The Forecast–Reconstruction Tradeoff Is an Artifact of Coupling a Single Latent to Both Objectives

Anonymous authors
Paper under double-blind review

Abstract

Latent dynamical models that share a single latent space between reconstruction and forecasting face a tradeoff: improving one objective degrades the other. We show this tradeoff is architectural, not inherent. For linear systems, we prove that reconstruction-optimal and prediction-optimal projections generically occupy different subspaces; empirically, we observe persistent gradient conflict on the shared encoder. We propose *carrier-residual decoupling*: a wide carrier handles reconstruction, a narrow residual code handles forecasting, and no parameters are shared between the two. On five dynamical systems (6–20D), a 64D spatiotemporal benchmark, three forecast heads (DMD, MLP, neural ODE), and four multi-task gradient methods, no param-matched coupled model Pareto-dominates the decoupled one.

Code: `https://github.com/Kirmaadaaa/ICLR27-ReconFcst-Tradeoff`

1 Introduction

Autoencoder-based latent dynamical models learn a low-dimensional representation of high-dimensional observations and fit a dynamics model in that latent space. The representation must simultaneously preserve enough variance for faithful reconstruction and exhibit temporal regularity that the dynamics model can exploit. These two requirements compete: the standard weighted loss $\mathcal{L} = (1-\varphi)\,\mathcal{L}_{\text{rec}} + \varphi\,\mathcal{L}_{\text{fcst}}$ traces a Pareto frontier as φ varies, and no single setting minimises both (Lusch et al., 2018; Otto & Rowley, 2019).

This tradeoff is widely accepted. We argue it reflects a limitation of the *coupled* architecture—one encoder, one latent, two objectives—rather than a fundamental property of the learning problem. Two arguments support this claim:

1. **Capacity bound.** The rank-k projection that minimises reconstruction error captures the top-k variance directions, while the projection that minimises prediction error captures the top-k *predictable* directions. For a linear system these are the eigenvectors of two different matrices (Σ_x vs. $F\Sigma_x F^\top$), which generically do not align (Proposition 1). For nonlinear systems the mismatch is strictly worse, because the relevant Jacobian varies across the state space (Corollary 1).
2. **Gradient conflict.** The gradients of reconstruction and forecasting losses on the shared encoder point in opposing directions 62–92% of training batches, so each gradient step compromises at least one objective (Section 3.2).

If the bottleneck is architectural, then allocating separate subspaces to each objective should remove it. We propose *carrier-residual decoupling*: a wide j-dim carrier optimised for reconstruction, and a narrow k-dim residual code optimised for dynamics, trained in three frozen-teacher phases with no shared gradient path.

We evaluate on five synthetic systems (6–20D) plus Kuramoto–Sivashinsky (64D), across three forecast heads (DMD, MLP, neural ODE), four multi-task gradient methods (PCGrad,

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.

CAGrad, Nash-MTL, FAMO), and three literature methods (mLaSDI, Koopa, AIKAE). All baselines are parameter-matched. The decoupled model achieves lower reconstruction and forecast error in the large majority of comparisons where a single model must serve both objectives.

2 Background

Koopman autoencoders and DMD. The Koopman operator (Koopman, 1931; Mezić, 2005) lifts nonlinear dynamics into a linear, infinite-dimensional representation. Deep autoencoders learn finite-dimensional approximations: the encoder maps observations to coordinates where dynamics are approximately linear, and the decoder maps back (Lusch et al., 2018; Takeishi et al., 2017; Otto & Rowley, 2019; Yeung et al., 2019; Champion et al., 2019; Gin et al., 2021). Physics-informed (Erichson et al., 2019; Pan & Duraisamy, 2020) and consistent (Azencot et al., 2020) variants add stability penalties; AIKAE (Li et al., 2025) adds auxiliary-task invariant subspaces; Koopa (Liu et al., 2024b) learns the operator end-to-end. All retain the coupled architecture. Dynamic Mode Decomposition (DMD) (Schmid, 2010) fits a best-fit linear operator $A = YX^{\dagger}$; its eigendecomposition $A = V\Lambda V^{-1}$ enables non-iterative multi-step prediction $z_{T+\tau} = V\Lambda^{\tau}V^{-1}z_T$ (Brunton et al., 2022). mLaSDI (Bonneville et al., 2024) stacks multiple latent-space DMD models for parametric problems.

Multi-task learning. Joint training of reconstruction and forecasting is multi-task learning (Sener & Koltun, 2018). When task gradients conflict, PCGrad (Yu et al., 2020) projects out the conflicting component, CAGrad (Liu et al., 2021) finds a common descent direction, Nash-MTL (Navon et al., 2022) frames the problem as a bargaining game, and FAMO (Liu et al., 2024a) adapts task weights online. All four methods modify the training procedure to mitigate gradient interference; none changes the underlying shared-encoder architecture.

Residual learning and neural ODEs. Residual connections (He et al., 2016) decompose a function into an identity plus a learned residual; our carrier-residual decomposition applies the same idea temporally. Neural ODEs (Chen et al., 2018; Rubanova et al., 2019) parameterise continuous-time dynamics $dz/dt = f_\theta(z)$; we use them as one of three pluggable forecast heads.

