interpret predictive IB learning as the construction of a compact Markov state, separating *minimality* (promoted by compression) from *sufficiency* (tested by residual predictability) (Section 6).

Four controlled experiments (Section 7) test transition recovery and composition, the filtering interpretation of the context encoder, recovery of a known minimal predictive state, and the objective-level distinction between JEPA latent prediction and HMM-style sequence learning.

2 Background and Related Work

Hidden Markov and predictive-state models. An HMM factorizes the joint distribution of hidden states and observations as (Rabiner, 1989)

$$p(s_{1:T}, x_{1:T}) = p(s_1) \prod_{t=1}^{T-1} p(s_{t+1} \mid s_t) \prod_{t=1}^{T} p(x_t \mid s_t), \tag{2}$$

with side information entering as $p(s_{t+1} \mid s_t, \xi_t)$. Filtering, prediction, and likelihood evaluation follow from the forward recursion. Predictive-state representations (Littman & Sutton, 2001) instead define state through predictions of future observations, a view that our sufficiency condition in Section 6 shares. Deep sequential latent-variable models (**???**) and latent world models (**?**) learn amortized inference networks and neural transitions trained through observation-space ELBOs. JEPA differs in imposing its principal objective in representation space; our analysis clarifies exactly what is lost or retained relative to such likelihood-based state-space models.

Temporal and probabilistic JEPA. A temporal JEPA uses an online encoder, an EMA target encoder, and a latent predictor, $Z_t = f_\theta(X_{\le t})$, $Z^{\mathrm{T}}_{t+h} = f_{\bar\theta}(X_{t+h})$, and $\widehat{Z}_{t+h} = P_\phi(Z_t, h, \xi_{t:t+h})$, and matches $\widehat{Z}_{t+h}$ to the stop-gradient target (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024; Huang, 2026c). Full PIB-VJEPA (Huang, 2026b) contains three conditionals: a current-state encoder $q_\theta(Z_t \mid X_{\le t})$, a latent transition $p_\phi(Z_{t+1} \mid Z_t, \xi_t)$, and a future target encoder $q_{\bar\theta}(Z_{t+1} \mid X_{t+1})$, trained with

$$\begin{aligned}\mathcal{L}_{\text{PIB-VJEPA}} = \mathbb{E}\big[-\log p_\phi(Z_{t+1} \mid Z_t, \xi_t)\big] + \gamma_{\mathrm{S}}\, \mathbb{E}\mathrm{KL}\big(q_\theta(Z_t \mid X_{\le t}) \,\|\, p^{\mathrm{S}}_{\mathrm{ref}}\big) \\ + \beta_{\mathrm{T}}\, \mathbb{E}\mathrm{KL}\big(q_{\bar\theta}(Z_{t+1} \mid X_{t+1}) \,\|\, p^{\mathrm{T}}_{\mathrm{ref}}\big),\end{aligned} \tag{3}$$

where the first term rewards predictability of the future target state and the KL terms compress the source-side (S) and target-side (T) representations towards reference priors. This is a variational implementation (Alemi et al., 2017) of the predictive information bottleneck (Tishby et al., 1999; Bialek et al., 2001; Still et al., 2012).

3 Markov-Chain JEPA

Markov dynamics can be organized by whether time and state are discrete or continuous. We begin with the discrete-time, discrete-state case because it gives the most transparent link to an HMM transition matrix (the full taxonomy is in Section C).

Construction. We specialize the JEPA state to a categorical variable $Z_t \in \{1, \dots, K\}$ (reserving S_t for HMM notation). The online encoder returns a soft state $q_t = q_\theta(Z_t \mid X_{\le t}) \in \Delta^{K-1}$ and the EMA target encoder returns $\bar q_{t+h} = q_{\bar\theta}(Z_{t+h} \mid X_{t+h}) \in \Delta^{K-1}$. The predictor is a time-homogeneous row-stochastic matrix $A \in [0,1]^{K\times K}$ with $A_{ij} = p_\phi(Z_{t+1} = j \mid Z_t = i)$, so that

$$\widehat{q}_{t+1} = q_t A, \qquad \widehat{q}_{t+h} = q_t A^h, \tag{4}$$

i.e., $\widehat{q}_{t+h}$ is a convex combination of the rows of A^h weighted by the current belief. The encoder and transition are trained by matching every predicted distribution to its stop-gradient target,

$$\mathcal{L}_{\mathrm{MC}} = \mathbb{E}\Big[\textstyle\sum_{h\in\mathcal{H}} w_h\, \mathrm{KL}\big(\mathrm{sg}(\bar q_{t+h}) \,\|\, q_t A^h\big)\Big], \tag{5}$$

with horizon weights $w_h \ge 0$ (e.g., uniform or exponentially discounted). As in JEPA, no observation reconstruction is required; a decoder or inverse target encoder may optionally map $q_t A^h$ back to observation space (Fig. 5, Section A).

Exact multi-horizon path consistency. Because all horizons are generated by one matrix, a direct prediction agrees exactly with any prediction composed through intermediate horizons.

Proposition 1 (Exact path consistency)**.** *If the latent dynamics form a time-homogeneous finite-state Markov chain with row-stochastic transition matrix A, then for any state distribution q_t and nonnegative integers h_1, h_2,*

$$q_t A^{h_1+h_2} = \big(q_t A^{h_1}\big) A^{h_2}. \tag{6}$$

Consequently, all prediction paths whose lengths sum to the same total horizon give the same predictive distribution.

This is the Chapman–Kolmogorov law for a homogeneous chain (proof in Section F.1). It lets long-horizon predictions be computed directly or by composing short ones without path-dependent discrepancies, which is useful for hierarchical planning and temporal abstraction. A softer alternative learns a separate A_h per horizon and penalizes $\mathcal{L}_{\mathrm{CK}} = \sum_{h_1,h_2} \|A_{h_1+h_2} - A_{h_1}A_{h_2}\|_F^2$; we use it as a baseline.

Avoiding discrete-state collapse. Because encoder, target, and transition are learned jointly, Eq. (5) admits degenerate agreement: *single-state collapse* ($q_t \approx e_j$ for all t, with $A_{jj} \approx 1$) and *uniform-assignment collapse* ($q_t \approx \mathrm{Unif}(K)$). An occupancy term on the minibatch average $\bar{q}_{\mathcal{B}} = |\mathcal{B}|^{-1}\sum_{t\in\mathcal{B}} q_t$ prevents the former but is blind to the latter, whereas a per-sample entropy term prevents the latter but not the former. We therefore combine them,

$$\mathcal{L}_{\mathrm{state}} = \lambda_{\mathrm{occ}}\,\mathrm{KL}\big(\bar{q}_{\mathcal{B}} \,\|\, \mathrm{Unif}(K)\big) + \lambda_{\mathrm{ent}}\,\tfrac{1}{|\mathcal{B}|}\textstyle\sum_{t\in\mathcal{B}} H(q_t), \tag{7}$$

encouraging diversity across observations and confidence within each observation; the basic MCJEPA objective is $\mathcal{L}_{\mathrm{MC}}+\mathcal{L}_{\mathrm{state}}$. A nonuniform prior π can replace $\mathrm{Unif}(K)$ when unequal state frequencies are expected (Section D).

4 From a Transition Matrix to a Neural Markov Kernel

The Markov principle only requires that the next state depend on the past through the current state and transition-relevant side information,

$$p(Z_{t+1} \mid Z_{\le t}, \xi_{\le t}) = p(Z_{t+1} \mid Z_t, \xi_t), \tag{8}$$

which constrains conditional independence, not the functional form of the predictor. Preserving Eq. (8), the fixed matrix generalizes along a hierarchy (summarized in Table 5, Section C): (i) *conditioned MCJEPA* uses a neural row-stochastic matrix $A_t = A_\phi(\xi_t)$, with $\widehat{q}_{t+h} = q_t \prod_{j=0}^{h-1} A_\phi(\xi_{t+j})$ ordered chronologically, which supports actions, goals, regimes, and irregular intervals; (ii) *continuous-state Markov JEPA* uses a kernel such as $p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \mathcal{N}(\mu_\phi(Z_t,\xi_t), \Sigma_\phi(Z_t,\xi_t))$, whose multi-step law marginalizes intermediate states, the continuous analogue of matrix multiplication, and can use mixtures (Huang, 2026a), flows, or diffusion models when futures are multimodal; (iii) *continuous-time* variants use a generator Q_ϕ with $A_\phi(\Delta t) = \exp(\Delta t\, Q_\phi)$ for discrete states, or a latent SDE $dZ = b_\phi\, d\tau + G_\phi\, dW$ for continuous states. Finally, a *deterministic* temporal JEPA predictor $\widehat{Z}_{t+1} = g_\phi(Z_t, \xi_t)$ is the Dirac kernel

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \delta\big(Z_{t+1} - g_\phi(Z_t, \xi_t)\big), \tag{9}$$

the $\Sigma_\phi \to 0$ limit of the Gaussian kernel. Classical JEPA thus sits at the deterministic boundary of the same state-space family. A recurrent predictor $p_\phi(Z_{t+1} \mid Z_t, M_t, \xi_t)$ with memory M_t is not first-order Markov in Z_t alone, but it becomes so in the augmented state $\widetilde{Z}_t = (Z_t, M_t)$. This motivates the representation-learning goal of Section 6: ideally Z_t *itself* summarizes all predictive history, so that a first-order transition suffices.

5 A Probabilistic JEPA Is Secretly an HMM

Under Eq. (2), observation-level data play the role of HMM observations, Z_t is the hidden state, p_ϕ the transition, and the history-dependent encoder $q_\theta(Z_t \mid X_{\le t})$ plays the state-inference role (a full dictionary is given in Table 4, Section B). With an observation model, PIB-VJEPA induces the one-step predictive distribution

$$p_\Theta(X_{t+1} \mid X_{\le t}) = \iint p_\psi(X_{t+1} \mid Z_{t+1})\, p_\phi(Z_{t+1} \mid Z_t, \xi_t)\, q_\theta(Z_t \mid X_{\le t})\, dZ_t\, dZ_{t+1}, \tag{10}$$

which reduces to HMM prediction when q_θ coincides with the Bayesian filter.

Three realizations of the emission direction. (1) An *explicit decoder* $p_\psi(X_t \mid Z_t)$ is the literal emission analogue. (2) An *inverse target encoder*: if $f_{\bar{\theta}}$ is bijective on the data domain, $\widehat{X}_{t+1} = f_{\bar{\theta}}^{-1}(\widehat{Z}_{t+1})$ is a deterministic (Dirac) emission; a normalized likelihood additionally needs a change-of-variables or noise model, and compressive JEPA encoders are generally not invertible. (3) An

implicit emission: when neither map exists, a local stochastic encoder $q_\theta(z \mid x)$ together with the data marginal still induces one. With $q_\theta(z) = \int p_{\mathrm{data}}(x) q_\theta(z \mid x)\, dx$, define

$$p_\theta^{\mathrm{imp}}(x \mid z) = \frac{p_{\mathrm{data}}(x)\, q_\theta(z \mid x)}{q_\theta(z)} \quad \text{whenever } q_\theta(z) > 0. \tag{11}$$

Proposition 2 (Implicit emission completion)**.** *The conditional in Eq.* (11) *is normalized and satisfies* $q_\theta(z \mid x) = p_\theta^{\mathrm{imp}}(x \mid z)\, q_\theta(z)/p_{\mathrm{data}}(x)$*. Thus every local stochastic encoder, together with the data marginal, defines a one-time latent-variable model for which the encoder is the exact posterior.*

This establishes an exact *static* observation–state correspondence (proof in Section F.2), but the induced conditional need not be tractable for generation or likelihood evaluation, and it does not by itself yield a sequence-level HMM.

Four levels of correspondence. The HMM correspondence strengthens as more state-space semantics are imposed:

1. *Computational correspondence:* both models implement observation-to-state inference, state transition, and state-to-observation prediction; no common distribution or objective is implied.
2. *Emission-complete latent-state representation:* a valid state-to-observation conditional (explicit, inverse, or implicit) together with a valid transition provides the ingredients of a latent Markov model, but the encoder need not be its Bayesian filter.
3. *Sequence-level HMM equivalence:* transition, emission, latent marginals, and history encoder are mutually consistent, so the model admits the HMM factorization (Theorem 1).
4. *Model-and-objective equivalence:* additionally, the HMM-compatible model is trained by its observation-sequence likelihood (or sequence evidence), rather than only by latent prediction.

Theorem 1 (Sufficient conditions for an exact HMM representation)**.** *Consider a probabilistic temporal JEPA with latent state* Z_t *and observations* X_t*. Its latent transition and emission define an HMM sequence model and its history-dependent encoder coincides with the corresponding filtering distribution, provided that: (i) the latent dynamics satisfy the first-order Markov property; (ii) a valid state-to-observation conditional is specified by an explicit decoder, an invertible target encoder, or the implicit construction in Eq.* (11)*; (iii) the latent-state marginals are consistent with the transition kernel; and (iv) the history-dependent encoder coincides with the filtering posterior induced by the transition and emission models. For the implicit emission, the local encoder must additionally satisfy observation-locality and Bayes-consistency conditions. Then the joint sequence distribution admits the factorization in Eq.* (2)*. These conditions are sufficient, not necessary, and are not guaranteed by standard JEPA training.*

The precise marginal- and filtering-consistency conditions and the proof are given in Section I. Level 4 is then a statement about training: standard probabilistic JEPA objectives such as Eq. (3) do not maximize $\log p(X_{1:T})$. HMM-style and hybrid objectives, which add sequence likelihood and filtering supervision, are developed in Section H and tested in Section 7.4.

6 Information Bottleneck Learning as Markovization

Predictive IB learning seeks a representation that compresses the past while retaining information about the future (Tishby et al., 1999; Bialek et al., 2001; Alemi et al., 2017):

$$\min_{q_\theta(Z_t \mid X_{\le t})} \; \mathrm{I}(X_{\le t}; Z_t) - \lambda\, \mathrm{I}(Z_t; Z_{t+1}). \tag{12}$$

Minimizing Eq. (12) alone does not guarantee sufficiency. The ideal target is the predictive-state condition (Littman & Sutton, 2001)

$$X_{>t} \perp X_{\le t} \mid Z_t, \tag{13}$$

and, for one-step latent prediction, the corresponding Markov-sufficiency condition is

$$\mathrm{I}(Z_{t+1}; X_{<t} \mid Z_t) = 0. \tag{14}$$

Table 1: Finite-HMM recovery (mean ± std over 5 seeds). $\mathcal{E}_A$ is the permutation-aligned transition error; $\mathcal{L}_1, \mathcal{L}_8$ are true-state prediction NLL at horizons 1 and 8. Bold: best mean per regime.

Regime	Model	ARI ↑	$\mathcal{E}_A \downarrow$	$\mathcal{L}_1 \downarrow$	$\mathcal{L}_8 \downarrow$
Separated	Gaussian HMM	$\mathbf{0.9947 \pm 0.0006}$	$\mathbf{0.0082 \pm 0.0020}$	$\mathbf{0.5240 \pm 0.0026}$	$\mathbf{1.3027 \pm 0.0050}$
	Horizon-specific A_h	0.9836 ± 0.0034	0.0231 ± 0.0018	0.5433 ± 0.0019	1.3036 ± 0.0058
	MCJEPA shared A	0.9796 ± 0.0055	0.0191 ± 0.0019	0.5438 ± 0.0018	1.3048 ± 0.0049
Ambiguous	Gaussian HMM	$\mathbf{0.7543 \pm 0.0115}$	$\mathbf{0.0091 \pm 0.0027}$	$\mathbf{0.6676 \pm 0.0103}$	$\mathbf{1.3127 \pm 0.0060}$
	Horizon-specific A_h	0.5581 ± 0.0476	0.2410 ± 0.0029	0.8488 ± 0.0306	1.3288 ± 0.0041
	MCJEPA shared A	0.5323 ± 0.0513	0.0901 ± 0.0091	0.8500 ± 0.0349	1.3491 ± 0.0093

Proposition 3 (Predictive sufficiency implies one-step Markov sufficiency)**.** *Assume Eq.* (13) *and that the future target state is generated as* $Z_{t+1} = g_{\bar{\theta}}(X_{t+1}, U_{t+1})$ *with target-encoder randomness satisfying* $U_{t+1} \perp X_{\le t} \mid (X_{t+1}, Z_t)$*. Then* $\mathrm{I}(Z_{t+1}; X_{<t} \mid Z_t) = 0$*.*

The proof uses the conditional data-processing inequality (Section F.3). Thus a predictively sufficient state *Markovizes* the process at the chosen scale: compression promotes *minimality* by discarding redundant history, while sufficiency prevents over-compression. Note that a first-order predictor does not imply a sufficient Markov state: if $p(Z_{t+1} \mid Z_t, Z_{t-1}) \neq p(Z_{t+1} \mid Z_t)$, a first-order transition is being imposed on a representation that has not Markovized the process.

Residual predictability as a diagnostic of sufficiency. Because Eq. (14) is hard to verify directly, we test its consequences. For categorical MCJEPA, the *probability innovation* $R_{t+1} = \mathrm{sg}(\bar{q}_{t+1}) - q_t A_t$ has zero conditional mean given the predictor's own inputs $\mathcal{G}_t = \sigma(q_t, \xi_t)$ whenever the transition attains the optimum of the forward-KL objective. Markov sufficiency further requires $\mathbb{E}[R_{t+1} \mid \mathcal{F}_t] = 0$ for the full history filtration $\mathcal{F}_t = \sigma(X_{\le t}, \xi_{\le t})$, i.e., a martingale-difference property. An operational test compares a restricted residual predictor $g_{\omega_0}(q_t, \xi_t)$ with a history-augmented one $g_{\omega_1}(q_t, \xi_t, q_{t-1}, \ldots, q_{t-K_r})$ on held-out data:

$$\Delta_{\mathrm{hist}} = \widehat{\mathbb{E}}\big\|R_{t+1} - \widehat{R}^{(0)}_{t+1}\big\|_2^2 - \widehat{\mathbb{E}}\big\|R_{t+1} - \widehat{R}^{(1)}_{t+1}\big\|_2^2. \tag{15}$$

A reliably positive Δ_{hist} (under matched capacity and out-of-sample evaluation) is evidence that older history carries transition-relevant information missing from Z_t. The diagnostic is asymmetric: detecting predictability refutes sufficiency, while failing to detect it does not prove the Markov property. Linear covariance tests, the separation of state insufficiency from predictor misspecification, and standardized innovations for continuous kernels are given in Section E.

7 Experiments

Our experiments are deliberately small and diagnostic: rather than targeting state-of-the-art forecasting, they test the structural claims above in synthetic settings where latent states, transitions, filters, and sufficient statistics are known. All learned-model results use five seeds (mean ± std). Categorical states are aligned to the ground truth by a Hungarian assignment; we report the adjusted Rand index (ARI), the permutation-aligned transition error $\mathcal{E}_A = \|P^\top \widehat{A} P - A^\star\|_F / \|A^\star\|_F$, true-state prediction NLL $\mathcal{L}_h$ at horizon $h \in \{1, 2, 4, 8\}$, and, where an explicit emission exists, observation-sequence NLL per step. Full definitions, data-generation processes, architectures, and hyperparameters are in Section J; extended results and discussion are in Section K.

7.1 Experiment 1: finite-HMM recovery and Markov composition

We fit MCJEPA (one shared A, $\widehat{q}_{t+h} = q_t A^h$) to a four-state HMM with Gaussian emissions in a *separated* and an *ambiguous* emission regime, and compare it with a horizon-specific categorical JEPA (an independent A_h per horizon) and a correctly specified Gaussian HMM (Table 1).

Results. In the separated regime all methods recover the process well; MCJEPA reaches ARI 0.9796 ± 0.0055 and $\mathcal{E}_A = 0.0191 \pm 0.0019$, close to the correctly specified HMM. Under ambiguity the HMM remains strongest, as expected from its correct emission family and explicit filtering. The horizon-specific model gains slight predictive flexibility but learns a much less faithful transition law ($\mathcal{E}_A = 0.2410 \pm 0.0029$ vs. 0.0901 ± 0.0091 for MCJEPA), and its horizons do not compose: its path

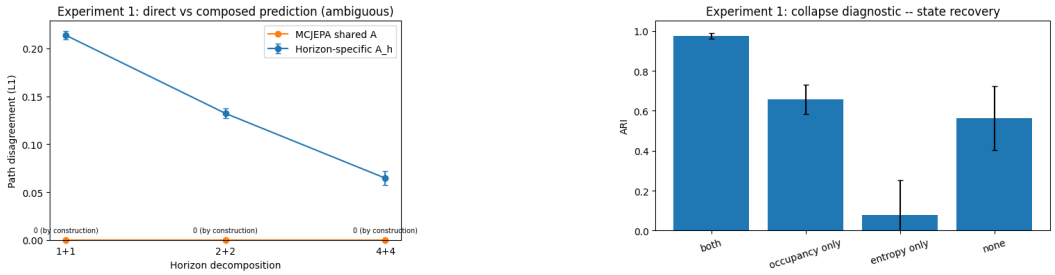


Figure 2: Exp 1. *Left:* direct-vs-composed disagreement $\mathcal{D}_{\text{path}}(h_1, h_2) = \mathbb{E}_t \| q_t A_{h_1+h_2} - (q_t A_{h_1}) A_{h_2} \|_1$ in the ambiguous regime for decompositions $1{+}1$, $2{+}2$, and $4{+}4$; MCJEPA is exactly path-consistent, independently learned A_h are not. *Right:* ARI under ablations of the state-use regularizer $\mathcal{L}_{\text{state}}$; occupancy and entropy terms are complementary. Error bars: $\pm$ 1 std over 5 seeds.

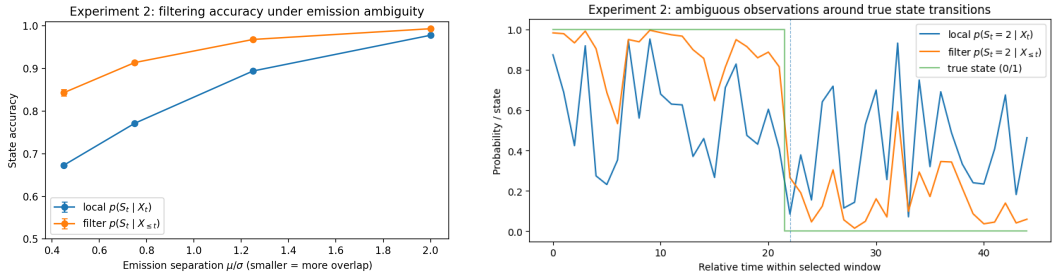


Figure 3: Experiment 2 (oracle posteriors from the known generating process). *Left:* filtering increasingly outperforms local evidence as emission overlap grows. *Right:* around a true state transition, the local posterior reacts to individual noisy observations, whereas the filtered belief integrates temporal evidence through the transition model.

disagreement is 0.2138, 0.1321, and 0.0645 for the $1{+}1$, $2{+}2$, and $4{+}4$ decompositions, whereas MCJEPA's is exactly zero (Fig. 2, left). The benefit of sharing A is structural, not a universally lower NLL. The collapse ablation (Fig. 2, right; Table 11) confirms the complementary roles of the two regularizers: using both gives ARI 0.9748 ± 0.0134 with about four effective states; occupancy alone leaves assignments diffuse (entropy 1.10); and entropy alone collapses four of five runs to a single state ($K_{\text{eff}} = 1.17 \pm 0.38$).

