NOETHER WORLD MODELS: AN EXACTLY CONSERVED FAMILY OF LATENT INVARIANTS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models predict by rolling a learned latent transition forward over many steps. Current methods pursue conservation through a soft penalty, a symplectic integrator or an implicit projection, so each exact guarantee covers a small set of scalars fixed before training and paid for with an implicit update. We propose the Noether World Model (NWM), which assembles the generator as a graph Laplacian over object subspaces, $\mathbf{H} = \mathbf{L} \otimes I_m$, and advances the latent with a Cayley transform. The resulting unitary operator conserves a $2m^2$-dimensional family of quadratic invariants exactly at every value of its weights, and that family is exactly the set of quadratic forms with that property. Experiments on three physics datasets and three planning environments show that NWM holds the entire family at double precision whereas no baseline holds more than the norm, with $32\times$ fewer parameters than the strongest physics-structured baseline. Decoded quality in distribution separates the nine designs by only 1.3 dB, whereas under a zero-shot change of force law NWM leads the unconstrained transitions by 1.9 dB, so the family pays where the data leaves the training distribution. Exactness has a price, since the operator fixes every member of the family and most goals in a latent it did not shape lie off the reachable set. The same family can nonetheless enter the training objective of a world model whose predictor we do not replace, and it improves planning only where that predictor already drifts, a condition measurable before retraining.

1 INTRODUCTION

World models have emerged as a general paradigm for prediction and physical reasoning, learning action-conditioned dynamics in a compact latent space and predicting future states from past observations and actions, which enables planning through imagined rollouts in model-based control (Hafner et al., 2025; Hansen et al., 2024), video prediction (Micheli et al., 2023), and object-centric physical reasoning (Daniel et al., 2026). While the learned dynamics can fit any transition the data present, they remain an unconstrained neural map, so that no quantity of the latent is held fixed along a rollout and the error compounds with horizon. Current remedies for this drift fall into two paths, one through a soft constraint that adds a conservation term to the training objective (Raissi et al., 2019; Krishnapriyan et al., 2021), and another through a hard constraint that builds the conservation law into the architecture, for instance by parametrizing a Hamiltonian with a neural network (Greydanus et al., 2019; Chen et al., 2020) or by projecting onto the level sets of learned invariants (Matsubara & Yaguchi, 2023). Both paths constrain a conserved quantity to some extent, and several reach high precision on it. The gap is not precision on a named quantity but the size of the set a rollout can hold fixed at once: no existing design holds an entire family of quantities exactly, in a single explicit step that depends on the state and the action.

We now illustrate why exact conservation of an entire family, and a way to use it where an exact operator cannot be, matter in a world model. For chaotic dynamics the conserved quantities are often the only features that remain predictable far into the future (Lu et al., 2023). When they are held only approximately, the per-step violation is fed back into the next prediction and grows with the horizon (Janner et al., 2019), until minor errors compound and render the model useless for planning (Talvitie, 2017). Fixing a single quantity in advance is not sufficient either. A Hamiltonian design assumes the data are given in canonical coordinates, which are generally unknown (Chen et al., 2021), and once the coordinates are learned rather than given, a model can conserve the named quantity exactly while the invariant the encoder actually represents drifts, so symplectic architectures lose their advantage

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.

over an unconstrained model (Wang, 2026). Nor can an exact operator be imposed everywhere, since a norm-preserving map cannot represent contracting or expanding dynamics (Vorontsov et al., 2017; Kerg et al., 2019), and on a latent whose norm varies along the true trajectory it degrades accuracy. A world model therefore needs a transition that fixes not one named quantity but an entire family of them, exactly and at every value of its weights, and a way to carry those invariants into a latent where such an operator cannot be used.

Inspired by Noether's theorem, which pairs every conserved quantity with a continuous symmetry of the dynamics (Noether, 1918; Goldstein et al., 2002), we propose the Noether World Model (NWM), which enforces the symmetries through the algebraic form of the transition operator rather than its training objective. NWM assembles the generator as a graph Laplacian over N object subspaces of width m, $\mathbf{H} = \mathbf{L} \otimes I_m$, and advances the latent by a Cayley transform. For every value of the learned parameters, the generator is Hermitian, its row sums vanish, and each of its off-diagonal blocks is a scalar multiple of the identity. These three properties make a single explicit step preserve not one quantity but a $2m^2$-dimensional family of quadratic forms, and that family is exactly the set of quadratic forms preserved at every value of the couplings (Theorem 1, Corollary 2, Figure 1). Because the generator is assembled from the latent and the action at every step, the operator varies with the input while the family it preserves does not, and the only learned component is the coupling that fills $\mathbf{L}$. Exactness is not free on every latent, and we prove why: the one-step error of any norm-preserving map, ours included, is bounded below by the variation of the encoded norm along the trajectory (Proposition 3, the normal component in Figure 1). The bound therefore decides where the invariants belong, since NWM replaces the transition where the encoded norm barely moves, whereas where it moves or a host architecture must be kept intact, the invariants enter the training objective as a soft constraint instead.