3 Theory

3.1 Capacity Bound

Consider a linear system $\mathbf{x}_{t+1} = F\,\mathbf{x}_t + \boldsymbol{\varepsilon}_t$, $\boldsymbol{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, Q)$, with stationary covariance $\Sigma_x = F\Sigma_x F^{\top} + Q$. We seek an orthonormal $P \in \mathbb{R}^{n\times k}$, $\Pi = PP^{\top}$. The reconstruction loss $\mathcal{L}_{\text{rec}}(P) = \text{tr}(\Sigma_x) - \text{tr}(P^{\top}\Sigma_x P)$ is minimised by the top-k eigenvectors of Σ_x. The prediction loss $\mathcal{L}_{\text{pred}}(P) = \text{tr}(M^{\top}M\Sigma_x) + \text{tr}(Q)$, $M = F - \Pi F\Pi$, captures the dynamics approximation error.

Proposition 1 (Capacity bound)**.** *Let Σ_x have distinct eigenvalues. If $F\Sigma_x \neq \Sigma_x F$, then the reconstruction-optimal projection P^*_{rec} is not a stationary point of the prediction loss, and no single rank-k projection simultaneously minimises both objectives.*

Proof. Consider n=2, k=1, $\Sigma_x = \text{diag}(\lambda_1, \lambda_2)$ with $\lambda_1 > \lambda_2 > 0$, so $P^*_{\text{rec}} = e_1$.

At $P = e_1$: $\mathcal{L}_{\text{pred}}(e_1) = \lambda_1 F_{21}^2 + \lambda_2 F_{12}^2 + \lambda_2 F_{22}^2 + \text{tr}(Q)$.

At $P = e_2$: $\mathcal{L}_{\text{pred}}(e_2) = \lambda_1 F_{11}^2 + \lambda_2 F_{12}^2 + \text{tr}(Q)$.

The reconstruction-optimal e_1 is prediction-optimal only when $\lambda_1 F_{21}^2 + \lambda_2 F_{22}^2 \leq \lambda_1 F_{11}^2$. This fails whenever the off-diagonal or lower-right entries of F are sufficiently large—the generic case when $F\Sigma_x \neq \Sigma_x F$.

For general n, k: replacing v_k with v_{k+1} in P increases $\mathcal{L}_{\text{rec}}$ by $\lambda_k - \lambda_{k+1} > 0$ but can decrease $\mathcal{L}_{\text{pred}}$ whenever F couples the retained and excluded eigenspaces. This coupling is absent only when F and Σ_x commute (measure zero). □

Corollary 1 (Nonlinear extension)**.** *Let $\mathbf{x}_{t+1} = f(\mathbf{x}_t) + \boldsymbol{\varepsilon}_t$ with differentiable f and Jacobian $J(\mathbf{x}) = \nabla f(\mathbf{x})$. If J is not constant on the support of the data distribution (i.e. the dynamics are genuinely nonlinear), then the capacity bound is strictly tighter than in the linear case: no single rank-k projection can simultaneously minimise reconstruction and prediction error.*

Proof. At each state $\mathbf{x}_0$, one-step prediction through a rank-k projection P incurs a local error governed by the Jacobian $J(\mathbf{x}_0)$, which plays the role of F in Proposition 1. The locally prediction-optimal projection therefore depends on $J(\mathbf{x}_0)$. When J varies across the attractor—as it does for any genuinely nonlinear system—the locally optimal P varies with it.

A global encoder must fix a single P. By Proposition 1, at any state where $J(\mathbf{x}_0)\Sigma_x \neq \Sigma_x J(\mathbf{x}_0)$, the reconstruction-optimal P is not locally prediction-optimal. Since J is non-constant, this condition holds on an open dense subset of the attractor. Thus the capacity constraint is at least as strong as in the constant-J (linear) case, and strictly stronger when J varies sufficiently. □

Implication. The capacity bound applies to all five benchmark systems: directly for the linear system (Linear 5D) and via Corollary 1 for the nonlinear ones (Coupled Harmonic, Brusselator, Duffing, Lorenz-96). Carrier-residual decoupling avoids this constraint: j dimensions serve reconstruction and k dimensions serve dynamics, so neither is forced into a compromise projection.

3.2 Gradient Conflict

Proposition 2 (Gradient conflict)**.** *In a coupled autoencoder with loss $(1-\varphi)\mathcal{L}_{\text{rec}} + \varphi\mathcal{L}_{\text{fcst}}$, let $\mathbf{g}_r$ and $\mathbf{g}_f$ be the gradients of the two losses w.r.t. the shared encoder. If $\langle \mathbf{g}_r, \mathbf{g}_f \rangle < 0$, then for any $\varphi \in (0, 1)$:* (a) *the combined gradient makes less progress on each objective than the individual gradient would;* (b) *gradient magnitude is reduced by inter-task cancellation;* (c) *extreme φ values actively increase the opposing objective. In the decoupled architecture, $\mathbf{g}_f = \mathbf{0}$ on the encoder and no conflict arises.*

Proof. Let $\mathbf{g}_\varphi = (1-\varphi)\mathbf{g}_r + \varphi\mathbf{g}_f$.