7.2 Experiment 2: Filtering Resolves Emission Ambiguity

To isolate the inference role of the context encoder, we use a persistent two-state HMM with Gaussian emissions at $\pm\mu$ and decrease the separation $\mu/\sigma \in \{2.0, 1.25, 0.75, 0.45\}$. Since the model is known, we compute the exact *local evidence* $p(S_t \mid X_t)$ and *filtering* posterior $p(S_t \mid X_{\leq t})$, the oracle counterparts of a local encoder $q_\theta(Z_t \mid X_t)$ and a history encoder $q_\theta(Z_t \mid X_{\leq t})$. Using oracles removes optimization confounds; this is not an HMM-versus-MCJEPA comparison. As emissions overlap, filtering becomes increasingly more informative (Fig. 3): at $\mu/\sigma = 0.45$, accuracy rises from 0.6730 ± 0.0037 (local) to 0.8433 ± 0.0080 (filtering) and state NLL falls from 0.6022 to 0.3683. Around a true transition, local evidence fluctuates with every observation while the filtered belief stays stable and switches only when accumulated evidence supports it. This supports interpreting a history-dependent JEPA encoder as an amortized filtering belief rather than an emission model.

7.3 Experiment 3: Predictive Compression and Markovization

We construct a binary second-order process with $p(X_{t+1} \mid X_{\leq t}) = p(X_{t+1} \mid X_{t-1}, X_t)$ and give the encoder the three-step history $H_t = (X_{t-2}, X_{t-1}, X_t)$ (8 values). The known minimal predictive state is $Z_t^\star = (X_{t-1}, X_t)$ (4 values): it turns a second-order observation process into a first-order state process, so compression should remove X_{t-2} but not X_{t-1}.

Table 2: Exact representation controls and residual history gain in Experiment 3. The minimal pair removes redundant history while preserving all one-step predictive information; the current observation alone is insufficient, which the residual diagnostic detects.

Representation Z_t	$\mathrm{I}(H_t; Z_t)$	$\mathrm{I}(Z_t; X_{t+1})$	NLL	States	Δ_{hist}
Full history H_t	1.7356	0.2807	0.3978	8	$\approx -3.7 \times 10^{-5}$
Minimal pair (X_{t-1}, X_t)	1.3378	0.2807	0.3978	4	$\approx -6.8 \times 10^{-6}$
Current observation X_t	0.6785	0.0192	0.6593	2	0.1139 ± 0.0026

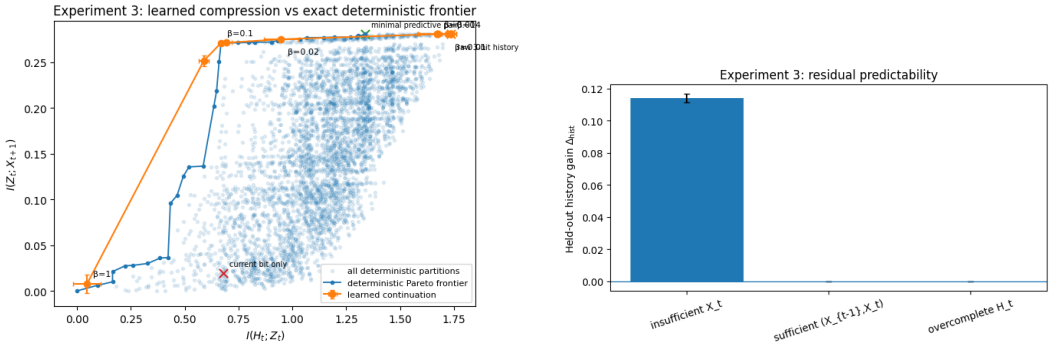


Figure 4: Experiment 3. *Left:* information plane of all 4140 deterministic partitions of the 8 histories (light points) with the exact nondominated frontier (blue), and the warm-started path of a learned stochastic encoder as the compression weight β increases (orange). *Right:* residual history gain Δ_{hist}; adding Z_{t-1} helps only the insufficient representation $Z_t = X_t$.

Compression selects the minimal sufficient state. Exact finite computation (Table 2) shows that the minimal pair stores less history than H_t ($\mathrm{I}(H_t; Z_t)$ of 1.3378 vs. 1.7356) while retaining the same predictive information (0.2807) and NLL (0.3978), whereas $Z_t = X_t$ loses predictive information (0.0192; NLL 0.6593). To check that this is not a favorable hand-picked case, we exhaustively enumerate all 4140 deterministic encoders $Z_t = f(H_t)$ and minimize $\mathcal{L}_{\mathrm{pred}} + \beta\, \mathrm{I}(H_t; Z_t)$. For every tested $0 < \beta \leq 0.014$ the global optimum is exactly $Z_t^\star$; at $\beta = 0$ prediction alone does not identify it (several partitions tie), and at $\beta = 0.015$ the optimum becomes a two-state, over-compressed representation. A learned stochastic encoder, warm-started at the overcomplete representation, traces a similar path from redundant to minimal to over-compressed states as β grows (Fig. 4, left).

Residual predictability detects over-compression. We apply Eq. (15) with one extra step of representation history, comparing held-out predictors of X_{t+1} from Z_t and from (Z_t, Z_{t-1}). For $Z_t = X_t$, $Z_{t-1} = X_{t-1}$ restores exactly the omitted variable and $\Delta_{\mathrm{hist}} = 0.1139 \pm 0.0026$; for the minimal and overcomplete states, Δ_{hist} is negligible (Table 2; Fig. 4, right). The two tools are complementary: the residual test certifies *sufficiency* but cannot tell that the eight-state representation is redundant, while the bottleneck supplies *minimality* by preferring the four-state representation among sufficient ones.

7.4 Experiment 4: HMM-style training of PIB-VJEPA

Experiment 4 probes the strongest level, *model-and-objective equivalence*. On the separated four-state process of Experiment 1 (retrained independently), with a fixed categorical latent family and context-encoder architecture, we compare three regimes. *JEPA-only* optimizes $\mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{state}}$; a Gaussian observation model is fitted post hoc for evaluation only. *HMM sequence + filter distillation* trains the transition p_ϕ and Gaussian emission p_ψ by the exact forward-algorithm likelihood $\mathcal{L}_{\mathrm{HMM}} = -\mathbb{E}[T^{-1} \log p_{\phi,\psi}(X_{1:T})]$ and distils the exact filter $\widetilde{q}_t = p_{\phi,\psi}(Z_t \mid X_{\leq t})$ into the encoder via $\mathcal{L}_{\mathrm{filter}} = \mathbb{E}_t \mathrm{KL}(\mathrm{sg}(\widetilde{q}_t) \,\|\, q_\theta(Z_t \mid X_{\leq t}))$, with *no* $\mathcal{L}_{\mathrm{MC}}$. The *hybrid* optimizes

$$\mathcal{L}_{\mathrm{hybrid}} = \lambda_{\mathrm{seq}} \mathcal{L}_{\mathrm{HMM}} + \lambda_{\mathrm{latent}} \mathcal{L}_{\mathrm{MC}} + \lambda_{\mathrm{filter}} \mathcal{L}_{\mathrm{filter}} + \mathcal{L}_{\mathrm{state}}, \tag{16}$$

where $\mathcal{L}_{\mathrm{MC}}$ is the same EMA-target JEPA loss and the HMM and latent-prediction terms share one transition matrix.

Table 3: Objective-level comparison in Experiment 4 (mean $\pm$ std over 5 seeds). $\mathcal{L}_{\text{state}}$ is shared by all regimes. Bold marks the best mean and does not imply significance. Filter KL is training aligned for the first two regimes and post hoc for JEPA-only.

Training regime (active signals)	ARI ↑	Seq. NLL ↓	$\mathcal{E}_A$ ↓	Filter KL ↓
HMM + filter ($\mathcal{L}_{\text{HMM}}$, $\mathcal{L}_{\text{filter}}$)	$\mathbf{0.9945 \pm 0.0008}$	$\mathbf{1.2659 \pm 0.0072}$	0.0105 ± 0.0014	$\mathbf{0.00006 \pm 0.00001}$
Hybrid ($\mathcal{L}_{\text{HMM}}$, $\mathcal{L}_{\text{MC}}$, $\mathcal{L}_{\text{filter}}$)	0.9905 ± 0.0008	1.2660 ± 0.0073	$\mathbf{0.0103 \pm 0.0017}$	0.00436 ± 0.00102
JEPA-only ($\mathcal{L}_{\text{MC}}$)	0.9802 ± 0.0053	1.2753 ± 0.0069	0.0191 ± 0.0020	0.02546 ± 0.00642

Results. JEPA-only training learns a meaningful state (ARI 0.9802) but, consistent with what it supervises, attains worse sequence NLL (1.2753) and transition error (0.0191) (Table 3). Adding HMM-style supervision to the same MCJEPA loss raises ARI to 0.9905, reduces $\mathcal{E}_A$ by about 46% to 0.0103, and closes about 98.6% of the JEPA-to-HMM sequence-NLL gap (a residual difference of 1.34×10^{-4} nats per step versus 9.41×10^{-3} for JEPA-only); the hybrid improves over JEPA-only on every reported metric for every seed. Conversely, the HMM+filter regime, with no $\mathcal{L}_{\text{MC}}$ at all, achieves the best state recovery and sequence NLL, so JEPA latent prediction is not required to train this architecture. Its near-zero filter KL is expected because it is trained for it; the ordering HMM+filter $<$ hybrid $<$ JEPA-only nonetheless shows how strongly each objective induces the filtering role. The same latent-state family therefore supports a continuum $\mathcal{L}_{\text{MC}} \leftrightarrow \mathcal{L}_{\text{MC}} + \mathcal{L}_{\text{HMM}} + \mathcal{L}_{\text{filter}} \leftrightarrow \mathcal{L}_{\text{HMM}} + \mathcal{L}_{\text{filter}}$, and the degree of HMM equivalence is set by the objective, not by the architecture alone. As the regimes are not compute-matched (mini-batch vs. full-data updates), we read this as an objective-behavior diagnostic (Section J.6).

8 Discussion and Limitations

Beyond MCJEPA. Markovianity is a conditional-independence property, not a restriction on functional form: any neural predictor $p_\phi(Z_{t+1} \mid Z_t, \xi_t)$ (MLP, Transformer, mixture density) is first-order Markov in Z_t. Hence MCJEPA $\subset$ Markov neural-predictor JEPA $\subset$ general temporal JEPA: MCJEPA exhibits the correspondence in its most explicit form but does not create it. Predictors with memory or higher order regain the interpretation through state augmentation.

Why reconstruction can stay optional. An HMM models $p(X_t \mid S_t)$ because its likelihood lives in observation space; JEPA can instead avoid reconstructing high-entropy nuisance detail, adding a decoder only when forecasting or likelihood is needed. Invertible target encoders conflict with compression, and implicit emissions may be intractable.

Limitations. Finite categorical states aid interpretability but do not by themselves yield semantically meaningful labels. Our theory gives sufficient rather than necessary conditions, and standard JEPA training does not enforce filtering or marginal consistency. Also, our experiments are intentionally small and synthetic, to isolate composition, filtering, Markov-state construction, and objective-level behavior under known dynamics; extending these diagnostics to high-dimensional video, control, and real-world partially observed systems could be important next steps.

9 Conclusion

We gave probabilistic temporal JEPA a state-space interpretation, made concrete by Markov-Chain JEPA with exact Chapman–Kolmogorov consistency and extended to conditioned, continuous-state, continuous-time, and Dirac-kernel dynamics. We distinguished four levels of HMM correspondence with sufficient conditions for an exact HMM representation, and recast predictive IB learning as Markov-state construction (compression for minimality, residual predictability for sufficiency). Controlled experiments support each element. Probabilistic temporal JEPA is thus not simply an HMM under another name: it exposes an HMM-compatible structure whose semantics strengthen as emission completeness, consistency, and sequence-level training are imposed.

AI use statement

AI tools were used both to aid and polish writing and for research ideation and execution. The central idea, namely the correspondence between probabilistic temporal JEPA and hidden Markov models,

was originally proposed by the authors, and the AI tools contributed to its subsequent development and elaboration. The authors provided the originating insight, research direction, judgement, and verification. All contents of the paper were manually reviewed and verified by the authors, who take full responsibility for the final content of this work.

Ethics statement

This work is primarily theoretical and uses data generated from known stochastic processes; no human subjects, personal data, or sensitive datasets are involved. We are not aware of direct negative societal impacts specific to this work beyond those of general-purpose representation learning and sequence modeling.

Reproducibility statement

Complete statements of all assumptions and full proofs are provided for Proposition 1 (Section F.1), Proposition 2 (Section F.2), Proposition 3 (Section F.3), and Theorem 1 (Section I). The training objectives and a minimal training algorithm are given in Sections A and H. All data-generating processes, evaluation metrics, architectures, optimization settings, hyperparameters, and random seeds for the four experiments are specified in Section J, including Section J.6. Code implementing all experiments will be released upon publication.

References

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=HyxQzBceg.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture, 2023. URL https://arxiv.org/abs/2301.08243.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido 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=QaCCuDfBk2. Featured Certification.

William Bialek, Ilya Nemenman, and Naftali Tishby. Predictability, complexity, and learning. *Neural Comput.*, 13(11):2409–2463, November 2001. ISSN 0899-7667. doi: 10.1162/089976601753195969. URL https://doi.org/10.1162/089976601753195969.

Yongchao Huang. Gaussian joint embeddings for self-supervised representation learning, 2026a. URL https://arxiv.org/abs/2603.26799.

Yongchao Huang. On the information bottleneck of vjepa. https://hal.science/hal-05622405, 2026b. HAL preprint, hal-05622405.

Yongchao Huang. VJEPA: Variational joint embedding predictive architectures as probabilistic world models. In *Forty-third International Conference on Machine Learning*, 2026c. URL https://openreview.net/forum?id=omJqMJt2fC.

Yann LeCun. A path towards autonomous machine intelligence version 0.9.2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

Michael Littman and Richard S Sutton. Predictive representations of state. In T. Dietterich, S. Becker, and Z. Ghahramani (eds.), *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001. URL https://proceedings.neurips.cc/paper_files/paper/2001/file/1e4d36177d71bbb3558e43af9577d70e-Paper.pdf.

L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. doi: 10.1109/5.18626.

Susanne Still, David A. Sivak, Anthony J. Bell, and Gavin E. Crooks. Thermodynamics of prediction. *Phys. Rev. Lett.*, 109:120604, Sep 2012. doi: 10.1103/PhysRevLett.109.120604. URL https://link.aps.org/doi/10.1103/PhysRevLett.109.120604.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control, and Computing*, 1999.

A Training Objectives and Minimal Algorithm

For the basic time-homogeneous MCJEPA model with one shared transition matrix A, the training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{state}},$$

where $\mathcal{L}_{\mathrm{MC}}$ is the multi-horizon latent-prediction objective in Eq. (5) and $\mathcal{L}_{\mathrm{state}}$ is the discrete-state regularizer in Eq. (7). Because all h-step predictions are generated by powers of the same matrix A, exact path consistency follows automatically from Proposition 1; no additional Chapman–Kolmogorov penalty is required.

A softer alternative may instead parameterize separate horizon-dependent transition matrices A_h. In that case, path consistency is no longer guaranteed and may be encouraged through

$$\mathcal{L}_{\mathrm{CK}} = \sum_{h_1,h_2} \|A_{h_1+h_2} - A_{h_1}A_{h_2}\|_F^2 ,$$

with an additional weight $\lambda_{\mathrm{CK}} \geq 0$. This penalty belongs only to the horizon-dependent variant and is unnecessary for the shared-A MCJEPA used in the main experiments.

For the basic shared-A model, a minimal training step is:

1. sample a time index t, prediction horizon $h \in \mathcal{H}$, observation history $X_{\leq t}$, future target X_{t+h}, and any required side information;
2. compute the current-state distribution
$$q_t = q_\theta(Z_t \mid X_{\leq t});$$
3. compute the EMA target distribution
$$\bar{q}_{t+h} = q_{\bar{\theta}}(Z_{t+h} \mid X_{t+h});$$
4. propagate the current state through the shared transition matrix,
$$\widehat{q}_{t+h} = q_t A^h;$$
5. evaluate the latent-prediction loss in Eq. (5) using $\mathrm{sg}(\bar{q}_{t+h})$ as the target, and add the state-use regularizer $\mathcal{L}_{\mathrm{state}}$;
6. update the online encoder parameters θ and the learnable transition parameters, then update the target encoder by exponential moving average,
$$\bar{\theta} \leftarrow \tau\bar{\theta} + (1-\tau)\theta, \qquad \tau \in [0,1).$$

For the conditioned discrete-state model in Eq. (18), Step 4 is replaced by the ordered transition composition

$$\widehat{q}_{t+h} = q_t \prod_{j=0}^{h-1} A_\phi(\xi_{t+j}),$$

as defined in Eq. (19). The remainder of the training procedure is unchanged.

The residual quantities introduced in Section 6 are used as held-out diagnostics of state and transition sufficiency rather than as part of the default MCJEPA training objective. In particular, residual-history gain is evaluated after fitting the representation and transition model so that residual predictability can diagnose information omitted from the current state without directly training the representation to satisfy the diagnostic.

Table 4: Direct correspondence between an HMM and full PIB-VJEPA. Observation-level inputs correspond to HMM observations, the stochastic context encoder plays the hidden-state inference role, the predictor defines the latent transition, and an explicit decoder, inverse target encoder, or induced implicit conditional can supply the emission direction.

Role	Hidden Markov model	Full PIB-VJEPA
Observed variable	Observation X_t generated by an emission model	Observation-level input X_t, such as a frame, pixel array, or time-series measurement
State inference	Filtering distribution $p(S_t \mid X_{\leq t})$	Stochastic context encoder $q_\theta(Z_t \mid X_{\leq t})$
Hidden state	Latent state S_t	Predictive latent representation Z_t
Future-state reference	Posterior or inferred belief over S_{t+1}	EMA target encoder $q_{\bar{\theta}}(Z_{t+1} \mid X_{t+1})$
State transition	$p(S_{t+1} \mid S_t)$ or transition matrix A	Probabilistic predictor $p_\phi(Z_{t+1} \mid Z_t, \xi_t)$
Emission direction	$p(X_{t+1} \mid S_{t+1})$ or emission matrix B	Decoder $p_\psi(X_{t+1} \mid Z_{t+1})$, inverse target encoder $f_{\bar{\theta}}^{-1}$, or induced implicit emission
Training signal	Observation likelihood or sequence ELBO	Latent target matching plus information-bottleneck regularization; observation decoding is optional

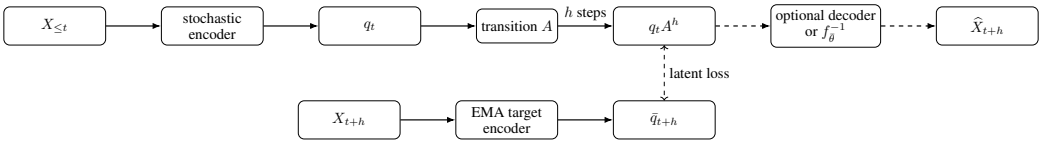


Figure 5: Markov-Chain JEPA. The context encoder infers a predictive-state distribution, the transition matrix propagates it to future latent states, and training matches the prediction to an EMA target distribution. The dashed observation branch is optional: a decoder or inverse target encoder can map the predicted latent state back to observation space when observation forecasting or an explicit emission model is required.

B HMM–PIB-VJEPA Dictionary and the MCJEPA Pipeline

The HMM correspondence is easiest to understand through the common encode–transition–emit pipeline summarized in Table 4. In both an HMM and full PIB-VJEPA, observations are used to infer a belief over a latent state, the state is advanced by a transition model, and the predicted state may be mapped back to observation space. Compared with a classical HMM, PIB-VJEPA primarily changes how these roles are parameterized and trained: state inference is amortized by an encoder, future-state supervision is supplied by a target encoder, and the principal predictive objective is imposed in representation space rather than through an observation-sequence likelihood.

The target branch remains a slowly moving representation target. The core JEPA objective does not require observation reconstruction, although an explicit decoder or inverse target-encoder map may be used to map the predicted state back to X_{t+h} for observation forecasting or HMM-style likelihood modeling.

As an alternative to introducing a separate decoder, PIB-VJEPA may use the inverse target-encoder construction $\widehat{X}_{t+1} = f_{\bar{\theta}}^{-1}(\widehat{Z}_{t+1})$. If

$$f_{\bar{\theta}} : \mathcal{X} \to \mathcal{S}$$

is bijective on the modeled data domain, then

$$\widehat{X}_{t+1} = f_{\bar{\theta}}^{-1}(\widehat{Z}_{t+1})$$

provides a deterministic state-to-observation map without requiring a separate decoder. A normalized emission likelihood additionally requires a tractable density, for example through a change-of-variables model or an explicit observation-noise distribution. Compatible dimensions and invertibility are strong conditions that ordinary compressed JEPA encoders generally do not satisfy, so we treat

Table 5: Hierarchy of latent Markov transitions for temporal JEPA.

Model	Time	State	Transition
MCJEPA	Discrete	Discrete	Fixed matrix A
Conditioned MCJEPA	Discrete	Discrete	$A_\phi(\xi_t)$
Continuous-state Markov JEPA	Discrete	Continuous	Kernel $p_\phi(Z_{t+1} \mid Z_t, \xi_t)$
Deterministic temporal JEPA	Discrete	Typically continuous	Dirac kernel at $g_\phi(Z_t, \xi_t)$
Continuous-time MCJEPA	Continuous	Discrete	Generator Q_ϕ, $\exp(\Delta t\, Q_\phi)$
Continuous-time stochastic JEPA	Continuous	Continuous	Latent SDE

inverse target encoding as an alternative realization of the emission direction rather than a universal requirement of PIB-VJEPA.

C From a Transition Matrix to a Neural Markov Kernel: Details

C.1 A taxonomy of latent Markov dynamics

It is useful to locate Markov-Chain JEPA within the broader family of Markov models. Markov dynamics can be organized along two independent axes: whether time is discrete or continuous, and whether the state space is discrete or continuous. These choices give four common cases:

1. *Discrete time and discrete state:* the transition law is represented by a row-stochastic transition matrix. This is the setting adopted by the basic MCJEPA construction developed below.
2. *Discrete time and continuous state:* the transition law is represented by a conditional probability density or, more generally, a Markov kernel over continuous latent representations.
3. *Continuous time and discrete state:* the dynamics form a continuous-time Markov chain specified by a transition-rate generator.
4. *Continuous time and continuous state:* the dynamics may be represented by a stochastic differential equation or another continuous-time Markov process.

We use *latent Markov dynamics* as an umbrella term for these 4 cases. More specifically, *Markov chain* commonly refers to a discrete-state process, whereas *Markov process* or *Markov kernel* also covers continuous-state models. Our main development focuses on discrete-time prediction because temporal JEPA training is typically organized around frames, tokens, or measurements indexed by discrete steps. We begin with the discrete-time, discrete-state case because it yields the most transparent connection to a classical HMM transition matrix. We then relax the fixed-matrix and discrete-state assumptions using conditioned transition matrices and continuous-state kernels. Continuous-time variants are included later to situate the framework more broadly and to accommodate irregularly sampled systems.

The taxonomy in Section C.1 places the basic MCJEPA construction in the *discrete-time, discrete-state* class. Its transition matrix is the simplest realization of latent Markov dynamics: the predictive state is categorical, time advances in discrete steps, and, in the time-homogeneous case, the same transition matrix is applied at every step. As shown in Eq. (4), this gives the particularly simple multi-step prediction

$$\widehat{q}_{t+h} = q_t A^h.$$

The Markov principle itself is more general. It requires that the next-state distribution depend on the past only through the current predictive state and the transition-relevant side information[1]:

$$p(Z_{t+1} \mid Z_{\le t}, \xi_{\le t}) = p(Z_{t+1} \mid Z_t, \xi_t). \tag{17}$$

The transition may therefore be fixed or conditioned, and the latent state may be discrete or continuous. We now generalize the fixed transition matrix progressively while preserving this conditional-independence structure.