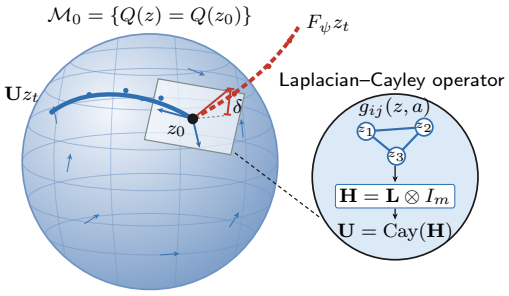

Figure 1: A rollout on the invariant set $\mathcal{M}_0$. The Laplacian–Cayley operator $\mathbf{U}$ (blue) advances z_t along the set at every step and for every value of the weights, holding all $2m^2$ forms of the family at once. A transition F_ψ under a penalty or an integrator (red) leaves the set, and the normal component δ of its step is the error no unitary operator can remove (Proposition 3). Inset: $\mathbf{H} = \mathbf{L} \otimes I_m$ is assembled from pairwise couplings g_{ij} and advanced by a Cayley step.

We evaluate NWM on three physics datasets, our own visual N-body simulator, the HNN 3-body release (Greydanus et al., 2019), and the Balls collision dataset (Lin et al., 2020), against eight alternative transitions under a shared encoder and two same-parameter controls. NWM holds every member of the invariant family at machine precision at $32\times$ fewer parameters, while decoded quality separates the nine designs by less than 1.3 dB, so what distinguishes them is what the rollout holds fixed rather than how well it predicts. The separation comes from the Laplacian structure alone: a control with the same coupling network and the same Cayley step, differing only in the Laplacian, returns none. Exactness has a price, since the operator fixes every member of the family, so most goals in a latent it did not shape lie off the reachable set, and planning there collapses to 3.8% whether the operator is given the host's capacity or a matched small one. The invariants do not depend on the operator, and the same family enters the objective of LeWM (Maes et al., 2026) on Push-T (Chi et al., 2023), Two Rooms (Sobal et al., 2025) and Reacher (Tassa et al., 2018) without changing it.

Our contributions are as follows. (i) We propose the Noether World Model, a latent world model whose transition operator is assembled as a graph Laplacian over object subspaces and advanced by a Cayley transform, and we prove that it conserves a $2m^2$-dimensional family of quadratic forms exactly at every value of its parameters, and that the family is exactly the set of forms with that property (§3.2, §3.3). **(ii)** We prove a lower bound on the one-step error of any norm-preserving map in terms of the variation of the encoded norm, and design two ways to use the invariants around it, the exact operator where that variation is small and a soft constraint on a fixed architecture where it is not (§3.4, §3.5). **(iii)** We validate NWM against eight alternative transitions and two same-parameter controls, and show that removing the Laplacian and nothing else eliminates the separation (§4). **(iv)** We carry the same invariants into LeWM (Maes et al., 2026) by adding one term to its objective, showing that they do not depend on the operator that fixes them (§4.4).

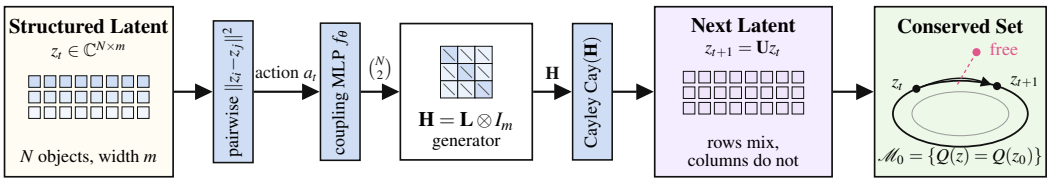

Figure 2: **Overall architecture of the Noether world model.** The latent holds N objects of width m, drawn at $N = 3$ with m shortened. The coupling MLP f_θ reads the pairwise distances and the action and emits one scalar per object pair, which is the only learned part of the operator. In $\mathbf{H} = \mathbf{L} \otimes I_m$ every cell is an $m \times m$ block proportional to I_m, marked by a diagonal stroke, with the diagonal blocks set by their own row. Row shading tracks object identity, so the mixed shades of the next latent show an update across objects and not channels, and the rose point lies outside the conserved set.