(a) $\langle \mathbf{g}_\varphi, \mathbf{g}_r \rangle = (1-\varphi)\|\mathbf{g}_r\|^2 + \varphi\langle \mathbf{g}_f, \mathbf{g}_r \rangle$. Since $\varphi > 0$ and $\langle \mathbf{g}_f, \mathbf{g}_r \rangle < 0$, this is strictly less than $\|\mathbf{g}_r\|^2$. Symmetric for $\mathbf{g}_f$.

(b) $\|\mathbf{g}_\varphi\|^2 = (1-\varphi)^2\|\mathbf{g}_r\|^2 + 2\varphi(1-\varphi)\langle \mathbf{g}_r, \mathbf{g}_f \rangle + \varphi^2\|\mathbf{g}_f\|^2$. The cross-term is negative.

(c) $\langle \mathbf{g}_\varphi, \mathbf{g}_r \rangle < 0$ iff $\varphi > \|\mathbf{g}_r\|^2 / (\|\mathbf{g}_r\|^2 + |\langle \mathbf{g}_f, \mathbf{g}_r \rangle|) =: \varphi_{\text{hi}}$. By Cauchy–Schwarz, $\varphi_{\text{hi}} \in (0, 1)$. The threshold φ_{lo} for $\langle \mathbf{g}_\varphi, \mathbf{g}_f \rangle < 0$ is analogous, and $\varphi_{\text{lo}} < \varphi_{\text{hi}}$. □

Empirical verification. We train coupled autoencoders at six φ values and log $\cos(\mathbf{g}_r, \mathbf{g}_f)$ on the encoder every 10 batches. Figure 1 shows the cosine is negative 62–92% of the time across all systems, with mean cosine between -0.23 and -0.03. The conflict persists throughout training.

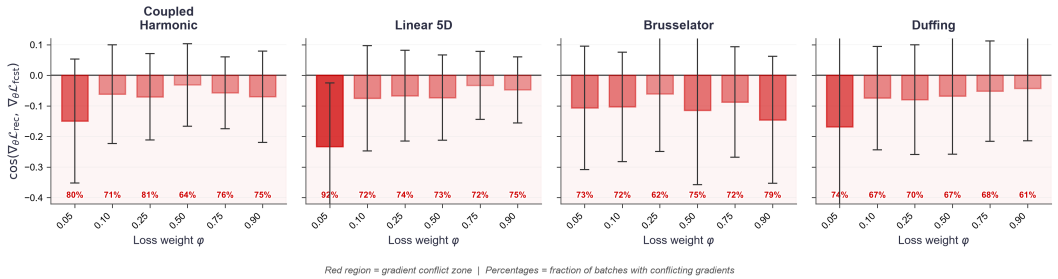


Figure 1: Cosine similarity between reconstruction and forecasting gradients on the shared encoder. Red bars: negative mean cosine. Percentages: fraction of batches with conflict.

4 Method

The core idea is to decouple a latent dynamical model into two non-overlapping subspaces: a *reconstruction subspace* of dimension j, optimised solely for observation fidelity, and a *dynamics subspace* of dimension $k \ll j$, optimised solely for temporal prediction. No parameters are shared between the two after the reconstruction module is trained, so neither objective's gradient can interfere with the other. Any encoder–decoder pair, any dynamics model, and any accumulation rule can fill these roles; the principle defines a family of *decoupled latent dynamical models*. Below we describe one concrete instantiation: a three-phase frozen-teacher pipeline.

Phase 1: Teacher autoencoder. An encoder $E : \mathbb{R}^n \to \mathbb{R}^j$ and decoder $D : \mathbb{R}^j \to \mathbb{R}^n$ are trained for reconstruction. The carrier $C_t = E(x_t)$ is wide enough ($j > k$) for near-perfect reconstruction.

Phase 2: Residual compressor. With E, D frozen, carrier residuals $\Delta C_t = C_{t+1} - C_t$ are compressed: $\psi : \mathbb{R}^j \to \mathbb{R}^k$, $\mu : \mathbb{R}^k \to \mathbb{R}^j$, trained to minimise $\|\Delta C_t - \mu(\psi(\Delta C_t))\|^2$. The residual code $b_t = \psi(\Delta C_t) \in \mathbb{R}^k$ captures temporal dynamics in a form amenable to prediction.

Phase 3: GRU accumulation. During forecasting, the carrier evolves via gated accumulation: $C_{t+1} = \mathrm{GRU}(\mu(b_t), C_t)$, trained with all prior modules frozen. This replaces naive addition $C_{t+1} = C_t + \mu(b_t)$, which compounds errors over long rollouts.

Pluggable forecast heads. Any model predicting b_{t+1} from b_t serves as the forecast head. We test three: **DMD** (post-hoc linear, $b_{t+1} = Ab_t$), **MLP** (learned nonlinear), and **neural ODE** ($db/dt = f_\theta(b)$, RK4 integration). All share the same frozen teacher, so reconstruction is identical across heads.