[1] See sufficiency of predictive state (Huang, 2026c).

C.2 NEURAL DISCRETE-STATE TRANSITIONS

The fixed matrix A assumes that the transition law is time homogeneous and independent of external information. A more expressive model allows a neural network to produce a row-stochastic transition matrix conditioned on side information:

$$A_t = A_\phi(\xi_t), \qquad A_t \in [0,1]^{K\times K}, \qquad \sum_{j=1}^{K}(A_t)_{ij} = 1. \tag{18}$$

The side information ξ_t may contain an action, elapsed time, goal, control input, regime indicator, or exogenous covariates. Its entries have the interpretation

$$(A_t)_{ij} = p_\phi(Z_{t+1} = j \mid Z_t = i, \xi_t).$$

For a sequence of conditioned transitions, the h-step predictive distribution is

$$\begin{aligned}\widehat{q}_{t+h} &= q_t \prod_{j=0}^{h-1} A_\phi(\xi_{t+j}) \\ &= q_t A_\phi(\xi_t) A_\phi(\xi_{t+1}) \cdots A_\phi(\xi_{t+h-1}),\end{aligned} \tag{19}$$

where the product is ordered chronologically from left to right. The transition matrices at different steps need not commute, so this ordering matters. In the time-homogeneous, unconditioned case,

$$A_\phi(\xi_{t+j}) = A \qquad \text{for all } j,$$

and Eq. (19) reduces to the $q_t A^h$ construction in Eq. (4).

This construction supports action-conditioned dynamics, changing goals or regimes, exogenous covariates, and irregular temporal intervals when elapsed time is included in ξ_t, while retaining a discrete and interpretable latent state space. It also preserves exact path composition whenever the same chronologically ordered transition sequence is used along both prediction paths.

C.3 CONTINUOUS-STATE STOCHASTIC TRANSITIONS

A categorical state may be too restrictive when the predictive representation varies continuously. In this case, the transition matrix is replaced by a Markov kernel

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t),$$

which assigns a probability distribution over the next continuous latent state for each current state and side-information value.

A simple example is a Gaussian transition,

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \mathcal{N}(Z_{t+1}; \mu_\phi(Z_t, \xi_t), \Sigma_\phi(Z_t, \xi_t)), \tag{20}$$

where the neural network outputs a conditional mean μ_ϕ and a valid covariance matrix Σ_ϕ. The mean describes the expected latent evolution, while the covariance represents stochastic uncertainty and unresolved variation around that mean.

The two-step transition is obtained by marginalizing over the intermediate latent state:

$$\begin{aligned}&p_\phi^{(2)}(Z_{t+2} \mid Z_t, \xi_{t:t+1}) \\ &\quad = \int p_\phi(Z_{t+2} \mid Z_{t+1}, \xi_{t+1}) p_\phi(Z_{t+1} \mid Z_t, \xi_t)\, dZ_{t+1}.\end{aligned}$$

More generally,

$$\begin{aligned}&p_\phi^{(h)}(Z_{t+h} \mid Z_t, \xi_{t:t+h-1}) \\ &\quad = \int \prod_{j=0}^{h-1} p_\phi(Z_{t+j+1} \mid Z_{t+j}, \xi_{t+j})\, dZ_{t+1} \cdots dZ_{t+h-1}.\end{aligned} \tag{21}$$

This is the continuous-state analogue of multiplying transition matrices in Eqs. (4) and (19): all possible intermediate latent states are marginalized out. Closed-form composition is available only for restricted transition families, so neural models may instead use sampling, Monte Carlo integration, moment propagation, or learned approximations to multi-step prediction.

A single Gaussian kernel cannot represent a genuinely multimodal conditional distribution. When multiple distinct futures are important, the same construction can instead use mixtures (Huang, 2026a), normalizing flows, diffusion-based transitions, or other expressive conditional distributions. The central requirement is not Gaussianity, but the Markov factorization in Eq. (8).

C.4 Continuous-time extensions

Suppose that observations are recorded at physical times

$$\tau_1 < \tau_2 < \cdots,$$

with elapsed interval

$$\Delta t_t = \tau_{t+1} - \tau_t.$$

Irregular sampling can already be handled within a discrete-time model by including the observed interval in the side information,

$$\xi_t = (\widetilde{\xi}_t, \Delta t_t),$$

where $\widetilde{\xi}_t$ contains the remaining actions, controls, goals, or exogenous covariates. The resulting discrete-time transition model then learns how state evolution changes with the supplied propagation interval.

A more explicit alternative is to model the latent dynamics directly in continuous time. For a discrete latent state, a continuous-time Markov chain is specified by a generator matrix

$$Q_\phi(\widetilde{\xi}_t),$$

whose off-diagonal entries are nonnegative transition rates and whose rows sum to zero. If the generator is held fixed over a propagation interval of duration $\Delta t \geq 0$, the corresponding transition matrix is

$$A_\phi(\Delta t, \widetilde{\xi}_t) = \exp\Big(\Delta t\, Q_\phi(\widetilde{\xi}_t)\Big). \tag{22}$$

Here, Δt denotes a generic continuous-time propagation duration, whereas $\Delta t_t = \tau_{t+1} - \tau_t$ denotes the particular interval between the tth and $(t+1)$th observations. Under a piecewise-constant conditioning assumption over this interval,

$$A_t = A_\phi(\Delta t_t, \widetilde{\xi}_t) = \exp\Big(\Delta t_t\, Q_\phi(\widetilde{\xi}_t)\Big).$$

The same generator can therefore propagate the latent state across observation intervals of different lengths or across arbitrary forecasting horizons. If the conditioning variables vary continuously within an interval, the corresponding transition requires the appropriate time-varying generator composition rather than a single matrix exponential.

For a continuous latent state, a continuous-time stochastic transition may instead be represented by a stochastic differential equation. Using τ for continuous physical time to distinguish it from the discrete observation index,

$$dZ(\tau) = b_\phi(Z(\tau), \xi(\tau))\, d\tau + G_\phi(Z(\tau), \xi(\tau))\, dW(\tau), \tag{23}$$

where b_ϕ is the drift, G_ϕ controls the diffusion, and $W(\tau)$ is a Wiener process. The drift describes systematic latent evolution, while the diffusion represents stochastic transition uncertainty. Setting $G_\phi = 0$ recovers deterministic continuous-time dynamics such as a neural ordinary differential equation.

These continuous-time constructions are not required for our main MCJEPA development, but they show that the transition-matrix formulation belongs to a broader family of latent Markov models.

C.5 Deterministic temporal JEPA as a degenerate kernel

A deterministic temporal JEPA uses a predictor

$$\widehat{Z}_{t+1} = g_\phi(Z_t, \xi_t).$$

Probabilistically, this is the Dirac transition kernel

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \delta(Z_{t+1} - g_\phi(Z_t, \xi_t)). \tag{24}$$

All conditional probability mass is concentrated at the predictor output, so the transition contains no intrinsic stochastic uncertainty. For example, if the Gaussian mean in Eq. (20) satisfies

$$\mu_\phi(Z_t, \xi_t) = g_\phi(Z_t, \xi_t),$$

then the Gaussian transition approaches this deterministic case as

$$\Sigma_\phi(Z_t, \xi_t) \to 0.$$

Classical deterministic temporal JEPA is therefore not separate from the latent Markov-kernel perspective. A deterministic encoder may be represented probabilistically by a point-mass state distribution, while a deterministic predictor is represented by the Dirac transition in Eq. (9). MCJEPA, conditioned discrete-state JEPA, continuous probabilistic VJEPA, and deterministic JEPA can thus be viewed within the same state-space framework, differing primarily in their state representation and transition family.

C.6 When a recurrent predictor is Markov

A recurrent predictor may appear to violate the first-order Markov assumption because its prediction can depend on a summary of the entire preceding latent history. Let M_t denote a recurrent memory state and suppose that the next latent state is predicted according to

$$p_\phi(Z_{t+1} \mid Z_t, M_t, \xi_t). \tag{25}$$

After sampling or predicting Z_{t+1}, the recurrent memory may be updated deterministically as

$$M_{t+1} = r_\phi(M_t, Z_{t+1}, \xi_t).$$

The process need not be first-order Markov in Z_t alone, because two histories yielding the same Z_t but different memories M_t may induce different next-state distributions. However, defining the augmented state

$$\widetilde{Z}_t = (Z_t, M_t)$$

restores a first-order representation:

$$p_\phi(\widetilde{Z}_{t+1} \mid \widetilde{Z}_{\le t}, \xi_{\le t}) = p_\phi(\widetilde{Z}_{t+1} \mid \widetilde{Z}_t, \xi_t).$$

This distinction motivates a central representation-learning objective of PIB-VJEPA: ideally, the learned predictive state Z_t itself should summarize the information from the observation history that is relevant to future prediction. When this succeeds, a simple first-order transition in Z_t is sufficient. When substantial predictive information remains outside Z_t, the model must either enlarge the state, augment it with memory, use higher-order dynamics, or accept that the latent process is not first-order Markov in the chosen representation.

D Avoiding Discrete-State Collapse: Details

The Markov-chain prediction objective in Eq. (5) does not by itself guarantee that the categorical latent states contain useful information. Because the online encoder, target encoder, and transition matrix are learned jointly, they may agree through a degenerate representation. This is the discrete-state analogue of representation collapse in deterministic self-supervised learning.

For each time index t, the online encoder produces

$$q_t = \big(q_{t1}, \ldots, q_{tK}\big), \qquad q_{tk} = q_\theta(Z_t = k \mid X_{\le t}),$$

where q_{tk} is the probability that the current observation history is represented by latent state k.

Two simple degenerate solutions are particularly important. First, all observations may be assigned to the same state:

$$q_t \approx e_j \qquad \text{for every } t,$$

where $e_j \in \{0,1\}^K$ is the jth standard basis vector, with $(e_j)_j = 1$ and $(e_j)_k = 0$ for all $k \neq j$. The transition matrix can then place nearly all probability on the self-transition A_{jj}, allowing the online and target branches to agree without learning meaningful temporal structure. We refer to this failure mode as *single-state collapse*.

Second, every observation may receive the same uniform assignment:

$$q_t \approx \mathrm{Unif}(K) = \left(\frac{1}{K}, \ldots, \frac{1}{K}\right).$$

A transition matrix that preserves the uniform distribution can again produce consistent predictions even though the latent state contains no information about the observation. We refer to this as *uniform-assignment collapse*.

To measure how the available states are used across a minibatch $\mathcal{B}$, define the average state occupancy

$$\bar{q}_{\mathcal{B}} = \frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} q_t.$$

We encourage aggregate use of the state space through

$$\mathrm{KL}\left(\bar{q}_{\mathcal{B}} \,\|\, \mathrm{Unif}(K)\right).$$

This term is zero when aggregate occupancy is uniform and increases when most probability mass is concentrated on only a few states, thereby discouraging single-state collapse and unused states.

Aggregate diversity alone is not sufficient. If every observation receives the uniform distribution, then

$$\bar{q}_{\mathcal{B}} = \mathrm{Unif}(K),$$

so the occupancy penalty is again zero. We therefore also control the entropy of each assignment,

$$H(q_t) = -\sum_{k=1}^{K} q_{tk} \log q_{tk}.$$

Entropy is maximal at $\log K$ for a uniform assignment and minimal at zero for a one-hot assignment. Because this entropy appears with a positive weight in a minimized loss, it encourages comparatively confident, low-entropy assignments.

Combining the two effects gives

$$\mathcal{L}_{\mathrm{state}} = \underbrace{\lambda_{\mathrm{occ}} \mathrm{KL}\left(\bar{q}_{\mathcal{B}} \,\|\, \mathrm{Unif}(K)\right)}_{\text{occupancy}} + \underbrace{\lambda_{\mathrm{ent}} \frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} H(q_t)}_{\text{entropy}}, \tag{26}$$

where $\lambda_{\mathrm{occ}}, \lambda_{\mathrm{ent}} \geq 0$. The complete basic MCJEPA objective is therefore

$$\mathcal{L} = \mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{state}}.$$

The two regularizers in $\mathcal{L}_{\mathrm{state}}$ play complementary roles. The occupancy term encourages *diversity across observations*, while the entropy term encourages *confidence within each observation*. Their combination therefore favors balanced but informative assignments: different observations can occupy different states while each individual observation receives a comparatively concentrated state distribution.

The relative weights must nevertheless be selected carefully. If λ_{occ} is too large, the model may artificially force every minibatch to use all states even when the underlying state distribution is imbalanced. If λ_{ent} is too large, the encoder may make prematurely hard and unstable assignments. In practice, the entropy weight may be introduced gradually, and the uniform occupancy target may be replaced by a nonuniform prior π when unequal state frequencies are expected:

$$\mathrm{KL}\left(\bar{q}_{\mathcal{B}} \,\|\, \pi\right).$$

E Residual Predictability Diagnostics: Details

The Markov-sufficiency condition in Eq. (14) is difficult to verify directly in a learned, high-dimensional representation. A more operational approach is to ask whether prediction errors retain systematic dependence on information preceding the current state. If older observations or latent states improve prediction after the current representation and transition-relevant side information have been accounted for, then the current state–predictor pair has not captured all transition-relevant information.

Residual diagnostics test consequences of Markov sufficiency rather than the full conditional-independence condition itself. In particular, the diagnostics below focus primarily on conditional-mean predictability. Detecting residual predictability from older history therefore provides evidence against Markov sufficiency, whereas failing to detect it does not prove the full Markov property.

E.1 Categorical probability innovations

For categorical MCJEPA, let

$$q_t = q_\theta(Z_t \mid X_{\le t})$$

denote the current-state distribution, and let

$$\bar{q}_{t+1} = q_{\bar{\theta}}(Z_{t+1} \mid X_{t+1})$$

denote the target-encoder distribution at the next time step. Write $A_t = A$ for the time-homogeneous model and $A_t = A_\phi(\xi_t)$ for an input-conditioned transition. The predicted next-state distribution is

$$\widehat{q}_{t+1} = q_t A_t.$$

We define the categorical probability innovation as

$$R_{t+1} = \mathrm{sg}(\bar{q}_{t+1}) - \widehat{q}_{t+1} = \mathrm{sg}(\bar{q}_{t+1}) - q_t A_t. \tag{27}$$

This vector measures the discrepancy between the target-encoder distribution and the transition-based prediction for each latent category. The stop-gradient operator ensures that, when auxiliary diagnostic models are fitted to these residuals, their gradients are not propagated into the target encoder.

To distinguish transition fitting from state sufficiency, define the current predictor information

$$\mathcal{G}_t = \sigma(q_t, \xi_t)$$

and the full observed-history filtration

$$\mathcal{F}_t = \sigma(X_{\le t}, \xi_{\le t}).$$

For fixed model parameters, $\mathcal{G}_t \subseteq \mathcal{F}_t$, since q_t is computed from the observation history. The transition predictor is conditionally mean-correct with respect to its own inputs when

$$q_t A_t = \mathbb{E}\left[\mathrm{sg}(\bar{q}_{t+1}) \mid \mathcal{G}_t\right].$$

This property is naturally associated with the forward-KL objective used by MCJEPA.[2] Under this condition,

$$\mathbb{E}[R_{t+1} \mid \mathcal{G}_t] = 0.$$

Thus, after conditioning on the inputs already available to the transition predictor, the residual has no remaining predictable conditional mean.

[2] Let $Y = \mathrm{sg}(\bar{q}_{t+1}) \in \Delta^{K-1}$ and condition on $\mathcal{G}_t$. For any predicted distribution $p \in \Delta^{K-1}$,

$$\mathbb{E}[\mathrm{KL}(Y\|p) \mid \mathcal{G}_t] = \mathbb{E}\left[\sum_j Y_j \log Y_j \,\middle|\, \mathcal{G}_t\right] - \sum_j \mathbb{E}[Y_j \mid \mathcal{G}_t] \log p_j.$$

The first term is independent of p, so minimizing the conditional expected KL is equivalent to minimizing the cross-entropy with $m_t = \mathbb{E}[Y \mid \mathcal{G}_t]$. The population optimum is therefore $p^\star = m_t$. Hence, when the transition family can represent the optimum of the MCJEPA objective, $q_t A_t = \mathbb{E}[\mathrm{sg}(\bar{q}_{t+1}) \mid \mathcal{G}_t]$. A restricted or imperfectly optimized transition family need not satisfy this equality exactly.

Markov sufficiency requires a stronger invariance: older history should not alter the conditional prediction once the current predictive state and side information are known. At the level of the target-encoder distribution, the corresponding conditional-mean implication is

$$\mathbb{E}\left[\mathrm{sg}(\bar{q}_{t+1}) \mid \mathcal{F}_t\right] = \mathbb{E}\left[\mathrm{sg}(\bar{q}_{t+1}) \mid \mathcal{G}_t\right].$$

Combining this condition with a conditionally mean-correct transition gives

$$\mathbb{E}[R_{t+1} \mid \mathcal{F}_t] = 0. \tag{28}$$

Assuming integrability, Eq. (28) gives the martingale-difference property of $\{R_{t+1}\}$ with respect to $\{\mathcal{F}_t\}$: once the current state and the available history are known, the residual has no systematic predictable component.

A consequence of Eq. (28) is that R_{t+1} is uncorrelated with any square-integrable function measurable with respect to $\mathcal{F}_t$. A simple linear diagnostic is therefore

$$\mathcal{D}_{\mathrm{lin}} = \sum_{k=1}^{K_{\mathrm{r}}} \left\|\mathrm{Cov}\left(R_{t+1}, q_{t-k}\right)\right\|_F^2,$$

where K_{r} is the maximum lag examined and $\|\cdot\|_F$ denotes the Frobenius norm. A large value indicates that some components of the residual remain linearly associated with earlier latent states. A value near zero rules out only this particular form of linear dependence and does not establish Markov sufficiency.

E.2 Testing incremental predictability from older history

The covariance diagnostic detects only linear dependence. A stronger test asks whether an auxiliary model can predict the residual from older history beyond what can already be predicted from the current state and side information. Consider a restricted residual predictor

$$\widehat{R}_{t+1}^{(0)} = g_{\omega_0}(q_t, \xi_t)$$

and a history-augmented predictor

$$\widehat{R}_{t+1}^{(1)} = g_{\omega_1}\left(q_t, \xi_t, q_{t-1}, \ldots, q_{t-K_{\mathrm{r}}}\right).$$

Their held-out prediction errors can be compared through

$$\Delta_{\mathrm{hist}} = \widehat{\mathbb{E}}\left[\left\|R_{t+1} - \widehat{R}_{t+1}^{(0)}\right\|_2^2\right] - \widehat{\mathbb{E}}\left[\left\|R_{t+1} - \widehat{R}_{t+1}^{(1)}\right\|_2^2\right]. \tag{29}$$

A reliably positive value of Δ_{hist} means that older latent history improves prediction beyond (q_t, ξ_t). This provides evidence against the conditional-mean sufficiency of the current state–predictor pair[3].

The same principle can be implemented by comparing restricted and history-augmented predictors of the target itself rather than predictors of the residual. If a common baseline prediction is used, the two formulations are equivalent because predicting $R_{t+1} = Y_{t+1} - \widehat{Y}_{t+1}$ is equivalent to correcting the baseline prediction $\widehat{Y}_{t+1}$. Experiment 3 uses the direct-prediction version of this diagnostic, comparing prediction from Z_t with prediction from (Z_t, Z_{t-1}).

E.3 Distinguishing state insufficiency from predictor misspecification

Residual predictability can arise because of either the representation or the transition model. First, the current representation may fail to summarize all past information relevant to predicting the future. Second, the transition model may be too restricted or insufficiently optimized even when the representation itself is sufficient.

The distinction between $\mathcal{G}_t$ and $\mathcal{F}_t$ helps separate these effects. Predictability of R_{t+1} from (q_t, ξ_t) alone indicates that the fitted transition has not captured the conditional mean available from its own

[3]The comparison should be evaluated on held-out data or through cross-fitting. The restricted and augmented auxiliary predictors should also have comparable capacity and regularization; otherwise an apparent history gain may reflect unequal model flexibility rather than genuinely additional predictive information.

inputs. The restricted auxiliary model g_{ω_0} can absorb part of this current-input misspecification. Additional held-out improvement after introducing $(q_{t-1}, \ldots, q_{t-K_r})$ then asks a more specific question: does older latent history contain predictive information not recoverable from the current state and side information?

Attributing a positive Δ_{hist} specifically to representation insufficiency nevertheless requires care. The current-input predictor and residual correction must be sufficiently expressive and well fitted, the restricted and augmented diagnostic models should be compared under matched capacity and regularization, and evaluation should be performed out of sample. Under these conditions, incremental predictability from older history is evidence that the current representation has omitted transition-relevant information rather than merely that the original transition parameterization was imperfect.

E.4 Continuous-state diagnostics

For a continuous probabilistic predictor, suppose that

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \mathcal{N}(\mu_t, \Sigma_t),$$

and let

$$Z_{t+1}^{\mathrm{T}} \sim q_{\bar{\theta}}(Z_{t+1} \mid X_{t+1})$$

denote a target-encoder latent sample. When Σ_t is positive definite, a standardized innovation may be defined as

$$R_{t+1}^{\text{std}} = \Sigma_t^{-1/2} \left(\text{sg}(Z_{t+1}^{\mathrm{T}}) - \mu_t \right).$$

For singular or nearly singular covariance matrices, a regularized or pseudoinverse square root should be used instead.

Under a correctly specified conditional Gaussian model, the standardized innovation has conditional mean zero and conditional covariance equal to the identity. Markov sufficiency further implies that older history should not systematically predict this innovation once the current state and side information are given. One may therefore examine lagged residual dependence, residual covariance, squared-residual dependence, or auxiliary history-prediction gains. For non-Gaussian predictors, score-based diagnostics, probability-integral-transform diagnostics in suitable scalar settings, or appropriate multivariate calibration diagnostics can be used instead.

These diagnostics test different consequences of correct conditional prediction. Zero autocorrelation, for example, rules out linear temporal dependence but does not exclude nonlinear or higher-order dependence. Likewise, calibrated marginal probability-integral-transform values do not establish the conditional independence required for Markov sufficiency.

Residual analysis should therefore be interpreted asymmetrically. Predictability from older history provides evidence against Markov sufficiency of the current state–predictor pair. Failure to detect such predictability means only that the chosen diagnostics do not reject sufficiency; it does not prove that the learned representation is Markov sufficient.

F Proofs

F.1 Proof of Proposition 1

For nonnegative integers h_1 and h_2, the definition of matrix powers together with associativity of matrix multiplication gives

$$A^{h_1+h_2} = A^{h_1} A^{h_2}.$$

Left-multiplying by the current state distribution q_t yields

$$q_t A^{h_1+h_2} = \left(q_t A^{h_1} \right) A^{h_2},$$

which is Eq. (6).

More generally, let a total horizon h be partitioned into nonnegative integers

$$h = h_1 + \cdots + h_m.$$

Repeated application of the same identity gives

$$A^h = A^{h_1} \cdots A^{h_m},$$

and therefore

$$q_t A^h = \big(\cdots((q_t A^{h_1})A^{h_2})\cdots\big)A^{h_m}.$$

Hence every decomposition of the same total horizon produces the same predictive distribution, proving the proposition.