5 Experiments

5.1 Setup

Systems. Five synthetic dynamical systems: Coupled Harmonic Oscillator (6D), Linear 5D, Brusselator (10D), forced Duffing (10D), and Lorenz-96 (20D). We additionally test Kuramoto–Sivashinsky (KS, 64D) for scalability.

Protocol. Carrier j=8 (systems $\leq$10D), j=12 (L-96), j=32 (KS-64D); residual k matches the coupled AE latent dim. All models: 400 epochs/phase, batch 512, Adam (10^{-3}). The decoupled model uses hidden width h=64; each coupled baseline's h is widened ($h \approx 73$–95) until its total parameter count meets or exceeds the decoupled model's (1.00–1.01×; Appendix B). Results: mean over 5 seeds.

Metrics. Reconstruction RMSE (R) and forecast RMSE at horizon τ (Fτ). For coupled DMD baselines and all decoupled variants, forecast uses spectral DMD ($z_{T+\tau} = V\Lambda^{\tau}V^{-1}z_T$). For coupled and decoupled MLP and ODE heads, forecast uses each head's own learned predictor.

Baselines. *Coupled*: AE trained with $(1-\varphi)\mathcal{L}_{\text{rec}} + \varphi\mathcal{L}_{\text{fcst}}$ at φ=0.5. *Decoupled*: carrier-residual architecture above.

5.2 Head Comparison

Does decoupling help regardless of how the dynamics are modelled? Table 1 pairs each coupled head (DMD, MLP, neural ODE) against the decoupled model. Each coupled model is param-matched and trained jointly at φ=0.5; each uses its own head for prediction.

The Decoupled columns are identical across rows because the reconstruction comes entirely from the frozen Phase 1 teacher and the forecast is evaluated via spectral DMD on the same frozen b trajectory, irrespective of which head was trained in Phase 2.5. On four of five systems, Decoupled achieves lower R *and* lower F1 than every coupled head. On Brusselator, the coupled ODE achieves F1= .005 (vs. Decoupled .013) at the cost of R= .005 (vs. Decoupled .002): exactly the tradeoff the capacity bound predicts.

Table 1: Head comparison (mean±std, 5 seeds). Coupled models are param-matched (h=73–76). Decoupled values are identical across heads: reconstruction depends only on the frozen teacher; forecast uses spectral DMD on the same frozen b trajectory (§4). **Bold**: best per row.

		Coupled		Decoupled	
System	Head	R	F1	R	F1
C. Harm.	DMD	.702±.092	.815±.056	**.006**±.000	**.012**±.000
	MLP	.221±.070	.218±.068	**.006**±.000	**.012**±.000
	ODE	.340±.042	.338±.041	**.006**±.000	**.012**±.000
Lin. 5D	DMD	.563±.110	.679±.115	**.005**±.001	**.005**±.001
	MLP	.013±.001	.014±.001	**.005**±.001	**.005**±.001
	ODE	.009±.001	.009±.001	**.005**±.001	**.005**±.001
Bruss.	DMD	.479±.267	.665±.263	**.002**±.000	**.013**±.001
	MLP	.027±.012	.026±.013	**.002**±.000	**.013**±.001
	ODE	.005±.002	**.005**±.002	**.002**±.000	.013±.001
Duffing	DMD	.577±.251	.589±.153	**.005**±.001	**.006**±.001
	MLP	.017±.015	.015±.016	**.005**±.001	**.006**±.001
	ODE	.009±.004	.008±.004	**.005**±.001	**.006**±.001
L-96	DMD	.740±.035	.810±.053	**.387**±.007	**.406**±.007
	MLP	.600±.009	.577±.006	**.387**±.007	**.406**±.007
	ODE	.598±.015	.574±.009	**.387**±.007	**.406**±.007

5.3 Scalability: KS-64D

To test whether the advantage holds in higher dimensions, we evaluate on Kuramoto–Sivashinsky with n=64 spatial modes (Table 2). The decoupled model (j=32, k=16, h=64, 33 488 params) is compared against a param-matched coupled baseline (k=16, h=95, 33 966 params). Decoupled achieves 6× lower reconstruction error and lower forecast error at every horizon.

Table 2: KS-64D (mean±std, 5 seeds). Coupled is param-matched (h=95, 33 966 params).

Method	R	F1	F10	F50	F100
Coupled	.378±.017	.363±.064	.810±.073	1.05±.05	1.10±.07
Decoupled	**.062**±.003	**.050**±.002	**.166**±.010	**.646**±.076	**.926**±.08

5.4 Ablation

What drives the improvement? Table 3 isolates each component. *Coupled_k*: param-matched k-dim AE + spectral DMD. *Coupled_j*: j-dim AE (wider latent, more capacity). *Frozen_j*: j-dim AE trained recon-only, spectral DMD applied post-hoc (no dynamics gradient on the encoder). *Frozen_k*: same but k-dim. *Decoupled*: full carrier-residual + spectral DMD.

Frozen_j achieves the lowest reconstruction error—it has the widest latent and no dynamics gradient to distort it—but it provides no dynamics pipeline for multi-step rollout. Frozen_k has the dynamics bottleneck but poor reconstruction. Decoupled matches Frozen_j on reconstruction while providing the residual code pathway for forecasting. No single baseline achieves both; Decoupled does.

Table 3: Ablation (mean, 5 seeds). Coupled_k is param-matched (h=73–76). **Bold**: best in each row.

	Coupled_k		Coupled_j		Frozen_j		Frozen_k		Decoupled	
System	R	F1	R	F1	R	F1	R	F1	R	F1
C. Harm.	.702	.815	.032	.032	**.006**	**.008**	.167	.169	**.006**	.012
Lin. 5D	.563	.679	**.005**	.007	**.005**	.006	.011	.035	**.005**	**.005**
Bruss.	.479	.665	**.002**	**.007**	**.002**	**.007**	**.002**	.048	**.002**	.013
Duffing	.577	.589	**.004**	**.005**	**.004**	**.005**	.009	.031	.005	.006
L-96	.740	.810	.401	.422	.401	.422	.523	.541	**.387**	**.406**

5.5 Multi-Task Gradient Methods

If the problem is solely gradient conflict, perhaps better optimisation can fix the coupled model. We test four multi-task methods:PCGrad, CAGrad, Nash-MTL, FAMO, against the naive weighted sum (Table 4). All coupled models are param-matched. None substantially improves over the naive baseline; all remain far from the decoupled result. The bottleneck is architectural, not optimisation-level.

Table 4: Multi-task methods vs. Decoupled: F1 RMSE (mean±std, 5 seeds). All coupled models param-matched (h=73–76).

System	Naive	PCGrad (Yu et al., 2020)	CAGrad (Liu et al., 2021)	Nash-MTL (Navon et al., 2022)	FAMO (Liu et al., 2024a)	Decoupled
C. Harm.	.815±.056	.754±.084	.893±.146	.774±.051	.723±.070	**.012**±.000
Lin. 5D	.679±.115	.562±.093	.654±.167	.581±.101	.607±.156	**.005**±.001
Bruss.	.665±.263	.417±.115	.560±.217	.577±.181	.878±.817	**.013**±.001
Duffing	.589±.153	.660±.180	.712±.451	.815±.529	.383±.188	**.006**±.001
L-96	.810±.053	.789±.044	.585±.031	.776±.026	.799±.022	**.406**±.007

5.6 Pareto Frontier

Figure 2 shows the full picture. Sweeping $\varphi \in \{0, 0.05, \ldots, 1\}$ traces a Pareto frontier for the coupled model (grey). The decoupled model (⋆) falls inside this frontier on four systems and performs better than the best recon setting and the best forecast setting on Coupled Harmonics. It achieves near-φ=0 reconstruction with comparable or better forecast quality on most systems we test.

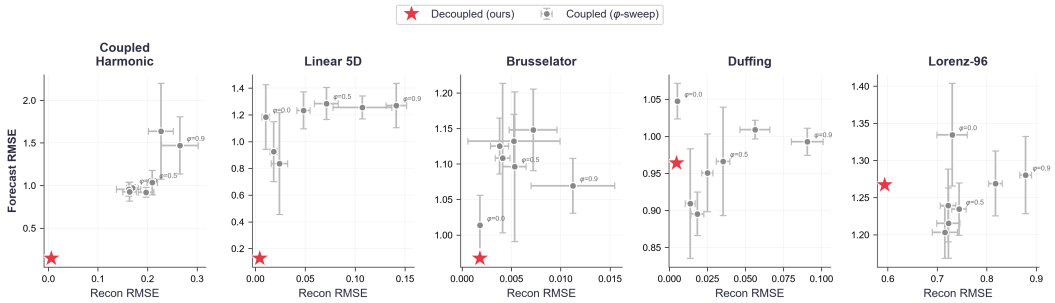


Figure 2: φ-sweep Pareto frontier (grey; mean±std, 5 seeds, spectral DMD). Decoupled ($\star$) falls inside the frontier on all systems.

5.7 Multi-Horizon Forecast

Table 10 (Appendix A) evaluates forecast RMSE at horizons $\tau \in \{1, 5, 10, 25, 50, 100\}$. Decoupled tracks the recon-only Frozen_j baseline closely at all horizons. The param-matched coupled model degrades rapidly because prediction errors in its shared latent space compound at each autoregressive step; by contrast, the decoupled architecture confines dynamics errors to the narrow b-space, leaving the wide carrier undistorted.

5.8 Literature Comparisons

We re-implement three literature methods, each run both standalone (param-matched) and integrated into our carrier-residual pipeline.

mLaSDI (Bonneville et al., 2024). Two sequential autoencoders on observation-space residuals. Table 5: Decoupled_mLaSDI improves F1 on the four structured systems; on L-96 standalone mLaSDI achieves lower forecast error.