F.2 Proof of Proposition 2

Recall from Eq. (11) that, whenever $q_\theta(z) > 0$,

$$p_\theta^{\mathrm{imp}}(x \mid z) = \frac{p_{\mathrm{data}}(x) q_\theta(z \mid x)}{q_\theta(z)},$$

where

$$q_\theta(z) = \int p_{\mathrm{data}}(x) q_\theta(z \mid x)\, dx.$$

For any z with $q_\theta(z) > 0$,

$$\begin{aligned}\int p_\theta^{\mathrm{imp}}(x \mid z)\, dx &= \frac{1}{q_\theta(z)} \int p_{\mathrm{data}}(x) q_\theta(z \mid x)\, dx \\ &= \frac{q_\theta(z)}{q_\theta(z)} = 1.\end{aligned}$$

Thus, $p_\theta^{\mathrm{imp}}(x \mid z)$ is a normalized conditional distribution. For discrete observations, the corresponding integrals are replaced by sums.

Moreover, for any x in the support of p_{data} and any z with $q_\theta(z) > 0$, direct substitution gives

$$\begin{aligned}\frac{p_\theta^{\mathrm{imp}}(x \mid z) q_\theta(z)}{p_{\mathrm{data}}(x)} &= \frac{p_{\mathrm{data}}(x) q_\theta(z \mid x)}{q_\theta(z)} \frac{q_\theta(z)}{p_{\mathrm{data}}(x)} \\ &= q_\theta(z \mid x).\end{aligned}$$

Hence $q_\theta(z \mid x)$ is exactly the posterior associated with the prior $q_\theta(z)$ and the implicit emission $p_\theta^{\mathrm{imp}}(x \mid z)$.

Equivalently, the resulting one-time joint distribution satisfies

$$p_\theta^{\mathrm{imp}}(x \mid z) q_\theta(z) = p_{\mathrm{data}}(x) q_\theta(z \mid x).$$

Its observation marginal is $p_{\mathrm{data}}(x)$, so the construction defines a valid static latent-variable model. As emphasized in the main text, this one-time Bayes completion does not by itself establish a sequence-level HMM; that stronger result additionally requires transition, marginal, and filtering consistency.

F.3 Proof of Proposition 3

By assumption, the future target state is generated as

$$Z_{t+1} = g_{\bar{\theta}}(X_{t+1}, U_{t+1}),$$

so Z_{t+1} is a stochastic post-processing of (X_{t+1}, U_{t+1}). The conditional data-processing inequality therefore gives

$$\mathrm{I}(Z_{t+1}; X_{<t} \mid Z_t) \le \mathrm{I}(X_{t+1}, U_{t+1}; X_{<t} \mid Z_t).$$

By the chain rule for conditional mutual information,

$$\begin{aligned}\mathrm{I}(X_{t+1}, U_{t+1}; X_{<t} \mid Z_t) = \mathrm{I}(X_{t+1}; X_{<t} \mid Z_t) \\ + \mathrm{I}(U_{t+1}; X_{<t} \mid X_{t+1}, Z_t).\end{aligned}$$

Predictive sufficiency,

$$X_{>t} \perp X_{\leq t} \mid Z_t,$$

implies

$$\mathrm{I}(X_{t+1}; X_{<t} \mid Z_t) = 0,$$

because X_{t+1} is contained in $X_{>t}$ and $X_{<t}$ is contained in $X_{\leq t}$.

Likewise, the assumed conditional independence of the target-encoder randomness,

$$U_{t+1} \perp X_{\leq t} \mid (X_{t+1}, Z_t),$$

implies

$$\mathrm{I}(U_{t+1}; X_{<t} \mid X_{t+1}, Z_t) = 0.$$

Consequently,

$$\mathrm{I}(X_{t+1}, U_{t+1}; X_{<t} \mid Z_t) = 0.$$

Combining this equality with the conditional data-processing inequality and the nonnegativity of conditional mutual information yields

$$\mathrm{I}(Z_{t+1}; X_{<t} \mid Z_t) = 0,$$

which proves the proposition. Thus, under predictive sufficiency and the stated target-encoder independence condition, the current representation Z_t screens off older observation history from the next latent state Z_{t+1}. For a deterministic target encoder, the auxiliary randomness U_{t+1} can be omitted.

G Three Realizations of the Emission Direction

The main text describes three alternative ways to complete the state-to-observation direction of the latent-state model. These constructions should not be interpreted as three progressively stronger notions of equivalence. Rather, each can supply the emission component required for an emission-complete latent-state representation. Exact sequence-level HMM equivalence additionally requires the transition, marginal-consistency, and filtering-consistency conditions developed in Section I.

G.1 Explicit decoder

The most direct realization introduces a probabilistic decoder

$$p_\psi(X_t \mid Z_t),$$

which has the same state-to-observation direction as an HMM emission model.

When both the latent state and observation space are finite, this conditional may be represented by an emission matrix B, for example

$$B_{jk} = p_\psi(X_t = k \mid Z_t = j).$$

For a categorical latent state with continuous observations, each latent state instead indexes an observation density. More generally, for images, signals, or other high-dimensional observations, $p_\psi(X_t \mid Z_t)$ may be parameterized by a Gaussian, discretized logistic, autoregressive, diffusion-based, or other suitable conditional observation model.

If the decoder is fitted only after JEPA representation learning while the latent model is held fixed, it acts as a post-hoc observation model or probe. If it participates jointly in training, it becomes part of the generative latent-state model and can contribute directly to observation-sequence likelihood.

G.2 Invertible target encoder

A second realization is available when the target encoder

$$f_{\bar{\theta}} : \mathcal{X} \to \mathcal{S}$$

is bijective on the modeled data domain. Its target representation satisfies

$$Z_t^{\mathrm{T}} = f_{\bar{\theta}}(X_t), \qquad X_t = f_{\bar{\theta}}^{-1}(Z_t^{\mathrm{T}}).$$

Identifying the latent state with this target representation gives the deterministic state-to-observation kernel

$$p(X_t \mid Z_t) = \delta\big(X_t - f_{\bar{\theta}}^{-1}(Z_t)\big)\,.$$

Thus, invertibility supplies the required state-to-observation direction without introducing a separate decoder.

A deterministic inverse should nevertheless be distinguished from a non-degenerate probabilistic emission. If $f_{\bar{\theta}}$ is a tractable invertible density model, the change-of-variables formula can be used to evaluate the observation density induced by a latent density. The conditional map $X_t = f_{\bar{\theta}}^{-1}(Z_t)$ itself remains deterministic, however. A non-degenerate conditional emission can instead be obtained by augmenting the inverse map with an observation-noise model, for example

$$X_t = f_{\bar{\theta}}^{-1}(Z_t) + \varepsilon_t,$$

with a specified noise distribution for ε_t.

Exact invertibility imposes strong architectural constraints. In particular, it prevents unrestricted dimensionality reduction and may require the representation to preserve observation details that a predictive information bottleneck would otherwise discard. Standard compressed JEPA encoders therefore need not admit this construction.

G.3 Implicit emission

When no explicit decoder is parameterized and the target encoder is not invertible, a local stochastic encoder can still induce a state-to-observation conditional. As defined in Eq. (11),

$$p_\theta^{\mathrm{imp}}(x \mid z) = \frac{p_{\mathrm{data}}(x) q_\theta(z \mid x)}{q_\theta(z)},$$

for $q_\theta(z) > 0$. As shown in the preceding proof, this conditional is normalized and, together with $q_\theta(z)$, reproduces the one-time joint distribution

$$p_\theta^{\mathrm{imp}}(x \mid z) q_\theta(z) = p_{\mathrm{data}}(x) q_\theta(z \mid x).$$

This construction provides an exact static probabilistic completion of the observation–state relationship, but it does not automatically provide a practical generative model. In particular, $p_\theta^{\mathrm{imp}}(x \mid z)$ depends explicitly on the generally unknown data marginal $p_{\mathrm{data}}(x)$, so direct sampling and likelihood evaluation may be intractable.

Moreover, the construction uses a *local* encoder $q_\theta(z \mid x)$. An arbitrary history-dependent encoder $q_\theta(Z_t \mid X_{\le t})$ cannot simply be reinterpreted as an emission model. To obtain an exact sequence-level HMM from the implicit construction, the resulting one-time conditionals must additionally be consistent with the latent transition and with the Bayesian filtering recursion, as formalized in Section I.

H HMM-Style Training Objectives for PIB-VJEPA

The HMM correspondence suggests an alternative to purely latent-space JEPA training. Once a valid observation model is available, the latent transition and emission can be trained from observation-sequence likelihood, while the history-dependent context encoder can be aligned with the corresponding Bayesian filtering distribution. This appendix summarizes the probabilistic objectives underlying the model-and-objective correspondence developed in the main text.

H.1 Sequence likelihood

Suppose that the latent-state model is equipped with an initial-state distribution $p_0(Z_1)$, transition model

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t),$$

and explicit emission model

$$p_\psi(X_t \mid Z_t).$$

The resulting conditional sequence model is

$$p_{\phi,\psi}\left(X_{1:T}, Z_{1:T} \mid \xi_{1:T-1}\right) = p_0(Z_1)\prod_{t=1}^{T} p_\psi(X_t \mid Z_t)\prod_{t=1}^{T-1} p_\phi(Z_{t+1} \mid Z_t, \xi_t).$$

Marginalizing the latent trajectory gives the observation-sequence evidence

$$p_{\phi,\psi}\left(X_{1:T} \mid \xi_{1:T-1}\right) = \int p_{\phi,\psi}\left(X_{1:T}, Z_{1:T} \mid \xi_{1:T-1}\right) dZ_{1:T}.$$

Training from this evidence directly constrains the transition–emission model in observation space, in contrast to the standard JEPA objective, which is imposed primarily in latent space.

H.2 Exact likelihood and filtering for categorical MCJEPA

For categorical states $Z_t \in \{1, \ldots, K\}$, the sequence likelihood and filtering posterior can be evaluated exactly by the HMM forward recursion. Let

$$\pi_j = p_0(Z_1 = j), \qquad b_\psi(X_t \mid j) = p_\psi(X_t \mid Z_t = j),$$

and

$$(A_t)_{ij} = p_\phi(Z_{t+1} = j \mid Z_t = i, \xi_t).$$

For the time-homogeneous MCJEPA model, $A_t = A$.

Define the forward message

$$\alpha_t(j) = p_{\phi,\psi}\left(X_{1:t}, Z_t = j \mid \xi_{1:t-1}\right).$$

It satisfies

$$\alpha_1(j) = \pi_j b_\psi(X_1 \mid j)$$

and

$$\alpha_{t+1}(j) = b_\psi(X_{t+1} \mid j)\sum_{i=1}^{K} \alpha_t(i)(A_t)_{ij}.$$

The sequence evidence is therefore

$$p_{\phi,\psi}\left(X_{1:T} \mid \xi_{1:T-1}\right) = \sum_{j=1}^{K} \alpha_T(j).$$

In practice, the recursion is evaluated in log space or with normalized forward messages for numerical stability.

Normalizing the forward messages also gives the exact filtering posterior

$$\widetilde{q}_t = p_{\phi,\psi}\left(Z_t \mid X_{\le t}, \xi_{<t}\right).$$

The history-dependent PIB-VJEPA context encoder can then be trained as an amortized filter using

$$\mathcal{L}_{\text{filter}} = \mathbb{E}_t\left[\mathrm{KL}\left(\mathrm{sg}(\widetilde{q}_t) \,\|\, q_\theta(Z_t \mid X_{\le t})\right)\right].$$

Thus, the encoder learns to approximate in one forward pass the current-state posterior that the HMM computes recursively.

This distinction is important: sequence likelihood trains the transition and emission model, whereas filtering distillation trains the context encoder to reproduce the corresponding filtering belief. Sequence likelihood alone does not require an independently parameterized history encoder to equal that filter.

H.3 CONTINUOUS-STATE EXTENSION

For continuous or nonlinear latent states, exact marginalization of $Z_{1:T}$ is generally unavailable. Introducing an approximate sequence posterior

$$q_\eta\left(Z_{1:T} \mid X_{1:T}, \xi_{1:T-1}\right)$$

gives the standard variational lower bound

$$\log p_{\phi,\psi}\left(X_{1:T} \mid \xi_{1:T-1}\right) \geq \mathbb{E}_{q_\eta}\Bigg[\sum_{t=1}^{T} \log p_\psi(X_t \mid Z_t) + \log p_0(Z_1) + \sum_{t=1}^{T-1} \log p_\phi(Z_{t+1} \mid Z_t, \xi_t) - \log q_\eta\left(Z_{1:T} \mid X_{1:T}, \xi_{1:T-1}\right)\Bigg].$$

The approximate posterior may be causal when online filtering is required or smoothing when full-sequence information is available during training. We include this extension to show how the same model-and-objective interpretation extends beyond the finite categorical setting; the experiments in this paper use exact finite-state inference.

H.4 RELATION TO JEPA AND HYBRID TRAINING

HMM-style sequence learning and JEPA latent prediction are distinct objectives even when they operate on the same latent-state architecture. In the categorical setting studied in Experiment 4, the main text compares

$$\underbrace{\mathcal{L}_{\mathrm{MC}}}_{\text{JEPA latent prediction}} \quad \text{and} \quad \underbrace{\mathcal{L}_{\mathrm{HMM}} + \mathcal{L}_{\mathrm{filter}}}_{\text{HMM-style training}},$$

with $\mathcal{L}_{\mathrm{state}}$ used as a shared state-use regularizer. The hybrid regime combines all three signals,

$$\mathcal{L}_{\mathrm{HMM}} + \mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{filter}},$$

with the corresponding weights given in Sections 7.4 and J.

Importantly, the exact HMM filtering posterior is used only as the target of $\mathcal{L}_{\mathrm{filter}}$. It does not replace the EMA future target in $\mathcal{L}_{\mathrm{MC}}$: the JEPA component retains the same target-encoder construction as the JEPA-only regime. The transition matrix is shared between the HMM sequence objective and the MCJEPA latent-prediction objective, so the two training signals constrain the same latent dynamics from observation-space and representation-space perspectives, respectively.

Consequently, adding an emission model alone does not make PIB-VJEPA training identical to HMM training. Model-and-objective equivalence additionally requires observation-sequence evidence and the corresponding sequence-inference semantics to participate in the learning objective.

I EXACT HMM REPRESENTATION CONDITIONS

This appendix makes precise the sufficient conditions in Theorem 1. We state the construction for a discrete latent state for clarity; the same argument extends to general Markov kernels by replacing sums with integrals. Side information ξ_t is treated as observed, so all sequence distributions below are conditional on $\xi_{1:T-1}$.

Let the latent transition be

$$A_t(i,j) = p_\phi(Z_{t+1} = j \mid Z_t = i, \xi_t),$$

and let

$$b_t(x \mid j)$$

denote a valid state-to-observation conditional. This emission may be supplied by an explicit decoder, an invertible target encoder interpreted as a deterministic kernel, or the implicit construction described below. Given an initial distribution ρ_1, these components define

$$p(z_{1:T}, x_{1:T} \mid \xi_{1:T-1}) = \rho_1(z_1) \prod_{t=1}^{T} b_t(x_t \mid z_t) \prod_{t=1}^{T-1} A_t(z_t, z_{t+1}), \tag{30}$$

which is the conditional HMM factorization.

I.1 Marginal and filtering consistency

Let

$$\rho_t(j) = p(Z_t = j \mid \xi_{1:t-1})$$

denote the latent marginal before observing X_t, conditional on the side-information history. Dynamic consistency requires

$$\rho_{t+1}(j) = \sum_i \rho_t(i) A_t(i,j). \tag{31}$$

Thus, the one-time latent marginals must be generated by the same transition kernel used by the temporal model.

For a realized observation history, let

$$q_t(j) = p(Z_t = j \mid X_{\le t}, \xi_{<t})$$

denote the filtering distribution. Its transition-based predictive prior is

$$\pi_t(j) = \sum_i q_{t-1}(i) A_{t-1}(i,j),$$

with $\pi_1 = \rho_1$. Bayes' rule then gives the filtering recursion

$$q_t(j) = \frac{b_t(X_t \mid j)\pi_t(j)}{\sum_k b_t(X_t \mid k)\pi_t(k)}. \tag{32}$$

The history-dependent PIB-VJEPA encoder is filtering-consistent when

$$q_\theta(Z_t \mid X_{\le t}, \xi_{<t}) = q_t \tag{33}$$

for the transition and emission model under consideration. An arbitrary history encoder need not satisfy this equality.

I.2 Explicit and invertible emissions

If a decoder directly specifies

$$b_t(x \mid z) = p_\psi(x \mid z),$$

then Eq. (30) follows immediately from the initial distribution, first-order transition, and emission model. If the target encoder is invertible, the deterministic kernel induced by

$$x = f_{\bar\theta}^{-1}(z)$$

plays the same role. In either case, if the latent marginals satisfy Eq. (31) and the history encoder satisfies Eq. (33), the resulting temporal JEPA admits the sequence-level HMM interpretation stated in Theorem 1.

I.3 Implicit-emission case

The less direct case begins with a local evidence encoder

$$e_{\theta,t}(z \mid x)$$

that depends only on the current observation. Let $p_t(x)$ be the one-time observation marginal and define its induced latent marginal

$$\rho_t(z) = \int p_t(x) e_{\theta,t}(z \mid x)\, dx. \tag{34}$$

For $\rho_t(z) > 0$, define

$$b_t^{\mathrm{imp}}(x \mid z) = \frac{p_t(x) e_{\theta,t}(z \mid x)}{\rho_t(z)}.$$

By the argument in the proof of implicit emission completion, this is a normalized state-to-observation conditional and satisfies

$$e_{\theta,t}(z \mid x) = \frac{b_t^{\mathrm{imp}}(x \mid z)\rho_t(z)}{p_t(x)}.$$

The locality assumption is important: $e_{\theta,t}(z \mid x_t)$ supplies the observation-dependent evidence factor, whereas the history-dependent context encoder represents the filtering belief. The two should not be identified.

Substituting the implicit emission into the filtering recursion gives

$$\begin{aligned} q_t(z) &\propto \pi_t(z) b_t^{\mathrm{imp}}(x_t \mid z) \\ &= \pi_t(z) \frac{p_t(x_t) e_{\theta,t}(z \mid x_t)}{\rho_t(z)} \\ &\propto \pi_t(z) \frac{e_{\theta,t}(z \mid x_t)}{\rho_t(z)}. \end{aligned}$$

Hence the history-dependent filtering distribution may equivalently be written as

$$q_t(z) \propto \pi_t(z) \frac{e_{\theta,t}(z \mid x_t)}{\rho_t(z)}. \tag{35}$$

For an exact sequence-level interpretation, the induced marginals in Eq. (34) must additionally be dynamically consistent with the transition:

$$\rho_{t+1}(z') = \sum_z \rho_t(z) A_t(z, z'). \tag{36}$$

Finally, the PIB-VJEPA history encoder must coincide with the filtering distribution generated by Eq. (35).

I.4 Completion of the proof for Theorem.1

We can now verify the four sufficient conditions in Theorem 1. First, A_t defines first-order latent Markov dynamics. Second, one of the three constructions above supplies a valid state-to-observation conditional. Third, Eq. (31), or Eq. (36) in the implicit case, ensures that the latent marginals evolve under the same transition kernel. Fourth, Eq. (33) identifies the history-dependent context encoder with the Bayesian filtering posterior of that transition–emission model.

Therefore the joint sequence distribution is precisely Eq. (30), and the context encoder represents its filtering distribution. This proves the sufficient-condition statement in Theorem 1.

These conditions are stronger than architectural correspondence alone. In particular, a valid transition and emission specify an HMM-compatible generative model, but an arbitrary JEPA history encoder need not equal its Bayesian filter, and independently induced one-time latent marginals need not evolve according to the learned transition. The additional consistency conditions are what promote an emission-complete latent-state representation to exact sequence-level HMM equivalence.

J Experimental Details

This appendix provides the data-generation procedures, model architectures, optimization settings, evaluation metrics, and supplementary results for the experiments in Section 7. The experiments are deliberately small and synthetic because their purpose is to isolate the structural claims of the paper under known latent dynamics rather than to benchmark large-scale forecasting performance.

Unless otherwise stated, experiments involving sampled data or learned models use five random seeds,

$$\{0, 1, 2, 3, 4\},$$

and report mean $\pm$ one standard deviation across seeds. Exact finite calculations in Experiment 3, such as deterministic-partition enumeration, are deterministic and are therefore reported without seed variability. We do not perform formal hypothesis tests; the experiments are intended as controlled structural diagnostics, and across-seed variability is reported to expose sampling and optimization variability.

For the finite-HMM experiments, the paper configuration uses 400 training sequences and 160 test sequences of length 80, with prediction horizons

$$\mathcal{H} = \{1, 2, 4, 8\}.$$

No separate validation split is used because hyperparameter selection is not the purpose of these controlled diagnostics. Within a seed, competing methods are evaluated on the same generated data whenever a paired comparison is intended. Experiment 4 independently regenerates the separated-emission data used in Experiment 1 with the same data-generating process and seeds, but none of the fitted Experiment 1 models is reused.

J.1 Evaluation metrics

We collect here the evaluation metrics used across the experiments. This also separates *representation recovery*, *transition recovery*, *predictive performance*, and *probabilistic-model fit*, which measure different aspects of the proposed correspondence.

State recovery: ARI and NMI. When ground-truth latent states are available, we convert each learned categorical distribution q_t to a hard state assignment

$$\widehat{S}_t = \arg\max_k q_t(k).$$

We report the adjusted Rand index (ARI) and normalized mutual information (NMI) between the learned assignments $\widehat{S}_t$ and ground-truth states S_t.

For a contingency table with entries n_{ij}, row sums a_i, column sums b_j, and total sample size N, ARI is

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \frac{\left(\sum_i \binom{a_i}{2}\right)\left(\sum_j \binom{b_j}{2}\right)}{\binom{N}{2}}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \frac{\left(\sum_i \binom{a_i}{2}\right)\left(\sum_j \binom{b_j}{2}\right)}{\binom{N}{2}}}. \tag{37}$$

ARI corrects the ordinary Rand index for agreement expected by chance. A value of 1 denotes identical partitions, while values near 0 correspond to chance-level agreement under the adjustment.

NMI is computed using the arithmetic normalization,

$$\mathrm{NMI}(S, \widehat{S}) = \frac{2I(S; \widehat{S})}{H(S) + H(\widehat{S})}. \tag{38}$$

NMI lies in $[0, 1]$, with larger values indicating greater shared information between the learned and ground-truth state partitions. Both ARI and NMI are invariant to permutation of categorical state labels.

Permutation alignment. Although ARI and NMI do not require label alignment, transition matrices and predicted categorical probabilities do. We therefore compute a Hungarian assignment on the *training-set* hard state assignments. Let M denote the resulting permutation matrix mapping learned-state order to ground-truth-state order. A learned transition matrix $\widehat{A}$ is aligned as

$$\widehat{A}_{\mathrm{aligned}} = M^\top \widehat{A} M.$$

The mapping is fitted only on training assignments and then held fixed for test evaluation.

Transition recovery. When the true transition matrix $A^\star$ is known, we measure normalized Frobenius error,

$$\mathcal{E}_A = \frac{\left\| \widehat{A}_{\mathrm{aligned}} - A^\star \right\|_F}{\|A^\star\|_F}. \tag{39}$$

Lower values indicate more faithful recovery of the underlying Markov transition law. This metric evaluates the learned dynamics themselves rather than only their downstream predictive consequences.