Table 5: mLaSDI (Bonneville et al., 2024) (param-matched) vs. Decoupled_mLaSDI (mean, 5 seeds). **Bold**: best per metric.

	mLaSDI		Decoupled_mLaSDI	
System	F1	F100	F1	F100
C. Harm.	.052	.961	**.006**	**.101**
Lin. 5D	.025	.591	**.005**	**.100**
Bruss.	.034	1.18	**.004**	**.728**
Duffing	.011	**1.04**	**.006**	1.14
L-96	**.286**	**1.25**	.398	1.27

Koopa (Liu et al., 2024b). Koopa learns a linear Koopman operator K (no bias) end-to-end, jointly optimising the encoder, operator, and decoder in a single training pass. When integrated into our decoupled pipeline, the operator is trained on the frozen b trajectory instead. Table 6: Decoupled_Koopa dominates on R and F1 across the four structured systems; on L-96 standalone Koopa retains a long-horizon advantage.

Table 6: Koopa (Liu et al., 2024b) (param-matched) vs. Decoupled_Koopa (mean, 5 seeds). **Bold**: best per metric.

	Koopa			Decoupled_Koopa		
System	R	F1	F10	R	F1	F10
C. Harm.	.175	.173	.486	**.006**	**.013**	**.136**
Lin. 5D	.010	.139	.434	**.005**	**.006**	**.066**
Bruss.	.002	.060	.425	**.002**	**.013**	**.348**
Duffing	.005	.024	.217	**.005**	**.006**	**.074**
L-96	.509	.476	**.897**	**.399**	**.417**	1.38

AIKAE (Li et al., 2025). Koopman operator with forward/backward consistency losses. Table 7: Decoupled_AIKAE dominates on R and F1 across the four structured systems; on L-96 AIKAE retains a long-horizon advantage.

Table 7: AIKAE (Li et al., 2025) (param-matched) vs. Decoupled_AIKAE (mean, 5 seeds). **Bold**: best per metric.

	AIKAE			Decoupled_AIKAE		
System	R	F1	F10	R	F1	F10
C. Harm.	.585	.931	.835	**.006**	**.013**	**.158**
Lin. 5D	.530	1.27	.850	**.005**	**.006**	**.093**
Bruss.	.454	.554	1.08	**.002**	**.014**	**.462**
Duffing	.483	.604	.662	**.005**	**.006**	**.096**
L-96	.744	.705	**.887**	**.399**	**.420**	1.62

5.9 Noise Robustness

We add observation noise $x_{\text{noisy}} = x + \sigma \cdot \text{std}(x) \cdot \varepsilon$, $\varepsilon \sim \mathcal{N}(0, I)$, and compare Frozen_j, Frozen_k, and Decoupled (Table 8). Decoupled matches Frozen_j at both noise levels; Frozen_k is consistently worst. The decoupling overhead introduces no additional noise sensitivity.

Table 8: Noise robustness: F1 RMSE (mean, 3 seeds). **Bold**: best per cell.

	$\sigma = 0.0$			$\sigma = 0.1$		
System	Frz_j	Frz_k	Decpld	Frz_j	Frz_k	Decpld
C. Harm.	.008	.169	**.007**	.135	.187	**.135**
Lin. 5D	**.006**	.035	.007	**.131**	.138	**.131**
Bruss.	.007	.048	**.007**	.113	.125	**.112**
Duffing	**.005**	.031	.005	.116	.118	**.116**
L-96	**.422**	.541	.423	**.433**	.547	.435

DMD under noise. When spectral DMD is used for long-horizon evaluation (500-step rollout), observation noise biases the estimated eigenvalues and degrades the forecast systematically. Figure 3 shows the full degradation curves: on Coupled Harmonic and Linear 5D, the Decoupled advantage persists across all noise levels, while on Brusselator and Duffing the two methods converge as noise dominates. Table 9 gives the exact numbers. At σ=0, Decoupled's spectral forecast is 5–9× lower than Coupled on the two linear-like systems; at σ=0.2 both methods degrade but Decoupled remains superior on three of four systems in both reconstruction and forecast.

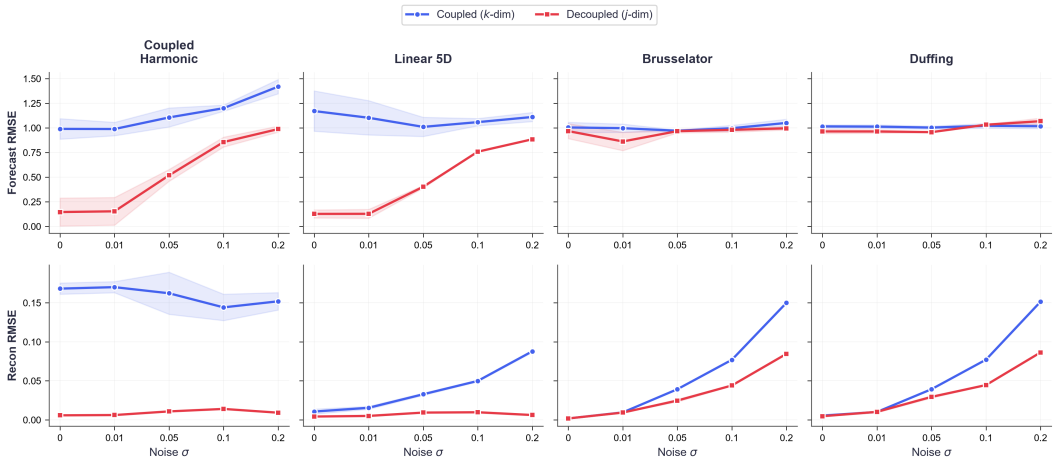


Figure 3: Noise degradation curves (500-step spectral DMD rollout). Top: forecast RMSE; bottom: reconstruction RMSE. Shaded bands: ±1 std over 3 seeds. Decoupled (red) maintains lower error than Coupled (blue) across all noise levels on Coupled Harmonic and Linear 5D; the two converge on the chaotic systems at high noise.