True-state prediction NLL. For prediction horizon h, let $\widehat{q}_{t+h}$ denote the predicted categorical distribution after alignment to ground-truth state order. We report

$$\mathcal{L}_h^{\text{state}} = -\mathbb{E}_t \left[\log \widehat{q}_{t+h}(S_{t+h}) \right]. \tag{40}$$

Thus the metric measures the probability assigned to the actual future latent state. Lower values are better. For the shared-transition models,

$$\widehat{q}_{t+h} = q_t A^h.$$

Observation-sequence NLL. For models equipped with a transition–emission likelihood, observation-space fit is measured by negative log-likelihood per time step,

$$\mathcal{L}_{\text{seq}} = -\frac{1}{T} \mathbb{E} \left[\log p_{\phi,\psi}(X_{1:T}) \right], \tag{41}$$

where $p_{\phi,\psi}(X_{1:T})$ denotes the marginal observation-sequence density induced by the latent transition and emission models after marginalizing the latent-state sequence. In the finite-state setting used in our experiments,

$$p_{\phi,\psi}(X_{1:T}) = \sum_{Z_{1:T}} p_\phi(Z_1) \left[\prod_{t=1}^{T} p_\psi(X_t \mid Z_t) \right] \left[\prod_{t=1}^{T-1} p_\phi(Z_{t+1} \mid Z_t) \right].$$

Thus, the likelihood integrates out the unobserved latent trajectory rather than conditioning on the ground-truth latent states. In practice, this marginalization is evaluated exactly and efficiently by the HMM forward algorithm rather than by explicitly enumerating all possible latent-state sequences.

Unlike state-prediction NLL, this metric evaluates the probability density assigned to the observed sequence rather than the probability assigned to the known synthetic latent state. In Experiment 4, the same mathematical quantity plays different roles across training regimes. For the HMM+filter and hybrid regimes, it is optimized during training as $\mathcal{L}_{\text{HMM}}$ and subsequently evaluated on held-out sequences. For the JEPA-only regime, no observation-sequence likelihood is optimized during representation learning; $\mathcal{L}_{\text{seq}}$ is computed only after fitting the post-hoc emission model. It therefore serves as a common evaluation metric across the three regimes rather than a common training objective.

Path disagreement. To measure whether direct and composed multi-step predictions agree, we use

$$\mathcal{D}_{\text{path}}(h_1, h_2) = \mathbb{E}_t \left[\| q_t A_{h_1+h_2} - (q_t A_{h_1}) A_{h_2} \|_1 \right]. \tag{42}$$

For MCJEPA with one shared transition matrix,

$$A_h = A^h,$$

and therefore

$$\mathcal{D}_{\text{path}}(h_1, h_2) = 0$$

algebraically, up to numerical precision. For independently learned horizon-specific matrices no such guarantee exists.

Filtering KL. When both an exact model-based filtering distribution and an amortized context-encoder distribution are available, we report

$$\mathcal{D}_{\text{filter}} = \mathbb{E}_t \left[\text{KL} \left(q_t^{\text{exact}} \,\middle\|\, q_t^{\text{enc}} \right) \right]. \tag{43}$$

Lower values mean that the amortized encoder more closely reproduces the corresponding filtering belief. In Experiment 4 this quantity is training aligned for the HMM+filter and hybrid regimes because both explicitly optimize filtering distillation; it is therefore interpreted as a role diagnostic rather than an independent generalization metric.

Accuracy, Brier score, and entropy. Experiment 2 additionally reports state accuracy,

$$\mathrm{Acc} = \frac{1}{N}\sum_t \mathbf{1}\left\{\arg\max_k q_t(k) = S_t\right\}, \tag{44}$$

the multiclass Brier score,

$$\mathrm{Brier} = \frac{1}{N}\sum_t \sum_k \left(q_t(k) - \mathbf{1}\{S_t = k\}\right)^2, \tag{45}$$

and mean posterior entropy,

$$\overline{H} = \frac{1}{N}\sum_t H(q_t). \tag{46}$$

Accuracy measures hard classification correctness, whereas NLL and Brier score retain information about probabilistic confidence. Posterior entropy is descriptive and should not be interpreted as a performance metric by itself.

J.2 Experiment 1: finite-HMM recovery and Markov composition

Data generation. We generate observations from a four-state stationary Gaussian HMM. The ground-truth transition matrix is

$$A^\star = \begin{bmatrix} 0.85 & 0.10 & 0.05 & 0.00 \\ 0.05 & 0.85 & 0.10 & 0.00 \\ 0.00 & 0.05 & 0.85 & 0.10 \\ 0.10 & 0.00 & 0.05 & 0.85 \end{bmatrix}.$$

The initial state is sampled from the stationary distribution of $A^\star$. Conditional on state $S_t = k$, the two-dimensional observation is generated as

$$X_t \mid S_t = k \sim \mathcal{N}(\mu_k, \sigma^2 I_2),$$

with state means

$$\mu_1 = (-1,-1), \qquad \mu_2 = (-1,1), \qquad \mu_3 = (1,1), \qquad \mu_4 = (1,-1).$$

We consider two emission regimes:

$$\sigma = 0.35 \quad \text{(separated)}, \qquad \sigma = 0.80 \quad \text{(ambiguous)}.$$

For each seed and regime we independently generate 400 training sequences and 160 test sequences, each of length 80.

MCJEPA architecture. The online context encoder is a one-layer GRU with hidden dimension 48, followed by a linear projection to $K = 4$ logits and a softmax:

$$q_\theta(Z_t \mid X_{\le t}) = \mathrm{softmax}\left(W h_t + b\right).$$

The target encoder has the same architecture but processes each X_t as an independent length-one sequence, producing a local target distribution. Its parameters are initialized from the online encoder and subsequently updated by exponential moving average.

For the shared-transition MCJEPA model, a trainable 4×4 logit matrix is row-normalized by softmax,

$$A = \mathrm{softmax}_{\mathrm{row}}(L_A),$$

and horizon-h prediction uses

$$\widehat{q}_{t+h} = q_t A^h.$$

The transition logits are initialized with a mild diagonal bias,

$$L_A = I_4.$$

The horizon-specific baseline uses the same online and target encoders but replaces the shared matrix with independent row-stochastic matrices

$$A_1, \quad A_2, \quad A_4, \quad A_8.$$

Each matrix is initialized with the same diagonal logit bias, but no constraint requires

$$A_h = A_1^h.$$

Warm start and optimization. To make the small synthetic recovery experiment stable and reproducible, both categorical JEPA variants receive an unsupervised K-means warm start. K-means with $K = 4$ and 10 initializations is fitted to individual training observations; ground-truth states are never used. The online encoder is then trained for 100 warm-start updates with Adam at learning rate

$$10^{-2}$$

to predict the K-means assignments, after which the target encoder is copied from the online encoder.

The main MCJEPA optimization runs for 180 epochs with mini-batches of 64 sequences and Adam learning rate

$$3 \times 10^{-3}.$$

The prediction loss averages the target-to-prediction KL divergence over $\mathcal{H} = \{1, 2, 4, 8\}$:

$$\mathcal{L}_{\text{pred}} = \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} \mathbb{E}\left[\mathrm{KL}\left(\bar{q}_{t+h} \,\|\, q_t A_h\right)\right],$$

where $A_h = A^h$ for MCJEPA and A_h is independently learned for the horizon-specific baseline.

The state-use terms are

$$\mathcal{L}_{\text{occ}} = \mathrm{KL}\left(\bar{q} \| \mathrm{Unif}(K)\right),$$
$$\mathcal{L}_{\text{ent}} = \mathbb{E}_t\left[H(q_t)\right],$$

with

$$\lambda_{\text{occ}} = 0.30, \qquad \lambda_{\text{ent}} = 0.02.$$

Because $\mathcal{L}_{\text{ent}}$ is minimized, it encourages confident per-example assignments. The target encoder uses EMA coefficient

$$\tau = 0.995.$$

Gradients are clipped to norm 5.

Gaussian-HMM baseline. The HMM baseline uses the correctly specified four-state family with a learned initial-state distribution, row-stochastic transition matrix, and state-conditional diagonal Gaussian emissions. The emission means are initialized from K-means cluster centers, and the diagonal variances are initialized from within-cluster variances with a small additive floor.

The HMM is trained directly through the observation-sequence NLL defined above, computed exactly by the forward algorithm. We use Adam with learning rate

$$3 \times 10^{-2}$$

for 350 optimization steps and clip gradients to norm 10.

Permutation alignment and evaluation. The Hungarian alignment and common metrics follow Section J.1. ARI and NMI evaluate recovery of the hidden-state partition, transition error evaluates recovery of $A^\star$, and true-state NLL evaluates future-state prediction at

$$h \in \{1, 2, 4, 8\}.$$

For path consistency we evaluate

$$(h_1, h_2) \in \{(1, 1), (2, 2), (4, 4)\}.$$

For the shared-transition model,

$$q_t A^{h_1+h_2} = (q_t A^{h_1}) A^{h_2}$$

exactly, so path disagreement is zero by construction. The horizon-specific baseline has no corresponding constraint.

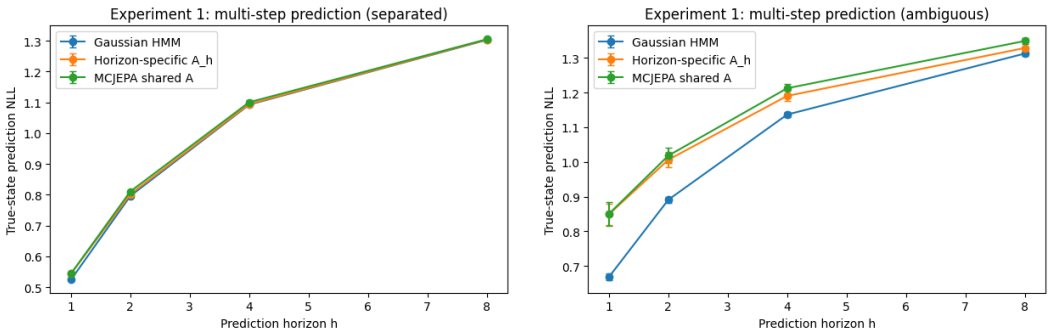


Figure 6: Complete multi-horizon state-prediction results for Experiment 1. *Left:* separated emissions. *Right:* ambiguous emissions. Lower true-state NLL is better. Error bars denote mean $\pm$ one standard deviation over five seeds.

State-usage metrics. In addition to the common evaluation metrics, we monitor both soft and hard effective state counts. If

$$\bar{q} = \frac{1}{N} \sum_{n} q_n$$

is the average soft assignment distribution, then

$$K_{\text{eff}}^{\text{soft}} = \exp\left(H(\bar{q})\right).$$

For hard assignments, let $\widehat{p}_k$ be the empirical frequency of state k. We define

$$K_{\text{eff}}^{\text{hard}} = \exp\left(-\sum_{k:\widehat{p}_k>0} \widehat{p}_k \log \widehat{p}_k\right).$$

The reported assignment entropy is

$$\overline{H}_{\text{assign}} = \mathbb{E}_t[H(q_t)].$$

Effective state count measures diversity of state usage, whereas assignment entropy measures confidence of individual assignments. Low assignment entropy is not desirable by itself: an encoder that confidently maps every observation to one state also has low entropy. State-use diversity and assignment confidence must therefore be interpreted jointly.

Collapse ablations. The collapse diagnostic is run separately from the warm-started recovery experiment. We generate a new separated-emission dataset with $\sigma = 0.35$ and train the shared-A model from random initialization, deliberately omitting the K-means warm start. The four settings are

$$(\lambda_{\text{occ}}, \lambda_{\text{ent}}) \in \{(0,0),\ (0.30,0),\ (0,0.02),\ (0.30,0.02)\}.$$

Each model is trained for 180 epochs with batch size 64; for this diagnostic the EMA coefficient is $\tau = 0.99$. This deliberately creates a more collapse-prone optimization problem and isolates the complementary roles of the two penalties: occupancy regularization discourages global state under-use, whereas entropy regularization encourages confident per-example assignments.

Supplementary results. Figure 6 reports the complete multi-horizon prediction curves omitted from the main text. In the separated regime, all three models remain close across horizons. Under ambiguous emissions, the correctly specified HMM remains strongest, while the horizon-specific and shared-transition JEPA models exhibit similar predictive NLL despite their substantially different structural consistency.

The separated-emission path-consistency result is shown in Fig. 7. As in the ambiguous regime, the shared-A model is exactly compositionally consistent, whereas independently trained horizon-specific matrices exhibit nonzero direct-versus-composed disagreement.

Figure 8 supplements the main-text ARI ablation with state-usage and assignment-confidence diagnostics.

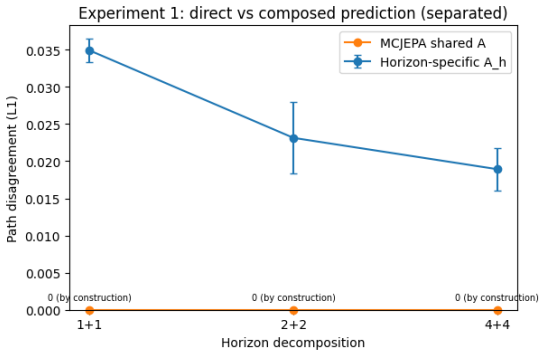


Figure 7: Direct-versus-composed prediction disagreement in the separated-emission regime. MC-JEPA has zero path disagreement by construction because all horizons are powers of one shared transition matrix.

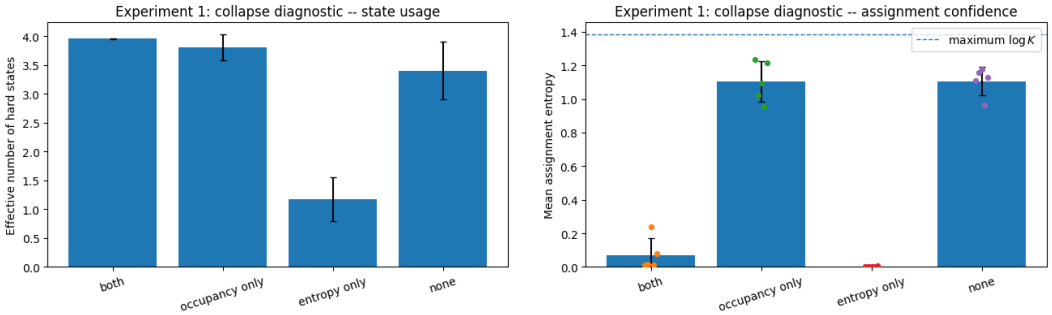


Figure 8: Additional collapse diagnostics for Experiment 1. *Left:* effective number of hard states. *Right:* mean per-example assignment entropy. Occupancy regularization primarily promotes broad global state usage, whereas entropy regularization promotes confident assignments; neither diagnostic should be interpreted in isolation.

J.3 Experiment 2: filtering under emission ambiguity

Data generation. Experiment 2 uses a persistent two-state HMM with transition matrix

$$A = \begin{bmatrix} 0.97 & 0.03 \\ 0.03 & 0.97 \end{bmatrix}.$$

Its stationary distribution is uniform,

$$\pi = (0.5, 0.5).$$

The scalar observation model is

$$X_t \mid S_t = \begin{cases} \mathcal{N}(-\mu, \sigma^2), & S_t = 0, \\ \mathcal{N}(+\mu, \sigma^2), & S_t = 1, \end{cases}$$

with

$$\sigma = 1.$$

Emission ambiguity is controlled by

$$\frac{\mu}{\sigma} \in \{2.0, 1.25, 0.75, 0.45\}.$$

For every seed and separation value we generate 160 sequences of length 80, corresponding to 12,800 state–observation pairs per setting. Because the comparison uses exact oracle posteriors, there is no learned train/test model split in this experiment; independent random seeds provide repeated sampled datasets.

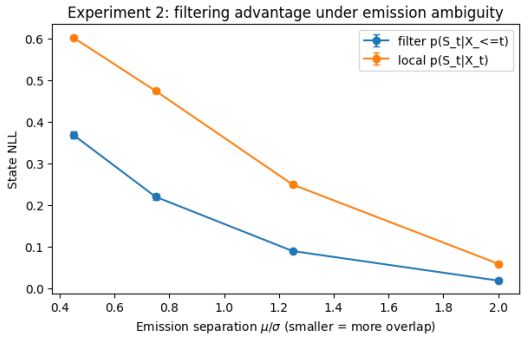


Figure 9: State NLL for exact local evidence and exact filtering under increasing emission ambiguity. Smaller μ/σ corresponds to stronger overlap between the two Gaussian emissions. Error bars denote mean $\pm$ one standard deviation over five independently generated datasets.

Exact local evidence. The local posterior uses only the current observation and the stationary state prior:

$$q_t^{\text{local}}(k) = p(S_t = k \mid X_t) = \frac{\pi_k p(X_t \mid S_t = k)}{\sum_j \pi_j p(X_t \mid S_t = j)}.$$

This is the oracle counterpart of a local encoder $q_\theta(Z_t \mid X_t)$.

Exact filtering. The filtering posterior incorporates both the propagated previous belief and the current emission evidence. At the first step,

$$q_0^{\text{filter}}(k) \propto \pi_k p(X_0 \mid S_0 = k).$$

Thereafter,

$$q_t^{\text{filter}} \propto \left(q_{t-1}^{\text{filter}} A\right) \odot p(X_t \mid S_t),$$

followed by normalization across the two states. This quantity is exactly

$$p(S_t \mid X_{\le t}).$$

No learned HMM and MCJEPA models are being compared in Experiment 2. Both curves are oracle calculations under the same known generating process. This design isolates the informational value of temporal history from representation-learning and optimization effects.

Evaluation. We use the accuracy, state NLL, Brier score, and posterior entropy defined in Section J.1. Accuracy gives the most immediately interpretable state-recovery comparison, while NLL measures whether the posterior assigns high probability to the realized state. Brier score provides a complementary proper probabilistic score, and entropy records posterior confidence.

Representative sequence selection. The representative trajectory in Fig. 3 is selected only for visualization; all quantitative results use all generated sequences. We use the most ambiguous setting,

$$\mu/\sigma = 0.45,$$

from the first seed and search 45-step windows centered on genuine latent-state transitions. Windows containing one to three true state switches receive a small preference, and among candidate windows we favor those in which filtering gives a larger realized-state NLL improvement over local evidence. This produces a transition-rich example that visibly illustrates the mechanism quantified by the aggregate experiment rather than selecting the first sequence arbitrarily.

Supplementary result. Figure 9 gives the complete state-NLL comparison across emission separations. The filtering advantage grows as μ/σ decreases, matching the accuracy trend reported in the main text.

J.4 Experiment 3: predictive compression and Markovization

Second-order binary process. Let

$$Y_t = X_{t+1}.$$

The data-generating process is

$$p(X_{t+1} = 1 \mid X_{t-1}, X_t) = \begin{cases} 0.10, & (X_{t-1}, X_t) = (0,0), \\ 0.90, & (X_{t-1}, X_t) = (0,1), \\ 0.80, & (X_{t-1}, X_t) = (1,0), \\ 0.20, & (X_{t-1}, X_t) = (1,1). \end{cases}$$

The corresponding first-order transition matrix on pair states

$$(00, 01, 10, 11)$$

is

$$A_{\text{pair}} = \begin{bmatrix} 0.9 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0.9 \\ 0.2 & 0.8 & 0 & 0 \\ 0 & 0 & 0.8 & 0.2 \end{bmatrix}.$$

Its stationary distribution is

$$\pi_{\text{pair}} = \left(\frac{16}{41}, \frac{8}{41}, \frac{8}{41}, \frac{9}{41} \right) \approx (0.3902, 0.1951, 0.1951, 0.2195).$$

The exact joint distribution of

$$H_t = (X_{t-2}, X_{t-1}, X_t)$$

and $Y_t = X_{t+1}$ is constructed analytically from this stationary pair chain. Consequently, the exact representation controls and deterministic frontier do not require Monte Carlo estimation.

Exact representation controls. A deterministic representation is a mapping

$$Z = f(H).$$

For such a mapping,

$$I(H; Z) = H(Z),$$

because $H(Z \mid H) = 0$. For each representation we construct the exact joint distribution $p(z, y)$ and compute

$$I(Z; Y) = \sum_{z,y} p(z, y) \log \frac{p(z, y)}{p(z)p(y)}.$$

All logarithms are natural, so information quantities and NLLs are measured in nats.

The optimal one-step probabilistic predictor for a fixed representation is the exact conditional distribution $p(Y \mid Z)$. Its prediction NLL is therefore

$$\mathcal{L}_{\text{pred}} = -\sum_{z,y} p(z, y) \log p(y \mid z) = H(Y \mid Z).$$

The three control mappings are

$$\begin{aligned} Z^{\text{over}} &= (X_{t-2}, X_{t-1}, X_t), \\ Z^{\text{suff}} &= (X_{t-1}, X_t), \\ Z^{\text{under}} &= X_t. \end{aligned}$$

The first has eight possible values, the second four, and the third two. Their exact information and prediction quantities are reproduced in Table 6 for completeness.

Table 6: Exact structural controls for Experiment 3. The four-state predictive pair removes redundant history while preserving all one-step predictive information.

Representation	$I(H;Z)$	$I(Z;Y)$	$H(Y \mid Z)$	States
Three-bit history	1.7356	0.2807	0.3978	8
Predictive pair	1.3378	0.2807	0.3978	4
Current bit only	0.6785	0.0192	0.6593	2

Table 7: Exact deterministic compression regimes in Experiment 3. At $\beta = 0$, the four-state entry is the lower-information representative selected among predictively tied optima.

Compression range	$I(H;Z)$	$I(Z;Y)$	Pred. NLL	States
$\beta = 0$ (tie-broken)	1.3378	0.2807	0.3978	4
$0 < \beta \leq 0.014$	1.3378	0.2807	0.3978	4
$0.015 \leq \beta \leq 0.30$	0.6689	0.2711	0.4074	2
$\beta = 1$	0	0	0.6785	1

Enumeration of deterministic partitions. Because H_t has only eight possible values, every deterministic compression can be enumerated. A deterministic encoder identifies histories that share the same output label and therefore corresponds to a set partition of the eight histories. The number of such partitions is the eighth Bell number,

$$B_8 = 4140.$$

The Bell number B_n counts the number of partitions of an n-element set into nonempty unlabeled subsets. Here, each of the eight possible three-bit histories is an element and each subset collects histories assigned to the same latent state. The implementation uses restricted-growth strings to enumerate each partition exactly once, thereby eliminating duplicates caused solely by relabeling latent states.

For every partition we compute

$$I(H;Z), \qquad I(Z;Y), \qquad H(Y \mid Z)$$

exactly. A representation belongs to the deterministic Pareto frontier if there is no other deterministic representation with no larger $I(H;Z)$ and strictly larger $I(Z;Y)$. A numerical tolerance of 10^{-12} is used when constructing this frontier. Of the 4140 deterministic partitions, 36 are nondominated.

Exact compression sweep. For each compression coefficient β, every deterministic partition is scored using

$$\mathcal{L}_{\mathrm{PIB}}^{\exp} = H(Y \mid Z) + \beta I(H;Z).$$

The tested values are

$$\beta \in \{0,\ 0.0005,\ 0.001,\ 0.003,\ 0.005,\ 0.01,\ 0.012,\ 0.014,\ 0.015,\ 0.02,\ 0.03,\ 0.10,\ 0.30,\ 1.0\}.$$

At score ties, using tolerance 10^{-10}, we first select the candidate with lower $I(H;Z)$, then the candidate with fewer occupied states, and finally larger $I(Z;Y)$. This convention matters at $\beta = 0$: several deterministic representations achieve the same minimum prediction loss, so the four-state minimal sufficient representation is the *reported tie-broken representative*, not a uniquely preferred solution of prediction alone.

The exact optimum evolves as shown in Table 7. For every tested positive value through $\beta = 0.014$, the optimum is the known four-state predictive pair. At $\beta = 0.015$, the optimum switches to a two-state compression and sacrifices a small amount of predictive information. At $\beta = 1$, complete compression to a single state becomes optimal.

Learned stochastic continuation. The orange continuation curve in Fig. 4 is generated separately from the exhaustive deterministic search. Its purpose is to show how gradient optimization behaves as compression pressure is gradually increased.

Because the underlying problem is finite, the learned encoder is represented directly as a categorical table

$$q_\theta(z \mid h), \qquad h \in \{0, \dots, 7\}, \qquad z \in \{0, \dots, 7\},$$

rather than by a neural sequence encoder. This removes architectural capacity as a confound. The predictor is a second categorical table,

$$p_\omega(Y \mid Z).$$

The encoder is initialized near the overcomplete identity mapping $Z = H$. Specifically, the diagonal encoder logits are initialized to $+8$, the off-diagonal logits to -8, and independent noise of scale 10^{-3} is added to break exact symmetry. The predictor is initialized from the exact conditional distribution $p(Y \mid H)$ associated with the identity representation.

For each tested β, the model minimizes

$$\mathcal{L}_{\text{learned}} = \mathcal{L}_{\text{pred}} + \beta I(H; Z)$$

using Adam with learning rate

$$3 \times 10^{-2}.$$

Each nonzero compression stage receives 2200 gradient steps. The solution at one value of β initializes the next, so the orange curve is a *continuation path* rather than a collection of independently initialized models. At $\beta = 0$, the deliberately overcomplete eight-state initialization is retained without an optimization stage. Five independent perturbation seeds are used, and the plotted curve reports their mean with standard deviations.

This distinguishes the two $\beta = 0$ constructions in the experiment. The exact deterministic sweep reports the lower-information four-state solution after tie-breaking among equally predictive partitions, whereas the learned continuation deliberately begins from the overcomplete eight-state solution so that the compression trajectory can be observed.

The learned encoder is stochastic, whereas the exact blue frontier contains only deterministic mappings. The learned continuation is therefore interpreted *relative to* the deterministic global reference, not as an optimization method expected to lie exactly on that frontier.

Residual-predictability diagnostic. The residual diagnostic uses five independently generated sequences of length 120,000, initialized from the stationary pair-state distribution. Each sequence is divided chronologically into 70% training and 30% held-out evaluation data.

The restricted predictor estimates

$$\widehat{p}_0 = \widehat{p}(X_{t+1} = 1 \mid Z_t),$$

whereas the history-augmented predictor receives one additional step of representation history,

$$\widehat{p}_1 = \widehat{p}(X_{t+1} = 1 \mid Z_t, Z_{t-1}).$$

Both are discrete lookup estimators fitted on the training portion with Laplace smoothing parameter

$$\alpha = 1.$$

For the three controlled representations, the augmented inputs specialize to

$$\begin{aligned}
Z_t^{\text{under}} = X_t: &\qquad (Z_t, Z_{t-1}) = (X_t, X_{t-1}),\\
Z_t^{\text{suff}} = (X_{t-1}, X_t): &\qquad (Z_t, Z_{t-1}) = \big((X_{t-1}, X_t), (X_{t-2}, X_{t-1})\big),\\
Z_t^{\text{over}} = (X_{t-2}, X_{t-1}, X_t): &\qquad (Z_t, Z_{t-1}) = \big((X_{t-2}, X_{t-1}, X_t), (X_{t-3}, X_{t-2}, X_{t-1})\big).
\end{aligned}$$

This makes the positive and negative controls transparent. For $Z_t = X_t$, the augmentation $Z_{t-1} = X_{t-1}$ restores exactly the transition-relevant variable omitted from the current state. For $Z_t = (X_{t-1}, X_t)$, the only genuinely new observation supplied by Z_{t-1} is X_{t-2}, which is redundant under the data-generating process. The overcomplete representation already contains still more history, so adding its previous state should likewise provide no one-step predictive benefit.

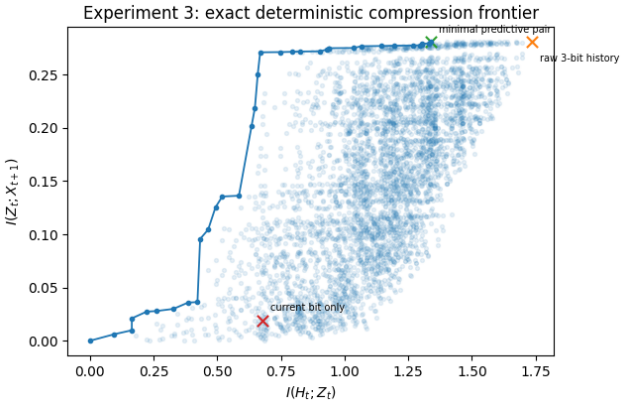


Figure 10: Exact deterministic compression frontier for Experiment 3. All 4140 deterministic partitions of the eight histories are evaluated exactly. The nondominated frontier gives the best achievable deterministic trade-offs between retaining history information and preserving information about X_{t+1}.

On the held-out portion we compute

$$\mathrm{MSE}_{\text{restricted}} = \mathbb{E}\left[(X_{t+1} - \widehat{p}_0)^2\right],$$
$$\mathrm{MSE}_{\text{history-augmented}} = \mathbb{E}\left[(X_{t+1} - \widehat{p}_1)^2\right],$$

and define

$$\Delta_{\text{hist}} = \mathrm{MSE}_{\text{restricted}} - \mathrm{MSE}_{\text{history-augmented}}.$$

Thus, positive Δ_{hist} means that Z_{t-1} contains predictive information absent from Z_t. Values near zero indicate that one further step of representation history does not improve held-out prediction.

The measured gains are

$$0.1139 \pm 0.0026$$

for the insufficient representation,

$$-6.8 \times 10^{-6}$$

for the minimal sufficient representation, and

$$-3.7 \times 10^{-5}$$

for the overcomplete representation. The tiny negative values are finite-sample fitting variation and are effectively zero at the scale of the experiment.

Operationally, this diagnostic measures the held-out gain from adding Z_{t-1} rather than fitting a separate neural regressor to residuals. It realizes the same sufficiency principle used in the main text: if omitted history still carries transition-relevant information, augmenting the predictor with an earlier representation state should reduce held-out prediction error.

Supplementary results. Figure 10 shows the exact deterministic frontier without the learned continuation overlay.

Figure 11 shows the number of occupied hard states along the compression sweep for both the exact deterministic optimum and the learned continuation. The learned warm-started path can differ from the global deterministic optimum because of stochastic parameterization and optimization path dependence.

J.5 Experiment 4: HMM-style training of PIB-VJEPA

Independent rerun with a shared latent-state family. Experiment 4 is implemented and run independently from Experiments 1–3. It regenerates the same separated-emission data used in

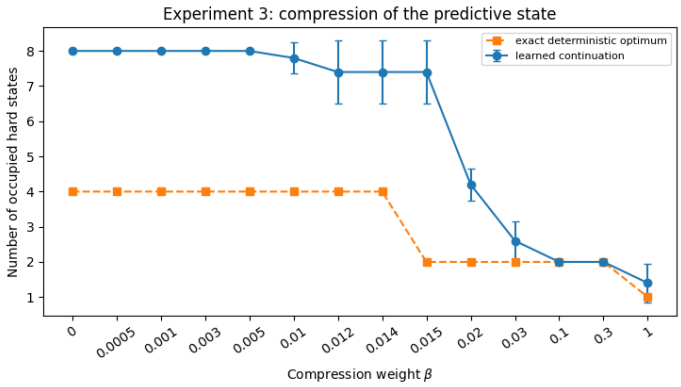


Figure 11: Number of occupied hard states as the compression weight β increases. The dashed reference gives the exact deterministic optimum, while the learned stochastic continuation follows the warm-started gradient trajectory.

Experiment 1 using the same ground-truth transition matrix, Gaussian state means, emission standard deviation

$$\sigma = 0.35,$$

and data seeds. For top-level seed $s \in \{0, \ldots, 4\}$, the training generator uses

$$1000 + 17s,$$

and the test generator uses

$$2000 + 17s.$$

Consequently, Experiment 4 sees the same paired data realizations as the separated condition of Experiment 1, but none of the fitted Experiment 1 models is reused. All three Experiment 4 regimes are trained afresh.

Each seed contains 400 training sequences and 160 test sequences of length 80. All regimes use $K = 4$ categorical latent states and the same basic one-layer GRU context-encoder family with hidden dimension 48.

Common initialization and state-use regularization. All categorical encoder regimes use an unsupervised K-means warm start with $K = 4$ and 10 K-means initializations; ground-truth states are never used. The context encoder is pretrained for 100 updates at learning rate

$$10^{-2}$$

to reproduce the K-means assignments.

The Gaussian HMM components in the HMM+filter and hybrid regimes are initialized from the same K-means partition. State-conditional means are initialized from cluster centers, diagonal variances from within-cluster variances plus 0.05, the initial-state logits are initialized uniformly, and the transition logits receive a diagonal bias

$$L_A = 1.5 I_4.$$

The same state-use regularization is applied to all three amortized context encoders:

$$\mathcal{L}_{\text{state}} = 0.30\, \mathcal{L}_{\text{occ}} + 0.02\, \mathcal{L}_{\text{ent}}.$$

Sharing these coefficients removes state-regularization strength as a confound in the objective-level comparison.

JEPA latent regime. The JEPA-only regime is trained from scratch using the same MCJEPA construction and optimization settings as the shared-A model in Experiment 1. Its objective is

$$\mathcal{L}_{\text{JEPA}} = \mathcal{L}_{\text{MC}} + \mathcal{L}_{\text{state}},$$

where

$$\mathcal{L}_{\mathrm{MC}} = \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} \mathbb{E}_t \left[\mathrm{KL}\left(\mathrm{sg}\big(q_{\bar{\theta}}(Z_{t+h} \mid X_{t+h})\big) \,\middle\|\, q_\theta(Z_t \mid X_{\le t}) A^h \right) \right].$$

The context encoder is trained for 180 epochs with mini-batches of 64 sequences, Adam learning rate

$$3 \times 10^{-3},$$

EMA coefficient

$$\tau = 0.995,$$

and gradient clipping at norm 5.

No observation model participates in this training. To evaluate observation-sequence NLL afterward, we fit a diagonal Gaussian emission distribution to each learned latent state using the soft training assignments:

$$\widehat{\mu}_k = \frac{\sum_{n,t} q_{nt}(k) X_{nt}}{\sum_{n,t} q_{nt}(k)}.$$

The diagonal variance is the corresponding soft-assignment-weighted second moment around $\widehat{\mu}_k$, with a minimum variance of 0.03. The initial-state distribution is estimated from the mean encoder distribution at the first time step. These post-hoc parameters do not backpropagate into either the JEPA encoder or transition matrix.

The fitted observation model and learned transition are then treated as a fixed HMM solely for evaluation of test sequence NLL and the post-hoc filtering diagnostic.

HMM sequence + filter-distillation regime. The second regime jointly maintains a Gaussian HMM and an amortized GRU context encoder but contains no JEPA latent-prediction loss. The HMM contributes

$$\mathcal{L}_{\mathrm{HMM}} = -\frac{1}{T} \mathbb{E} \left[\log p_{\phi,\psi}(X_{1:T}) \right].$$

At each update, its current exact filtering posterior is computed and detached,

$$\widetilde{q}_t = \mathrm{sg}\left(p_{\phi,\psi}(Z_t \mid X_{\le t}) \right),$$

and the context encoder is trained through

$$\mathcal{L}_{\mathrm{filter}} = \mathbb{E}_t \left[\mathrm{KL}\left(\widetilde{q}_t \,\|\, q_\theta(Z_t \mid X_{\le t}) \right) \right].$$

The implemented objective is

$$\mathcal{L}_{\mathrm{HMM+filter}} = 1.0\, \mathcal{L}_{\mathrm{HMM}} + 0.5\, \mathcal{L}_{\mathrm{filter}} + 0.30\, \mathcal{L}_{\mathrm{occ}} + 0.02\, \mathcal{L}_{\mathrm{ent}}.$$

The HMM and encoder are jointly optimized for 600 full-data updates using Adam with learning rate

$$10^{-2}.$$

Gradients are clipped to norm 10. Because the filtering target is detached, $\mathcal{L}_{\mathrm{filter}}$ updates the amortized encoder but not the HMM parameters. Likewise, $\mathcal{L}_{\mathrm{state}}$ acts only on the encoder. Thus, the generative transition and emission parameters are learned through sequence likelihood, while the context encoder learns to amortize the corresponding Bayesian filtering operation.

Hybrid HMM + latent regime. The hybrid uses the same Gaussian HMM and amortized context-encoder families as the preceding regime but adds the genuine MCJEPA latent-prediction objective. A separate target encoder is initialized from the context encoder after K-means pretraining and subsequently updated only by EMA.

Its three principal losses are

$$\begin{aligned}
\mathcal{L}_{\mathrm{HMM}} &= -\frac{1}{T} \mathbb{E} \left[\log p_{\phi,\psi}(X_{1:T}) \right], \\
\mathcal{L}_{\mathrm{filter}} &= \mathbb{E}_t \left[\mathrm{KL}\left(\mathrm{sg}(\widetilde{q}_t) \,\|\, q_\theta(Z_t \mid X_{\le t}) \right) \right], \\
\mathcal{L}_{\mathrm{MC}} &= \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} \mathbb{E}_t \left[\mathrm{KL}\left(\mathrm{sg}\big(q_{\bar{\theta}}(Z_{t+h} \mid X_{t+h})\big) \,\middle\|\, q_\theta(Z_t \mid X_{\le t}) A^h \right) \right].
\end{aligned}$$

Table 8: Effective parameter updates in the revised Experiment 4 implementation. The target encoder receives no gradient and is updated only by EMA.

Loss	Context encoder	Transition A	Emission / initial-state parameters
$\mathcal{L}_{\mathrm{HMM}}$	–	✓	✓
$\mathcal{L}_{\mathrm{MC}}$	✓	✓	–
$\mathcal{L}_{\mathrm{filter}}$	✓	–	–
$\mathcal{L}_{\mathrm{state}}$	✓	–	–

The critical implementation distinction is that the HMM filtering posterior

$$\widetilde{q}_t = p_{\phi,\psi}(Z_t \mid X_{\leq t})$$

is used *only* by $\mathcal{L}_{\mathrm{filter}}$. It is not substituted for the future JEPA target in $\mathcal{L}_{\mathrm{MC}}$. Instead, the latter uses the same EMA local-target construction as the JEPA-only baseline. The hybrid comparison is therefore objective-faithful: its $\mathcal{L}_{\mathrm{MC}}$ remains the original JEPA latent-prediction signal.

The transition matrix A is shared between the HMM and JEPA objectives. Hence the same latent dynamics are trained simultaneously by observation-sequence evidence and latent predictive alignment. The complete implemented objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{hybrid}} = 1.0\,\mathcal{L}_{\mathrm{HMM}} + 1.0\,\mathcal{L}_{\mathrm{MC}} + 0.5\,\mathcal{L}_{\mathrm{filter}} \\ + 0.30\,\mathcal{L}_{\mathrm{occ}} + 0.02\,\mathcal{L}_{\mathrm{ent}}.\end{aligned}$$

The model is trained for 600 full-data updates with Adam learning rate

$$10^{-2},$$

EMA coefficient

$$\tau = 0.995,$$

and gradient clipping at norm 10.

Which parameters are trained by each objective? For clarity, Table 8 summarizes the effective parameter flow. Both the exact filtering target and EMA JEPA target are stop-gradient quantities.

This separation clarifies the interpretation of the comparison. In HMM+filter training, the generative parameters are learned from sequence evidence and the context encoder amortizes the resulting filter. In hybrid training, the transition additionally receives the JEPA latent-prediction signal, while the emission and initial-state parameters remain trained through sequence evidence.

Forward algorithm and numerical stabilization. All HMM sequence likelihoods are evaluated exactly in log space. Let

$$\ell_t(k) = \log p_\psi(X_t \mid Z_t = k).$$

The forward recursion is initialized as

$$\alpha_1(k) = \log \pi_k + \ell_1(k)$$

and updated by

$$\alpha_t(j) = \ell_t(j) + \mathrm{logsumexp}_i \left[\alpha_{t-1}(i) + \log A_{ij}\right].$$

The sequence log-likelihood is

$$\log p(X_{1:T}) = \mathrm{logsumexp}_k\, \alpha_T(k).$$

Learned HMM log variances are clamped to

$$[-7, 4]$$

before evaluating Gaussian emissions. A numerical floor

$$\varepsilon = 10^{-8}$$

is used when taking logarithms or normalizing probabilities.

Table 9: Paired observation-sequence NLL difference relative to HMM+filter training. Differences are computed within each seed before aggregation. Lower is better.

Training regime	Δ sequence NLL / step
HMM sequence + filter distill	0
Hybrid HMM + latent	0.000134 ± 0.000092
JEPA latent objective	0.009409 ± 0.003164

Table 10: True-state multi-horizon prediction NLL in Experiment 4. Values are mean $\pm$ standard deviation over five seeds. Lower is better.

Training regime	$h = 1$	$h = 2$	$h = 4$	$h = 8$
HMM sequence + filter distill	$\mathbf{0.5271 \pm 0.0027}$	$\mathbf{0.7983 \pm 0.0045}$	$\mathbf{1.0939 \pm 0.0054}$	1.3035 ± 0.0047
Hybrid HMM + latent	0.5295 ± 0.0030	0.7996 ± 0.0048	1.0943 ± 0.0057	$\mathbf{1.3029 \pm 0.0051}$
JEPA latent objective	0.5437 ± 0.0019	0.8106 ± 0.0040	1.1004 ± 0.0056	1.3049 ± 0.0049

Evaluation protocol. The common metrics and Hungarian alignment follow Section J.1. For every regime, the alignment is obtained from training-set hard assignments and held fixed during test evaluation. We report test ARI and NMI, transition error $\mathcal{E}_A$, observation-sequence NLL per time step, filtering KL, and true-state prediction NLL at

$$h \in \{1, 2, 4, 8\}.$$

The three groups of metrics have distinct interpretations. ARI and NMI evaluate representation recovery; transition error and true-state NLL evaluate learned latent dynamics; sequence NLL evaluates the complete transition–emission model. Filtering KL is treated separately because it is an explicitly optimized quantity for two of the three regimes.

Paired sequence-evidence comparison. All three regimes within a seed are evaluated on exactly the same test realization. The sequence-evidence figure therefore uses a paired difference. For regime r and seed s, we compute

$$\Delta\mathrm{NLL}_{r,s} = \mathrm{NLL}_{r,s} - \mathrm{NLL}_{\mathrm{HMM+filter},s}$$

before averaging across seeds. This removes variability caused by different sampled test sequences and makes the objective-induced difference easier to see.

Thus, the hybrid retains only a very small sequence-evidence gap relative to HMM-style training, whereas the latent-only JEPA model remains clearly separated. We use this paired comparison descriptively rather than as a formal hypothesis test.

Multi-horizon prediction. For each regime we propagate the current context distribution through powers of its learned transition matrix,

$$\widehat{q}_{t+h} = q_t A^h,$$

align the result to ground-truth state order, and evaluate true-state NLL. The complete results are shown in Table 10.

Both HMM-style and hybrid training improve over JEPA-only at every evaluated horizon. The differences are largest at shorter horizons and narrow by $h = 8$, where repeated application of the transition matrix increasingly mixes the predictive state distribution.

Filtering agreement. Filtering KL is defined in Section J.1. The exact reference is the filtering distribution implied by the probabilistic model associated with each regime. For HMM+filter and hybrid training, this is the exact filter of their jointly trained Gaussian HMM. For JEPA-only, it is the filter obtained after fitting the post-hoc Gaussian observation model to the learned JEPA states.

The resulting values are

$$0.000060 \pm 0.000007$$

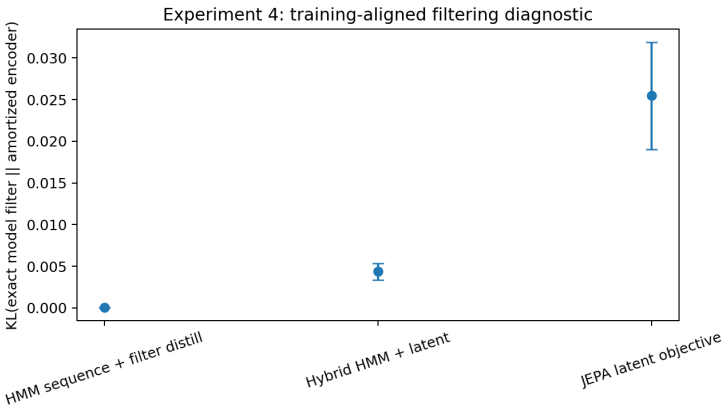


Figure 12: Filtering agreement across the three Experiment 4 training regimes. The quantity is $\mathrm{KL}(q_t^{\mathrm{exact}} \| q_t^{\mathrm{enc}})$ averaged over test time points. HMM+filter and hybrid training explicitly optimize filtering alignment, so their values should be interpreted as training-aligned role diagnostics. JEPA-only is evaluated post hoc using the filtering distribution induced by its fitted observation model.

for HMM+filter,

$$0.004355 \pm 0.001015$$

for the hybrid, and

$$0.025460 \pm 0.006423$$

for JEPA-only.

These quantities do not compare every model with one common external filtering oracle. Moreover, filtering KL is explicitly optimized for HMM+filter and hybrid training, so for those regimes it is a *training-aligned role diagnostic*. For JEPA-only it is instead a post-hoc diagnostic of how closely the learned context representation happens to agree with the filtering distribution induced by its fitted observation model.

Seed-wise consistency. The aggregate improvement of hybrid training over JEPA-only is not produced by one favorable seed. For each of the five paired runs, hybrid training improves over JEPA-only in ARI, NMI, observation-sequence NLL, transition error, filtering KL, and true-state prediction NLL at every evaluated horizon $h \in \{1, 2, 4, 8\}$. We report this pattern descriptively and do not infer formal statistical significance from five seeds.

Supplementary filtering diagnostic. Figure 12 reports filtering agreement separately from the two generative-model metrics emphasized in the main text.

J.6 Reproducibility and computation

Code implementing all four experiments will be released upon publication; the repository link is withheld to preserve anonymity during review.

Five fixed seeds

$$\{0, 1, 2, 3, 4\}.$$

were used. The implementation uses single-precision PyTorch tensors,

$$\texttt{torch.float32},$$

and automatically selects CUDA when available, otherwise falling back to CPU.

Experiments 1–3. The principal paper-mode settings are

$$N_{\mathrm{train}} = 400, \qquad N_{\mathrm{test}} = 160, \qquad T = 80, \qquad d_{\mathrm{hidden}} = 48, \qquad B = 64.$$

The main optimization budgets are

$$180 \text{ MCJEPA epochs}, \qquad 100 \text{ MCJEPA warm-start steps}, \qquad 350 \text{ Gaussian-HMM steps},$$

and

$$2200 \text{ updates per nonzero Experiment 3 compression stage.}$$

Experiment 3 uses five independent perturbation seeds for the learned information-bottleneck continuation and five independently generated long sequences for the residual diagnostic.

Experiment 4. The standalone Experiment 4 script uses

$$N_{\text{train}} = 400, \qquad N_{\text{test}} = 160, \qquad T = 80, \qquad d_{\text{hidden}} = 48.$$

The JEPA-only baseline uses 180 epochs with batch size 64, learning rate 3×10^{-3}, 100 K-means warm-start updates, and EMA coefficient 0.995. The HMM+filter and hybrid regimes each use 600 full-data joint updates with Adam learning rate

$$10^{-2}.$$

Their common objective coefficients are

$$\lambda_{\text{seq}} = 1.0, \qquad \lambda_{\text{latent}} = 1.0, \qquad \lambda_{\text{filter}} = 0.5,$$

where λ_{latent} applies only to the hybrid, together with

$$\lambda_{\text{occ}} = 0.30, \qquad \lambda_{\text{ent}} = 0.02.$$

The hybrid EMA coefficient is

$$\tau = 0.995.$$

The JEPA-only model is initialized with top-level seed s, while the independently trained HMM+filter and hybrid regimes use deterministic seed offsets associated with s so that each run remains reproducible while avoiding accidental reuse of identical parameter initialization streams.