Table 9: DMD noise sensitivity: 500-step spectral rollout RMSE (mean, 3 seeds). **Bold**: best per cell.

		σ=0	σ=0.01	σ=0.05	σ=0.1	σ=0.2
C. Harm.	Coupled	.990	.988	1.11	1.20	1.42
	Decoupled	**.146**	**.153**	**.518**	**.856**	**.988**
Lin. 5D	Coupled	1.17	1.10	1.01	1.06	1.11
	Decoupled	**.127**	**.128**	**.402**	**.758**	**.884**
Bruss.	Coupled	1.00	**.997**	**.971**	**.996**	1.05
	Decoupled	**.967**	**.862**	.967	.980	**.995**
Duffing	Coupled	1.02	1.01	1.00	**1.02**	**1.02**
	Decoupled	**.964**	**.963**	**.957**	1.03	1.07

Forecast RMSE only; reconstruction RMSE visible in Figure 3.

6 Discussion

The reconstruction–forecasting tradeoff arises when a single latent must serve both objectives. The capacity bound (Proposition 1) shows that the optimal projections for the two tasks generically differ; gradient conflict (Proposition 2) shows that joint training wastes optimiser effort through inter-task cancellation.

Carrier-residual decoupling addresses both: a wide carrier handles reconstruction, a narrow residual code handles dynamics, and no parameters are shared after Phase 1. The ablation (§5.4) shows that no single frozen baseline achieves what Decoupled provides. The head comparison (§5.2) shows that no coupled variant—even with a nonlinear ODE head—Pareto-dominates the decoupled model. Multi-task gradient methods do not close the gap (§5.5). Three literature methods improve when run on the decoupled pipeline (§5.8).

Limitations. The decoupled model adds parameters (ψ, μ, GRU) beyond the baseline AE; all coupled baselines are widened to match (1.00–1.01×; Appendix B). The three-phase training is more involved than a single end-to-end pass. Proposition 1 is proved for linear systems and extended to nonlinear ones via Corollary 1, but the nonlinear bound relies on the Jacobian argument rather than a direct global analysis. All systems are synthetic. On Lorenz-96 (chaotic), the advantage is modest.

Future work. Extending decoupling to irregular time series, integrating it within transformer-based temporal models, and developing rules for choosing j and k are natural next steps.

References

Omri Azencot, N. Benjamin Erichson, Vanessa Lin, and Michael W. Mahoney. Forecasting sequential data using consistent Koopman autoencoders. In *ICML*, pp. 475–485, 2020.

Christophe Bonneville, Youngsoo Choi, Debojyoti Ghosh, and Jonathan L. Belof. gLaSDI: Parametric physics-informed greedy latent space dynamics identification. *J. Comput. Phys.*, 506:112903, 2024.

Steven L. Brunton, Marko Budišić, Eurika Kaiser, and J. Nathan Kutz. Modern Koopman theory for dynamical systems. *SIAM Review*, 64(2):229–340, 2022.

Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Data-driven discovery of coordinates and governing equations. *Proc. Natl. Acad. Sci.*, 116(45):22445–22451, 2019.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *NeurIPS*, 2018.

N. Benjamin Erichson, Michael Muehlebach, and Michael W. Mahoney. Physics-informed autoencoders for Lyapunov-stable fluid flow prediction. *arXiv preprint arXiv:1905.10866*, 2019.

Craig Gin, Bethany Lusch, Steven L. Brunton, and J. Nathan Kutz. Deep learning models for global coordinate transformations that linearize PDEs. *European J. Appl. Math.*, 32 (3):515–539, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Bernard O. Koopman. Hamiltonian systems and transformations in Hilbert space. *Proc. Natl. Acad. Sci.*, 17(5):315–318, 1931.

Yingzhuo Li, Xinjie Lin, and Omri Azencot. Deep Koopman autoencoder for learning invariant subspaces via auxiliary tasks. *Trans. Mach. Learn. Res. (TMLR)*, 2025.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *NeurIPS*, 2021.

Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. FAMO: Fast adaptive multitask optimization. In *NeurIPS*, 2024a.

Yong Liu, Chenyu Li, Jianmin Wang, and Mingsheng Long. Koopa: Learning non-stationary time series with Koopman predictors. In *NeurIPS*, 2024b.

Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, 2018.

Igor Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41:309–325, 2005.

Aviv Navon, Idan Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *ICML*, 2022.

Samuel E. Otto and Clarence W. Rowley. Linearly recurrent autoencoder networks for learning dynamics. *SIAM J. Appl. Dyn. Syst.*, 18(1):558–593, 2019.