Because the JEPA-only objective is naturally optimized with sequence mini-batches whereas the differentiable HMM sequence objective is evaluated on the full training collection in the HMM+filter and hybrid implementations, Experiment 4 should be interpreted as an *objective-behavior diagnostic*, not as a compute-matched optimization-efficiency benchmark. The latent-state family, data realization, context-encoder family, state-use regularization, and evaluation protocol are controlled across regimes.

Randomness and numerical reproducibility. Randomness is seeded for Python's `random` module, NumPy, PyTorch, and all available CUDA devices. The data generators use fixed seed offsets so that training data, test data, collapse diagnostics, Experiment 2 filtering datasets, Experiment 3 continuation runs, residual-diagnostic sequences, and Experiment 4 model initializations can be reproduced independently from the top-level seed.

We do not enforce PyTorch deterministic-algorithm mode, so exact bitwise reproducibility across different CUDA libraries or hardware is not guaranteed. The scripts do not record the specific accelerator model or wall-clock runtime, and package versions are not pinned in the experimental source; we therefore do not report hardware-specific timing claims. The implementation depends on NumPy, Pandas, PyTorch, scikit-learn, SciPy, and Matplotlib.

K Extended Experimental Results and Discussion

This appendix reproduces the complete discussion of the four experiments summarized in Section 7, including all reported statistics. Tables and figures shown in the main text are referenced rather than repeated.

K.1 Experiment 1: finite-HMM recovery and Markov composition

We first test the basic MCJEPA construction on a finite HMM with four latent states and continuous observations. We consider a separated-emission regime, in which observations are highly informative about the state, and an ambiguous-emission regime, in which state inference becomes more difficult. MCJEPA uses a categorical history encoder and one shared row-stochastic transition matrix A, yielding

$$\widehat{q}_{t+h} = q_t A^h.$$

We compare it with a horizon-specific categorical JEPA that learns an independent A_h for each prediction horizon, and with a correctly specified Gaussian HMM.

Recovering states and transitions. Table 1 reports ARI, aligned transition error $\mathcal{E}_A$, and true-state prediction NLL at multiple horizons. In the separated regime, all three methods recover the latent process well. MCJEPA attains

$$\mathrm{ARI} = 0.9796 \pm 0.0055$$

and transition error

$$\mathcal{E}_A = 0.0191 \pm 0.0019,$$

close to the correctly specified HMM.

The ambiguous regime exposes the structural trade-off more clearly. The HMM remains strongest because it explicitly models the correct emission family and performs probabilistic filtering. The horizon-specific predictor obtains somewhat better state recovery than MCJEPA, but its aligned one-step transition error is

$$0.2410 \pm 0.0029,$$

compared with MCJEPA's

$$0.0901 \pm 0.0091.$$

Thus, independently fitting each horizon provides additional predictive flexibility but yields a substantially less faithful underlying transition law.

Importantly, these results do not imply that the shared-matrix constraint universally minimizes predictive NLL. Under ambiguity, the independently parameterized A_h model is slightly better at several horizons. The benefit of MCJEPA is instead structural: all horizons are generated by one transition mechanism and must therefore compose consistently.

Exact path composition. The left panel of Fig. 2 illustrates the consequence of using a shared Markov transition. To quantify disagreement between a direct prediction and a composed prediction with the same total horizon, we define

$$\mathcal{D}_{\mathrm{path}}(h_1, h_2) = \mathbb{E}_t \left[\| q_t A_{h_1+h_2} - (q_t A_{h_1}) A_{h_2} \|_1 \right], \tag{47}$$

where the expectation is taken over valid evaluation time points. Thus, $\mathcal{D}_{\mathrm{path}} = 0$ means that the direct and composed predicted state distributions agree exactly.

The labels $1+1$, $2+2$, and $4+4$ denote three decompositions of the same total prediction horizon. Specifically, $1+1$ compares a direct two-step prediction with two successive one-step predictions,

$$q_t A_2 \quad \text{versus} \quad (q_t A_1) A_1,$$

$2+2$ compares a direct four-step prediction with two successive two-step predictions,

$$q_t A_4 \quad \text{versus} \quad (q_t A_2) A_2,$$

and $4+4$ compares a direct eight-step prediction with two successive four-step predictions,

$$q_t A_8 \quad \text{versus} \quad (q_t A_4) A_4.$$

For MCJEPA, $A_h = A^h$, so

$$q_t A^{h_1+h_2} = (q_t A^{h_1}) A^{h_2}$$

exactly, and hence $\mathcal{D}_{\mathrm{path}} = 0$ by construction. Independently learned horizon-specific matrices A_h, however, are not constrained to satisfy these composition identities. In the ambiguous regime, their path disagreement is

$$0.2138 \pm 0.0041, \qquad 0.1321 \pm 0.0053, \qquad 0.0645 \pm 0.0072$$

for the $1+1$, $2+2$, and $4+4$ decompositions, respectively. The corresponding separated-regime disagreement is smaller but remains nonzero, confirming that independently trained horizon predictors need not define one coherent Markov chain.

Table 11: Collapse ablation in Experiment 1. Values are mean ± standard deviation over 5 seeds. $K_{\text{eff}}^{\text{hard}}$ is the effective number of states induced by hard assignments. Assignment entropy measures per-example confidence and should be interpreted jointly with state usage: very low entropy can also arise from single-state collapse. The occupancy and entropy terms are complementary: the former encourages global state use, while the latter encourages confident per-example assignments.

Regularization	ARI ↑	$K_{\text{eff}}^{\text{hard}}$	Assignment entropy
Both	$\mathbf{0.9748 \pm 0.0134}$	3.9551 ± 0.0084	0.0716 ± 0.0987
Occupancy only	0.6582 ± 0.0736	3.8092 ± 0.2249	1.1037 ± 0.1216
Entropy only	0.0776 ± 0.1734	1.1703 ± 0.3807	0.0015 ± 0.0034
None	0.5626 ± 0.1612	3.4007 ± 0.5026	1.1062 ± 0.0837

Preventing discrete-state collapse. We next ablate the occupancy and entropy components of the state-use regularizer $\mathcal{L}_{\text{state}}$ in Eq. (7). Table 11 shows that the two terms address complementary failure modes. Using both gives

$$\text{ARI} = 0.9748 \pm 0.0134$$

and approximately four effective hard states. Occupancy regularization alone maintains broad state use but leaves assignments highly uncertain, with mean assignment entropy

$$1.1037 \pm 0.1216.$$

Entropy regularization alone instead makes assignments confident but usually collapses them onto a single state: four of the five runs use one effective state, and the average effective hard-state count is only

$$1.1703 \pm 0.3807.$$

Using neither term produces less severe but unstable state use and substantially weaker recovery.

The right panel of Fig. 2 visualizes the ARI column of Table 11. Together, the state-recovery and collapse diagnostics support our intended interpretation: occupancy prevents global state under-use, whereas the entropy term prevents diffuse per-sample assignments; both are needed to obtain confident and diverse state assignments without collapse.

The complete multi-horizon curves, the separated-regime path-consistency result, and the state-usage and assignment-confidence ablations are provided in Appendix J.

K.2 Experiment 2: filtering resolves emission ambiguity

Experiment 2 isolates the distinction between local observation evidence and filtering. We use a persistent two-state HMM whose emission distributions are made progressively more overlapping[4]. Because the generating model is known, we can compute both

$$\text{local evidence: } p(S_t \mid X_t)$$

and

$$\text{filtering: } p(S_t \mid X_{\leq t})$$

exactly. In MCJEPA terms, these two quantities are the oracle counterparts of a local encoder $q_\theta(Z_t \mid X_t)$ and a history-dependent encoder $q_\theta(Z_t \mid X_{\leq t})$, respectively. We use the exact posteriors here rather than learned encoders so that the experiment isolates the informational value of observation history without representation-learning or optimization confounds; it is therefore not a comparison between an HMM and MCJEPA.

Figure 3 shows that the oracle filtering distribution becomes increasingly more informative than the oracle local-evidence distribution as individual observations become ambiguous. At the most overlapping setting, local evidence reaches state accuracy

$$0.6730 \pm 0.0037,$$

[4]The two states have Gaussian emissions centered at $-\mu$ and $+\mu$ with common standard deviation σ. We control emission ambiguity through the separation ratio $\mu/\sigma \in \{2.0, 1.25, 0.75, 0.45\}$; decreasing μ/σ increases the overlap between the two emission distributions and therefore makes the current observation less informative about the latent state.

whereas filtering reaches

$$0.8433 \pm 0.0080.$$

The corresponding state NLL decreases from

$$0.6022 \pm 0.0030$$

to

$$0.3683 \pm 0.0088.$$

The advantage diminishes as the emissions become locally separable, as expected.

The right panel of Fig. 3 illustrates the mechanism around a true state transition. Local evidence fluctuates strongly with individual observations. Filtering instead combines the current observation with the propagated state belief, remaining stable through many locally ambiguous measurements and changing when the accumulated evidence supports a transition. This directly supports the probabilistic distinction made earlier in the paper: $p(S_t \mid X_t)$ is local observation evidence, whereas $p(S_t \mid X_{\leq t})$ is the HMM filtering belief.

The corresponding NLL curve is provided in Appendix J.

K.3 Experiment 3: predictive compression and Markovization

Experiment 3 asks a simple question: can predictive compression discard unnecessary history while retaining exactly the information needed to predict the future?

We construct a binary second-order process satisfying

$$p(X_{t+1} \mid X_{\leq t}) = p(X_{t+1} \mid X_{t-1}, X_t). \tag{48}$$

Thus, although the entire observation history is available, only the two most recent observations are needed to predict X_{t+1}.

We deliberately give the encoder a longer three-step history,

$$H_t = (X_{t-2}, X_{t-1}, X_t), \tag{49}$$

which has eight possible values. The known minimal predictive state is

$$Z_t^\star = (X_{t-1}, X_t), \tag{50}$$

which has only four possible values. The older bit X_{t-2} is therefore redundant once $Z_t^\star$ is known.

This construction gives us a controlled ground truth for what predictive compression should do:

$$\underbrace{(X_{t-2}, X_{t-1}, X_t)}_{\text{too much history}} \longrightarrow \underbrace{(X_{t-1}, X_t)}_{\text{just enough}} \longrightarrow \underbrace{X_t}_{\text{too little}}.$$

In MCJEPA terms, these are three controlled choices of latent state supplied to the predictor: an overcomplete state $Z_t = H_t$, the minimal sufficient state $Z_t = Z_t^\star$, and an insufficient state $Z_t = X_t$.

The middle representation also explains the term *Markovization*. Although the observation process is second-order in X_t, defining

$$Z_t^\star = (X_{t-1}, X_t)$$

turns it into a first-order state process: the information needed for the next transition is contained in the current state $Z_t^\star$, without requiring older history. Predictive compression should therefore remove X_{t-2}, but should not remove X_{t-1}.

Does compression recover the correct predictive state? We first compare the three controlled representations exactly. Because the process is finite, their information quantities and optimal one-step prediction losses can be computed without representation-learning or optimization error.

Table 2 gives the key result. The full three-bit history retains

$$\mathrm{I}(H_t; Z_t) = 1.7356,$$

whereas the four-state predictive pair retains only

$$\mathrm{I}(H_t; Z_t) = 1.3378.$$

Despite this compression, the two representations contain exactly the same information about the next observation,

$$\mathrm{I}(Z_t; X_{t+1}) = 0.2807,$$

and achieve the same prediction NLL,

$$0.3978.$$

Hence, removing X_{t-2} reduces the amount of past information stored in the state without sacrificing prediction.

Compressing further to $Z_t = X_t$, however, removes information that is genuinely needed. Predictive information falls from 0.2807 to 0.0192, and prediction NLL increases from 0.3978 to 0.6593. The controlled construction therefore has a known sufficiency–minimality boundary: eight states are predictively sufficient but redundant, four states are sufficient and minimal for this process, and two states are insufficient.

Is the four-state solution truly optimal, or just a favorable example? The comparison above considers only three hand-specified representations. We therefore use the small history space to perform an exhaustive check over *every deterministic compression* of the eight possible histories represented by H_t.

A deterministic encoder

$$Z_t = f(H_t)$$

groups histories that are assigned to the same latent state. We enumerate all such groupings and evaluate each one using

$$\mathcal{L}_{\mathrm{PIB}}^{\mathrm{exp}} = \mathcal{L}_{\mathrm{pred}} + \beta\, \mathrm{I}(H_t; Z_t), \tag{51}$$

where the first term rewards accurate prediction and the second penalizes retaining unnecessary information about the history.

Because there are only eight possible histories, all 4140 deterministic partitions can be enumerated exactly. This provides a global deterministic reference rather than relying on a few hand-designed candidates. For every tested positive compression weight

$$0 < \beta \leq 0.014,$$

the globally optimal deterministic representation is exactly the known four-state predictive state

$$Z_t^\star = (X_{t-1}, X_t).$$

Thus, when compression is strong enough to penalize redundant history but not so strong that predictive information is sacrificed, the predictive-bottleneck objective selects the known minimal sufficient Markov state.

At $\beta = 0$, several predictively equivalent deterministic representations attain the same minimum prediction loss; the four-state state is therefore not identified by prediction alone. Once $\beta > 0$, however, redundant stored history is penalized. At $\beta = 0.015$, the deterministic optimum changes to a two-state representation. Its retained predictive information decreases from 0.2807 to 0.2711, indicating that compression has begun to remove information useful for prediction. With still stronger compression, the optimum eventually collapses to a single state. The resulting progression is therefore

$$\text{redundant representation} \longrightarrow \text{minimal predictive state} \longrightarrow \text{over-compressed state}.$$

The left panel of Fig. 4 visualizes the compression–prediction trade-off in two complementary ways. Each light-blue point corresponds to one of the 4140 deterministic partitions of the eight possible histories, positioned according to the amount of history information it retains, $\mathrm{I}(H_t; Z_t)$ on the horizontal axis, and the amount of predictive information it preserves, $\mathrm{I}(Z_t; X_{t+1})$ on the vertical axis. The blue curve connects the nondominated deterministic solutions and therefore gives the exact deterministic reference frontier.

The orange curve is obtained differently. We initialize a stochastic encoder at the overcomplete eight-history representation and follow a warm-started continuation path as the compression weight β is increased. Each orange marker shows the representation obtained at one value of β. At $\beta = 0$, the overcomplete initialization is retained with no compression pressure. For subsequent values, increasing β makes representations with smaller $\mathrm{I}(H_t; Z_t)$ increasingly preferable. The learned

solution is therefore encouraged to move leftward in the information plane. Ideally, this removes redundant history while remaining near the top of the plot, where predictive information is preserved. If β becomes too large, however, compression also removes information needed for prediction and the trajectory moves downward.

The labeled β values do not represent different data-generating processes; they are different settings of the same predictive-compression objective and trace how the learned representation changes as compression pressure increases. The learned trajectory need not coincide with the exact blue frontier because the encoder is stochastic and is optimized by gradient descent, whereas the blue frontier is obtained by exhaustive enumeration over deterministic partitions. We therefore use the deterministic frontier as a global reference and the orange continuation path as a practical illustration of how predictive compression behaves during learning. Full enumeration details, the complete β sweep, continuation optimization settings, and the corresponding state-count trajectories are provided in Appendix J.

Did compression remove too much information? The previous results identify which representations achieve a favorable trade-off between compression and prediction. We next ask a complementary question: can we detect when compression has gone too far and removed information that is still useful for predicting the future?

We instantiate the residual-history diagnostic from Section 6 using one additional step of representation history. The intuition is simple. If Z_t already contains all information needed to predict X_{t+1}, then additionally conditioning on the previous representation state Z_{t-1} should not improve held-out prediction. Conversely, if Z_{t-1} still contains transition-relevant information that is absent from Z_t, then the current representation is predictively insufficient.

We test the same three controlled representations:

$$Z_t^{\text{over}} = (X_{t-2}, X_{t-1}, X_t), \qquad Z_t^{\text{suff}} = (X_{t-1}, X_t), \qquad Z_t^{\text{under}} = X_t.$$

For each representation, we compare two predictors of X_{t+1}. The *restricted* predictor uses only the current representation,

$$\widehat{p}_{\text{restricted}} = \widehat{p}(X_{t+1} = 1 \mid Z_t),$$

whereas the *history-augmented* predictor additionally receives the previous representation state,

$$\widehat{p}_{\text{history-augmented}} = \widehat{p}(X_{t+1} = 1 \mid Z_t, Z_{t-1}).$$

We define the residual history gain as

$$\Delta_{\text{hist}} = \text{MSE}_{\text{restricted}} - \text{MSE}_{\text{history-augmented}}. \tag{52}$$

Thus,

$$\Delta_{\text{hist}} > 0$$

means that the previous representation state contains predictive information not already captured by Z_t. By contrast,

$$\Delta_{\text{hist}} \approx 0$$

means that adding one further step of representation history provides essentially no additional predictive benefit. The lookup predictors, chronological train–test split, and fitting procedure are detailed in Appendix J.

The three controlled representations make this diagnostic especially transparent. For the insufficient representation

$$Z_t^{\text{under}} = X_t,$$

the previous representation is simply

$$Z_{t-1}^{\text{under}} = X_{t-1}.$$

Thus, the history-augmented predictor restores exactly the variable omitted from Z_t that is required by the second-order transition law. As expected, this produces a substantial held-out prediction gain,

$$\Delta_{\text{hist}} = 0.1139 \pm 0.0026.$$

This is the intended positive control: $Z_t = X_t$ is insufficient because the omitted X_{t-1} remains informative about X_{t+1}.

For the minimal sufficient representation,

$$Z_t^{\text{suff}} = (X_{t-1}, X_t),$$

we have

$$Z_{t-1}^{\text{suff}} = (X_{t-2}, X_{t-1}).$$

Since X_{t-1} is already contained in Z_t, the only genuinely additional observation supplied by Z_{t-1} is X_{t-2}, which is redundant for predicting X_{t+1} by construction. Correspondingly,

$$\Delta_{\text{hist}} \approx -6.8 \times 10^{-6}.$$

For the overcomplete representation,

$$Z_t^{\text{over}} = (X_{t-2}, X_{t-1}, X_t),$$

the previous representation

$$Z_{t-1}^{\text{over}} = (X_{t-3}, X_{t-2}, X_{t-1})$$

adds only still older information beyond what is already available in Z_t. We obtain

$$\Delta_{\text{hist}} \approx -3.7 \times 10^{-5}.$$

Both near-zero values are negligible at the scale of the experiment. The tiny negative values are attributable to finite-sample fitting variation rather than a meaningful advantage of the restricted predictor. Once the current representation already contains all transition-relevant information, adding Z_{t-1} does not improve held-out prediction.

Viewed together, the three controlled cases reveal a clear sufficiency–minimality boundary. Compressing from the overcomplete eight-state representation to the four-state minimal representation reduces $\mathrm{I}(H_t; Z_t)$ from 1.7356 to 1.3378 while leaving Δ_{hist} effectively zero, indicating that redundant history has been removed without sacrificing predictive sufficiency. Compressing further to the two-state representation reduces $\mathrm{I}(H_t; Z_t)$ to 0.6785, but Δ_{hist} rises sharply to 0.1139 ± 0.0026: the previous representation $Z_{t-1} = X_{t-1}$ now contains substantial transition-relevant information missing from $Z_t = X_t$. Thus, in this controlled example, the four-state representation lies at the natural elbow between retaining redundant history and compressing away information required for prediction.

This result highlights the distinction between *sufficiency* and *minimality*. The residual-history diagnostic tests sufficiency: it correctly identifies $Z_t = X_t$ as missing predictive information, while both the four-state and eight-state representations pass because their current state already contains all information required for one-step prediction. The diagnostic cannot, however, determine that the eight-state representation stores redundant history. The predictive bottleneck supplies this complementary notion of minimality by preferring the smaller four-state representation among predictively sufficient alternatives.

Together, the two diagnostics play complementary roles. Predictive compression asks how much of the past can be discarded while preserving future prediction, thereby favoring a compact Markov state. Residual predictability asks whether compression has discarded too much: a positive Δ_{hist} indicates that the previous representation state contains transition-relevant information not already captured by Z_t. In this controlled process, the four-state representation is the known minimal sufficient target: it preserves all one-step predictive information while storing less history than the overcomplete eight-state representation. Detailed enumeration, optimization settings, complete compression sweeps, residual-predictor specifications, and supplementary plots are provided in Appendix J.

K.4 Experiment 4: HMM-style training of PIB-VJEPA

Experiment 4 examines the final and strongest level of correspondence considered in this paper: *model-and-objective equivalence*. Even when JEPA and HMM formulations share an emission-complete latent-state representation, or satisfy the conditions for sequence-level HMM equivalence, they need not be trained by the same objective. Standard JEPA training optimizes prediction in

latent space, whereas HMM training additionally optimizes the probability of the observed sequence through an explicit transition–emission model.

To isolate this distinction, we use the same four-state latent family and separated-emission data-generating process as Experiment 1, but train all three Experiment 4 regimes independently. Because the latent state is categorical and the observation model is Gaussian, observation-sequence likelihood and filtering posteriors can be evaluated exactly using the HMM forward recursion. We compare three regimes—JEPA-only, HMM-style, and their hybrid—while keeping the latent-state family and amortized context-encoder architecture fixed.

Training regimes. The first regime is the *JEPA latent objective*. It uses the same MCJEPA construction as Experiment 1:

$$\mathcal{L}_{\mathrm{JEPA}} = \mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{state}}, \tag{53}$$

where $\mathcal{L}_{\mathrm{MC}}$ is defined in Eq. (5) and $\mathcal{L}_{\mathrm{state}}$ in Eq. (7). The online context encoder produces

$$q_\theta(Z_t \mid X_{\le t}),$$

while the EMA target encoder produces the local future target

$$q_{\bar{\theta}}(Z_{t+h} \mid X_{t+h}).$$

A single row-stochastic transition matrix generates all horizons through A^h. This regime therefore represents the latent-prediction viewpoint: no observation-sequence likelihood or filtering target influences representation learning. For evaluation of sequence NLL, a Gaussian observation model is fitted only after JEPA training and consequently does not influence the learned representation.

The second regime is *HMM sequence + filter distillation*. Here, the latent transition model

$$p_\phi(Z_{t+1} \mid Z_t)$$

and Gaussian emission model

$$p_\psi(X_t \mid Z_t)$$

are trained through exact observation-sequence negative log-likelihood,

$$\mathcal{L}_{\mathrm{HMM}} = -\mathbb{E}\left[\frac{1}{T}\log p_{\phi,\psi}(X_{1:T})\right], \tag{54}$$

computed by the HMM forward algorithm.

Sequence likelihood trains the generative transition–emission model, but it does not by itself require the amortized context encoder to represent the corresponding HMM filtering belief. We therefore additionally distill the exact filtering posterior into the context encoder through

$$\mathcal{L}_{\mathrm{filter}} = \mathbb{E}_t\left[\mathrm{KL}\left(\mathrm{sg}(\widetilde{q}_t) \,\|\, q_\theta(Z_t \mid X_{\le t})\right)\right]. \tag{55}$$

where

$$\widetilde{q}_t = p_{\phi,\psi}(Z_t \mid X_{\le t})$$

denotes the exact filtering posterior under the current HMM transition and emission models.[5]

[5]Both $\widetilde{q}_t$ and $q_\theta(Z_t \mid X_{\le t})$ represent a belief over the current latent state after observations through time t have been assimilated. The exact HMM filter first propagates the previous filtering belief through the transition kernel,

$$\pi_t^-(z_t) = \sum_{z_{t-1}} p_\phi(z_t \mid z_{t-1})\widetilde{q}_{t-1}(z_{t-1}),$$

and then incorporates the current observation through the emission likelihood,

$$\widetilde{q}_t(z_t) = \frac{p_\psi(X_t \mid z_t)\,\pi_t^-(z_t)}{\sum_{z_t'} p_\psi(X_t \mid z_t')\,\pi_t^-(z_t')}.$$

Thus, ϕ determines how probability mass is propagated between latent states, whereas ψ determines how the current observation updates that predictive prior. For a discrete HMM with transition matrix A,

$$\widetilde{q}_t(j) = \frac{p_\psi(X_t \mid Z_t = j)\sum_i \widetilde{q}_{t-1}(i)A_{ij}}{\sum_{j'} p_\psi(X_t \mid Z_t = j')\sum_i \widetilde{q}_{t-1}(i)A_{ij'}}.$$

Hence $\mathcal{L}_{\mathrm{filter}}$ compares like with like: it distills the exact current-state filtering belief into the amortized context encoder rather than comparing the context encoder with the pre-observation predictive prior.

The complete HMM-style regime uses

$$\mathcal{L}_{\mathrm{HMM+filter}} = \lambda_{\mathrm{seq}}\mathcal{L}_{\mathrm{HMM}} + \lambda_{\mathrm{filter}}\mathcal{L}_{\mathrm{filter}} + \mathcal{L}_{\mathrm{state}}. \tag{56}$$

Importantly, this regime contains *no JEPA latent-prediction loss* $\mathcal{L}_{\mathrm{MC}}$.

The third regime is the *hybrid HMM + latent* objective:

$$\begin{aligned}\mathcal{L}_{\mathrm{hybrid}} = \underbrace{\lambda_{\mathrm{seq}}\mathcal{L}_{\mathrm{HMM}}}_{\text{sequence evidence}} + \underbrace{\lambda_{\mathrm{latent}}\mathcal{L}_{\mathrm{MC}}}_{\text{JEPA latent prediction}} \\ + \underbrace{\lambda_{\mathrm{filter}}\mathcal{L}_{\mathrm{filter}}}_{\text{filtering-posterior alignment}} + \underbrace{\mathcal{L}_{\mathrm{state}}}_{\text{state-use regularization}}.\end{aligned} \tag{57}$$

Importantly, $\mathcal{L}_{\mathrm{MC}}$ is implemented in exactly the same way as in the JEPA-only regime: the future latent target is produced by the EMA target encoder,

$$q_{\bar{\theta}}(Z_{t+h} \mid X_{t+h}),$$

and the context prediction is generated by the shared transition,

$$q_{\theta}(Z_t \mid X_{\le t})A^h.$$

The exact HMM filtering posterior $\widetilde{q}_t$ is used only in $\mathcal{L}_{\mathrm{filter}}$ and does not replace the JEPA target in $\mathcal{L}_{\mathrm{MC}}$. Moreover, the transition matrix used by the HMM sequence model is the same transition matrix used by the latent-prediction objective, so both training signals act on the same latent dynamics.

The state-use regularizer and its coefficients are shared across all three encoder-training regimes. The remaining objective weights, initialization scheme, and optimization settings are given in Appendix J.

Motivation and comparison design. The three regimes form a controlled objective-level comparison. The *JEPA latent objective* uses latent predictive alignment but no observation-sequence likelihood or filtering target. The *HMM sequence + filter distillation* regime does the converse: it uses observation-sequence likelihood and filtering-posterior supervision but no JEPA latent-prediction loss. The *hybrid* regime adds both HMM-style signals to the same MCJEPA latent-prediction objective used by the JEPA-only baseline.

This comparison addresses three related questions. First, does adding HMM-style sequence and filtering supervision improve MCJEPA relative to latent-only training? Second, can the hybrid retain the genuine JEPA latent-prediction objective while approaching the probabilistic-model recovery achieved by HMM-style training? Third, is $\mathcal{L}_{\mathrm{MC}}$ necessary for training this latent-state architecture at all, or can the same architecture instead be trained through HMM-style sequence and filtering supervision?

Results. Table 3 reveals a clear objective-level distinction. The JEPA-only model successfully learns a meaningful predictive latent state, reaching

$$\mathrm{ARI} = 0.9802 \pm 0.0053,$$

but its observation-sequence NLL is

$$1.2753 \pm 0.0069,$$

and its transition-recovery error is

$$\mathcal{E}_A = 0.0191 \pm 0.0020.$$

This is consistent with what the objective directly supervises: $\mathcal{L}_{\mathrm{MC}}$ trains predictive agreement in latent space, but does not directly maximize observation-sequence likelihood or jointly train an emission model with the representation.

Adding HMM-style supervision produces a substantial improvement. The hybrid retains the same EMA-target MCJEPA loss but additionally optimizes sequence likelihood and filtering alignment. Its ARI rises to

$$0.9905 \pm 0.0008,$$

its sequence NLL decreases to

$$1.2660 \pm 0.0073,$$

and its transition error falls to

$$\mathcal{E}_A = 0.0103 \pm 0.0017.$$

Relative to JEPA-only training, this corresponds to an approximately 46% reduction in transition-matrix error. Moreover, the improvement is seed-consistent: for each of the five random seeds, the hybrid improves over JEPA-only on ARI, NMI, sequence NLL, transition recovery, filtering KL, and true-state prediction NLL at every evaluated horizon $h \in \{1, 2, 4, 8\}$. Thus, the gain from HMM-style supervision is not driven by a single favorable run.

The hybrid also nearly closes the observation-sequence likelihood gap to HMM-style training. The HMM+filter regime achieves sequence NLL

$$1.265885 \pm 0.007220,$$

whereas the hybrid obtains

$$1.266019 \pm 0.007271.$$

Their difference is only

$$1.34 \times 10^{-4}$$

NLL per time step, compared with a JEPA-to-HMM gap of approximately

$$9.41 \times 10^{-3}.$$

Equivalently, the hybrid closes approximately 98.6% of the JEPA-only sequence-NLL gap to HMM-style training while retaining the genuine JEPA latent-prediction objective.

Transition recovery shows the same qualitative result. The hybrid has the numerically smallest mean error,

$$0.0103 \pm 0.0017,$$

compared with

$$0.0105 \pm 0.0014$$

for HMM+filter and

$$0.0191 \pm 0.0020$$

for JEPA-only. The difference between hybrid and HMM-style training is small relative to the across-seed variability, so we interpret the two as achieving comparable transition recovery rather than claiming that the hybrid is superior to the correctly specified HMM objective. The important contrast is that both recover the transition substantially more faithfully than latent-only JEPA training.

The HMM+filter regime provides the complementary result. Despite containing no $\mathcal{L}_{\mathrm{MC}}$, it achieves the strongest state recovery,

$$\mathrm{ARI} = 0.9945 \pm 0.0008,$$

and

$$\mathrm{NMI} = 0.9884 \pm 0.0014,$$

together with the best sequence NLL and transition recovery comparable to the hybrid. Thus, the JEPA latent-prediction objective is not required to train this categorical latent-state architecture successfully: the same architecture can instead be trained using HMM-style sequence and filtering supervision, together with the common state-use regularizer.

The multi-horizon prediction results reinforce this conclusion. At every evaluated horizon $h \in \{1, 2, 4, 8\}$, both HMM-style and hybrid training achieve lower true-state prediction NLL than JEPA-only training. The differences are largest at shorter horizons and diminish at longer horizons as the transition dynamics mix. Complete multi-horizon results are provided in Appendix J.

Filtering as a role diagnostic. Filtering KL requires a different interpretation from sequence NLL and transition recovery. The HMM+filter regime reaches

$$\mathrm{KL}_{\mathrm{filter}} = 0.000060 \pm 0.000007,$$

which is expected because its amortized encoder is explicitly trained to reproduce the exact HMM filtering posterior. The hybrid obtains

$$0.004355 \pm 0.001015,$$

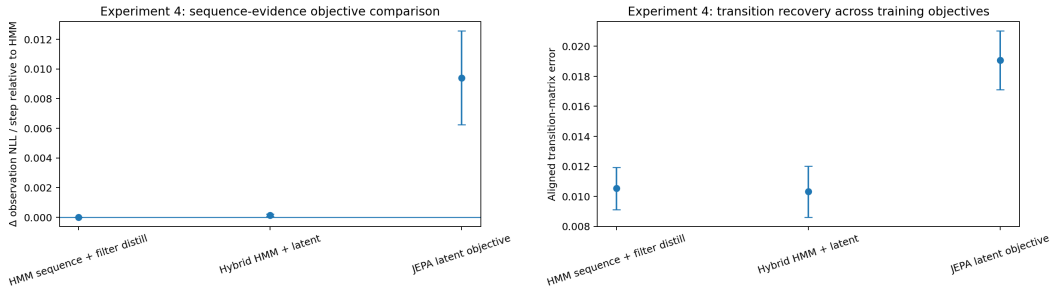


Figure 13: Objective-level comparison in Experiment 4. *Left:* observation-sequence NLL per time step relative to the HMM sequence + filter-distillation regime. Hybrid training nearly closes the entire JEPA-to-HMM sequence-likelihood gap while retaining the genuine MCJEPA latent-prediction objective. *Right:* permutation-aligned transition-matrix recovery error. HMM-style and hybrid training achieve closely matched transition recovery, and both substantially outperform latent-only JEPA training. Error bars denote mean $\pm$ one standard deviation over five seeds.

while JEPA-only gives

$$0.025460 \pm 0.006423.$$

For the HMM+filter and hybrid regimes this quantity is training aligned and should therefore be interpreted as a diagnostic that the intended filtering role has been learned, rather than as an independent generalization metric. For JEPA-only, by contrast, the filtering distribution is constructed only after fitting the post-hoc observation model, so its filtering KL measures how closely latent-only representation learning happens to agree with the filter induced by that fitted probabilistic model.

The ordering is nevertheless informative about the roles induced by the different objectives. HMM+filter explicitly learns an amortized filter; the hybrid remains substantially aligned with that filtering interpretation while simultaneously satisfying the EMA-target JEPA objective; and JEPA-only has no requirement that its history encoder coincide with a Bayesian filtering belief. The corresponding diagnostic is reported separately in Appendix J.

Figure 13 focuses on the two metrics that most directly expose the objective-level distinction. The left panel reports observation-sequence NLL relative to HMM-style training. The hybrid lies almost on the HMM reference, whereas JEPA-only retains a clear positive gap. The right panel reports permutation-aligned transition recovery error: HMM-style and hybrid training form a closely matched pair, while JEPA-only exhibits substantially larger error.

Taken together, these results support the distinction developed earlier between an HMM-compatible latent-state representation and full model-and-objective equivalence. The same HMM-compatible latent-state architecture can support JEPA-style latent prediction, HMM-style probabilistic sequence learning, or a combination of the two, but sharing the underlying probabilistic model class does not imply that different training objectives recover the same fitted model. Explicit sequence likelihood supplies transition–emission supervision that latent prediction alone does not provide, while filtering distillation connects the resulting HMM posterior back to the amortized PIB-VJEPA context encoder.

Experiment 4 supports two particularly important practical conclusions. First, incorporating HMM-style sequence and filtering supervision into MCJEPA improves recovery of the underlying probabilistic latent-state model while retaining the original JEPA latent-prediction objective. The hybrid improves over JEPA-only training on every reported metric for every seed, nearly matches HMM-style sequence likelihood, and recovers the transition dynamics at essentially the same level as HMM-style training. Second, the same latent-state architecture can be trained successfully without $\mathcal{L}_{\mathrm{MC}}$: the HMM sequence + filter-distillation regime attains the strongest state recovery and sequence likelihood despite omitting the JEPA latent-prediction objective altogether. Thus, the distinction between MCJEPA and an HMM is not determined by architecture alone; it also depends fundamentally on the objective used to train that architecture.

More generally, the three regimes expose a continuum of training objectives on the same latent-state family:

$$\underbrace{\mathcal{L}_{\mathrm{MC}}}_{\text{JEPA-style training}} \quad \longleftrightarrow \quad \underbrace{\mathcal{L}_{\mathrm{MC}} + \mathcal{L}_{\mathrm{HMM}} + \mathcal{L}_{\mathrm{filter}}}_{\text{hybrid training}} \quad \longleftrightarrow \quad \underbrace{\mathcal{L}_{\mathrm{HMM}} + \mathcal{L}_{\mathrm{filter}}}_{\text{HMM-style training}},$$

with $\mathcal{L}_{\mathrm{state}}$ shared as a state-use regularizer. The hybrid demonstrates that HMM-style probabilistic sequence learning can be incorporated without abandoning JEPA-style latent prediction, while the HMM-style end of the spectrum shows that the same architecture can also be trained without latent-prediction supervision altogether. This objective continuum makes precise the paper's broader claim: probabilistic temporal JEPA and HMMs can share an underlying latent Markov architecture while differing in how strongly HMM-equivalent probabilistic semantics are enforced by training.

Further optimization details, complete multi-horizon results, and the training-aligned filtering diagnostic are reported in Appendix J.

K.5 Summary of Experiments

The four experiments test complementary and progressively stronger aspects of the proposed HMM interpretation of probabilistic temporal JEPA. Experiment 1 establishes the finite-state Markov structure: MCJEPA learns an explicit shared transition matrix whose powers generate all prediction horizons, thereby guaranteeing exact direct-versus-composed consistency. The correctly specified Gaussian HMM remains strongest when emissions are ambiguous, while independently trained horizon-specific predictors can gain some predictive flexibility at the cost of a less faithful and non-compositional transition law. The collapse ablations further show that occupancy and entropy regularization play complementary roles in stable discrete-state learning.

Experiment 2 isolates the inference role of the context encoder. Using exact oracle posteriors, it shows that history-based filtering $p(S_t \mid X_{\leq t})$ becomes increasingly more informative than local evidence $p(S_t \mid X_t)$ as emissions overlap. This supports the interpretation of a history-dependent context encoder $q_\theta(Z_t \mid X_{\leq t})$ as an amortized filtering belief rather than an emission model.

Experiment 3 addresses how such a Markov state can be constructed. In a controlled second-order binary process, predictive compression removes the redundant history variable X_{t-2} and selects the known minimal sufficient state $Z_t^\star = (X_{t-1}, X_t)$ without loss of predictive information, whereas further compression becomes insufficient. Exhaustive enumeration of all deterministic history partitions provides a global reference for the prediction–compression trade-off, while the residual-history diagnostic supplies the complementary sufficiency test: adding the previous representation state Z_{t-1} provides essentially no predictive benefit once Z_t is sufficient, but yields a large held-out gain when $Z_t = X_t$ has discarded the transition-relevant variable X_{t-1}. Together, these results separate *minimality* from *sufficiency* and show how predictive compression can Markovize an observation process by constructing a compact predictive state.

Finally, Experiment 4 makes the objective-level distinction explicit through a controlled comparison on the same latent-state family. The architecture can be trained with the original JEPA-style latent-prediction objective, HMM-style sequence likelihood and filtering distillation, or a hybrid containing all three signals. Adding HMM-style supervision to the genuine MCJEPA objective improves state recovery, observation-sequence likelihood, transition recovery, filtering agreement, and multi-horizon prediction relative to latent-only training. The hybrid nearly closes the entire JEPA-to-HMM sequence-likelihood gap and recovers the transition dynamics at essentially the same level as HMM-style training, while retaining the original EMA-target JEPA latent-prediction objective. Conversely, the HMM-style regime achieves the strongest state recovery and sequence likelihood despite using no $\mathcal{L}_{\mathrm{MC}}$, showing that the same latent-state architecture can also be trained successfully without JEPA latent-prediction supervision.

Taken together, the experiments support a progressively stronger view of probabilistic temporal JEPA: it can instantiate a coherent latent Markov architecture; its context encoder can acquire the role of a filtering distribution; predictive compression can construct a compact sufficient Markov state; and the same architecture can be trained along a continuum from JEPA-style latent prediction, through hybrid JEPA–HMM learning, to HMM-style probabilistic sequence learning without $\mathcal{L}_{\mathrm{MC}}$. The degree of HMM equivalence therefore depends not only on architectural structure, but also on the probabilistic components and, critically, the objective used to train them.

L Extended Discussion

What is "secretly an HMM"? The central claim is structural and probabilistic, but not unconditional. Full, time-indexed PIB-VJEPA exposes the same three computational roles as an HMM: inference of a latent-state belief from observations, propagation of that state through a Markov transition, and a state-to-observation map. The correspondence becomes progressively stronger across the four levels developed in this paper: computational correspondence, emission-complete latent-state representation, sequence-level HMM equivalence, and model-and-objective equivalence. In particular, the stochastic encoder is *not* an emission model; it plays the recognition or filtering role. The emission direction may instead be supplied by a decoder, by the inverse of an invertible target encoder, or implicitly through the Bayes-consistent conditional induced by a local stochastic encoder. Reaching sequence-level HMM equivalence further requires Markov, marginal-consistency, and filtering-consistency conditions, as formalized in Theorem 1; reaching model-and-objective equivalence additionally requires HMM-compatible sequence-level probabilistic training.

Beyond MCJEPA: when is a general JEPA predictor Markov? The HMM correspondence is not specific to MCJEPA, nor does a neural-network predictor cease to be Markov merely because it is nonlinear or highly expressive. Markovianity is a conditional-independence property rather than a restriction on the functional form of the predictor. A general temporal JEPA may use an arbitrary neural transition

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t),$$

implemented, for example, by an MLP, Transformer, mixture model, or another conditional density estimator. It remains *first-order* Markov with respect to Z_t whenever

$$p_\phi(Z_{t+1} \mid Z_{\le t}, \xi_{\le t}) = p_\phi(Z_{t+1} \mid Z_t, \xi_t).$$

The transition can therefore be arbitrarily nonlinear; for example,

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \mathcal{N}(Z_{t+1}; \mu_\phi(Z_t, \xi_t), \Sigma_\phi(Z_t, \xi_t)),$$

while a deterministic temporal JEPA is recovered through the Dirac kernel

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t) = \delta(Z_{t+1} - P_\phi(Z_t, \xi_t)).$$

MCJEPA is therefore an explicit finite-state instantiation of a broader latent-Markov interpretation of temporal JEPA: it replaces the general transition kernel by

$$p_\phi(Z_{t+1} = j \mid Z_t = i) = A_{ij},$$

making the Markov property and multi-step Chapman–Kolmogorov composition especially transparent. Conceptually,

$$\text{MCJEPA} \subset \text{Markov neural-predictor JEPA} \subset \text{general temporal JEPA}.$$

Thus, MCJEPA demonstrates the correspondence in its simplest explicit form; it does not create or solely represent the correspondence.

If the predictor genuinely depends on information beyond Z_t, however, the latent process need not be first-order Markov in Z_t alone. For example, a recurrent predictor may use (Eq.25)

$$p_\phi(Z_{t+1} \mid Z_t, M_t, \xi_t),$$

where M_t summarizes additional history (i.e. memory), or a higher-order predictor may depend directly on $(Z_t, \dots, Z_{t-k+1})$. In such cases, exact HMM correspondence with Z_t as the hidden state does not follow. However, a first-order representation can often be recovered by *augmenting the state*. For recurrent dynamics, one may define

$$\widetilde{Z}_t = (Z_t, M_t),$$

while for a kth-order predictor one may use

$$\widetilde{Z}_t = (Z_t, Z_{t-1}, \dots, Z_{t-k+1}).$$

If the augmented state contains all transition-relevant information from the past, then

$$p(\widetilde{Z}_{t+1} \mid \widetilde{Z}_{\le t}, \xi_{\le t}) = p(\widetilde{Z}_{t+1} \mid \widetilde{Z}_t, \xi_t),$$

and the first-order latent-state interpretation is restored. The correspondence developed in this paper therefore applies beyond MCJEPA to general probabilistic temporal JEPA whenever the chosen latent state, possibly after augmentation, admits such a first-order Markov transition together with the additional emission and consistency conditions required for sequence-level equivalence.

Predictive representation learning as Markov-state construction. The HMM perspective changes the interpretation of the JEPA representation itself. Rather than viewing Z_t only as a feature vector useful for predicting another feature vector, we can ask whether it constitutes a *predictive state*: does it retain the information from the past that is needed for future prediction while discarding redundant history? This distinction also separates two notions that can otherwise be conflated. A predictor may be architecturally first-order,

$$p_\phi(Z_{t+1} \mid Z_t, \xi_t),$$

without Z_t being a sufficient Markov representation of the underlying process. In particular,

$$\text{first-order predictor} \not\Rightarrow \text{sufficient Markov state.}$$

If older history remains predictive after conditioning on Z_t, for example if

$$p(Z_{t+1} \mid Z_t, Z_{t-1}) \neq p(Z_{t+1} \mid Z_t),$$

then the architecture is imposing a first-order transition on a representation that has not fully Markovized the process.

This gives predictive information-bottleneck learning a state-space interpretation. Ideally, the representation should satisfy predictive sufficiency,

$$X_{>t} \perp X_{\leq t} \mid Z_t,$$

while retaining as little redundant information about the past as necessary. Compression therefore promotes minimality, whereas predictive sufficiency prevents over-compression. Together they can transform a non-Markov observation process into an approximately Markov latent process rather than merely forcing a Markov predictor onto an insufficient representation. Experiment 3 illustrates this distinction explicitly: compression removes redundant history to recover the known minimal sufficient state, while the residual-history diagnostic tests whether transition-relevant information remains outside the current representation. Thus, compression and residual predictability provide complementary tools for *learning and testing* a Markov representation rather than assuming one a priori.

Architecture and objective are separate design choices. The HMM correspondence also clarifies a distinction that is easy to obscure: sharing an encode–transition–emit architecture does not imply sharing a training objective. Standard JEPA training may optimize only latent predictive alignment and never maximize observation-sequence likelihood. Experiment 4 makes this distinction operational. Adding HMM-style sequence and filtering supervision to the genuine MCJEPA objective substantially improves recovery of the probabilistic latent-state model, with the hybrid approaching HMM-level sequence likelihood and transition recovery while retaining EMA-target latent prediction. Conversely, the same latent-state architecture can be trained successfully using HMM sequence likelihood and filtering distillation without $\mathcal{L}_{\mathrm{MC}}$ at all. The resulting continuum

$$\text{JEPA latent prediction} \longleftrightarrow \text{hybrid JEPA–HMM training} \longleftrightarrow \text{HMM-style sequence learning}$$

shows that the boundary between probabilistic temporal JEPA and classical state-space modeling is determined not only by model components, but also by which probabilistic semantics the objective enforces.

Why observation reconstruction can remain optional. An HMM explicitly models $p(X_t \mid S_t)$ because its likelihood is defined in observation space. JEPA may instead deliberately concentrate learning on the information required for future prediction, avoiding the cost of reconstructing high-entropy observation details that are irrelevant to the predictive task. A decoder can be introduced when observation forecasting or sequence likelihood is required, but it need not participate in the core representation-learning objective. An invertible target encoder provides another realization of the observation map, although invertibility limits the encoder's ability to discard nuisance information or reduce dimensionality and can therefore conflict with predictive compression. The implicit-emission construction establishes a probabilistic completion even without either explicit map, but that induced conditional need not be tractable enough for practical generation or likelihood evaluation.