Shaowu Pan and Karthik Duraisamy. Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability. *SIAM J. Appl. Dyn. Syst.*, 19(1):480–509, 2020.

Yulia Rubanova, Ricky T. Q. Chen, and David Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *NeurIPS*, 2019.

Peter J. Schmid. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.*, 656:5–28, 2010.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *NeurIPS*, 2018.

Naoya Takeishi, Yoshinobu Kawahara, and Takehisa Yairi. Learning Koopman invariant subspaces for dynamic mode decomposition. In *NeurIPS*, pp. 1130–1140, 2017.

Enoch Yeung, Soumya Kundu, and Nathan Hodas. Learning deep neural network representations for Koopman operators of nonlinear dynamical systems. In *American Control Conf. (ACC)*, pp. 4832–4839, 2019.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *NeurIPS*, 2020.

A Multi-Horizon Forecast

Table 10 evaluates forecast RMSE at horizons $\tau \in \{1, 5, 10, 25, 50, 100\}$. Coupled_k is param-matched. Decoupled tracks Frozen_j closely at all horizons while the coupled model degrades rapidly.

Table 10: Multi-horizon forecast RMSE (mean, 5 seeds, spectral DMD). Coupled_k is param-matched. **Bold**: best per row.

System	Method	F1	F5	F10	F25	F50	F100
C. Harm.	Coupled_k	.715	.735	.786	.960	1.20	1.40
	Frozen_j	.007	.022	.040	.069	.085	.088
	Decoupled	**.007**	**.022**	**.039**	**.068**	**.083**	**.086**
Lin. 5D	Coupled_k	.574	.590	.667	.900	1.26	1.73
	Frozen_j	**.006**	**.018**	**.033**	**.064**	**.091**	**.095**
	Decoupled	.006	.018	.033	.065	.092	.098
Bruss.	Coupled_k	.467	.609	.899	1.33	1.67	1.54
	Frozen_j	.009	.036	.159	.501	**.801**	**.726**
	Decoupled	**.008**	**.035**	**.157**	**.497**	.803	.736
Duffing	Coupled_k	.506	.532	.623	.890	1.53	2.63
	Frozen_j	.006	**.010**	**.050**	.340	.744	1.10
	Decoupled	**.005**	.011	.052	**.338**	**.735**	**1.09**
L-96	Coupled_k	.701	.854	.931	1.00	1.00	1.05
	Frozen_j	**.420**	.815	1.03	1.18	**1.29**	1.25
	Decoupled	.420	**.812**	**1.03**	**1.18**	1.30	**1.25**

B Training Details and Reproducibility

All models use 2-hidden-layer MLPs with ELU activations and hidden width h=64. Training: Adam, lr=10^{-3}, batch 512, 400 epochs/phase. Seeds: $\{0, 1, 2, 42, 123\}$ (noise: $\{0, 1, 42\}$).

Phase 1 trains E+D. Phase 2 freezes E+D, trains ψ+μ. Phase 3 freezes all but the GRU. DMD is fitted post-hoc.

Coupled MLP and ODE heads are trained jointly at φ=0.5 with gradient clipping (max_norm=1). Decoupled heads are trained on the frozen b trajectory (Phase 2.5).

Parameter matching. Each coupled baseline uses a wider hidden layer h so its param count $\geq$ the decoupled model's (Table 11).

Table 11: Parameter counts.

System	n	Coupled h	Coupled	Decoupled	Ratio
C. Harmonic	6	73	12 436	12 362	1.01×
Linear 5D	5	73	12 289	12 233	1.00×
Brusselator	10	73	12 870	12 749	1.01×
Duffing	10	73	12 870	12 749	1.01×
Lorenz-96	20	76	16 546	16 478	1.00×
KS-64D	64	95	33 966	33 488	1.01×

Latent dimensions: C. Harmonic (k=4, j=8), Linear 5D (k=4, j=8), Brusselator (k=3, j=8), Duffing (k=3, j=8), L-96 (k=10, j=12), KS-64D (k=16, j=32).

Data generation. All systems: `solve_ivp` (RK45, rtol=atol=10^{-10}, Δt=0.05, $[0, 500]$ time units). First 200 samples discarded. Train: 4000 samples; test: 500.

C AI Assistance Disclosure

This work used Claude (Anthropic) as a writing aid. All experimental code, theory, architecture, and scientific claims are the authors' sole work. Claude was used to improve grammar, clarity and was used to act as an adversarial reviewer to identify gaps in the exposition. The authors carefully reviewed and edited all content produced by the tool, and take full responsibility for the content of this paper.

Claude was also used to cross check and verify the math and theory of this paper. To ensure quality and validity of the work both propositions and the nonlinear corollary have been formally verified in Lean 4 using the Mathlib library. The proof assistant's kernel machine-checks every deduction against its type theory, providing a certificate of logical validity independent of human review. The formalisation source is included in the supplementary material.

All code required to reproduce the experiments is included in the github repository mentioned under the abstract. Said repository also contains the Lean 4 formalisation of the propositions and corollary for reviewers to verify.