2 Related Work

Latent world models. Latent world models learn a dynamics model in a learned representation and plan inside it (Ha & Schmidhuber, 2018; Hafner et al., 2020; Hansen et al., 2024), and the joint-embedding line (LeCun, 2022) predicts directly in latent space without reconstruction, as in PLDM (Sobal et al., 2025), DINO-WM (Zhou et al., 2025) and LeWM (Maes et al., 2026), while LPWM (Daniel et al., 2026) factors the latent into object-centric particles, which is the structure our operator acts on. Across these designs the objective constrains what the encoder represents while the transition remains an unconstrained neural map, so no quantity of the latent is held fixed along a rollout, and we keep the encoder and the planner of these models and replace only the transition.

Physics-structured predictors. Learned dynamics obtain conservation by parameterising a scalar Hamiltonian and integrating its symplectic gradient (Greydanus et al., 2019; Zhong et al., 2020; Toth et al., 2020), so the guarantee lives in the continuous-time field and the learned step inherits it only up to a discretisation error (Offen & Ober-Blöbaum, 2022), whereas discrete-gradient integrators (Matsubara et al., 2020) and projection onto learned first integrals (Matsubara & Yaguchi, 2023) conserve exactly in discrete time at the price of an implicit step. A softer route penalises the residual of a known equation (Raissi et al., 2019) and conserves only on average (Krishnapriyan et al., 2021), and Noether's Razor (van der Ouderaa et al., 2024) learns the symmetry itself through a Bayesian objective, so that its invariant holds at a maximiser of the marginal likelihood rather than at every value of the weights. In every case the conserved quantities are a finite set fixed before training and the operator that conserves them is approximate or implicit, whereas ours is explicit, reads the state and the action, and conserves a family fixed by its structure.

Unitary and equivariant networks. Two architectural lines fix a property of the map by construction, since unitary recurrent networks (Arjovsky et al., 2016) with Cayley or exponential parameterisations (Helfrich et al., 2018; Lezcano-Casado & Martínez-Rubio, 2019) preserve the norm for every input, but the matrix is fixed, so it adapts to no state or action and any invariant beyond the norm depends on the matrix learned rather than on the design, while Interaction Networks (Battaglia et al., 2016) and EGNN (Satorras et al., 2021) learn input-dependent pairwise messages whose equivariance fixes how the output transforms under a group rather than which quantity stays constant. The graph line therefore offers input dependence without conservation and the unitary line the reverse, and our operator assembles input-dependent pairwise couplings into a generator whose unitary step conserves an entire family of quadratic forms (§3.3), which Appendix A places against each design in turn.

3 Method

We formalise the problem in §3.1 and construct a transition operator whose conservation comes from its algebraic form rather than from training. The construction has four components: the Laplacian–Cayley assembly (§3.2), the family of quadratic forms it conserves (§3.3), a lower bound on what exactness costs (§3.4), and two relaxations for the latents on which that bound is high (§3.5). The guarantee costs 3,963 parameters and one $N \times N$ solve per step, and Figure 2 shows the assembled operator from the structured latent through to the set it conserves.

3.1 Problem formulation

A latent world model rolls a state forward under a sequence of actions, and the transition operator carries that rollout from one step to the next (Hafner et al., 2020; Maes et al., 2026). The operator is left unconstrained, so nothing in it keeps any quantity of the state fixed as the rollout proceeds. We therefore constrain the algebraic form of the operator instead of the objective or the integrator that surrounds it (Raissi et al., 2019; Greydanus et al., 2019). Formally, an encoder Φ maps an observation x_t to a latent $z_t \in \mathbb{C}^d$ partitioned into N object subspaces $z = (z_1, \dots, z_N)$ with $z_i \in \mathbb{C}^m$ and $d = Nm$, with $N = 3$ throughout and varied in §4.2. The objective is to learn an operator $\mathbf{U}$ that advances the latent and leaves a set $\mathcal{Q}$ of functions of z unchanged,

$$z_{t+1} = \mathbf{U}(z_t, a_t)\, z_t, \qquad Q\big(\mathbf{U}(z, a)\, z\big) = Q(z) \quad \text{for all } Q \in \mathcal{Q},\ z,\ a, \text{ and all } (\theta, \phi), \tag{1}$$

where (θ, ϕ) are the parameters of the operator. Two properties separate Equation 1 from existing designs, in that the identity holds at every parameter value, whereas a penalty reaches it only at a minimiser and an integrator only as the step size vanishes, and in that $\mathbf{U}$ reads both z_t and a_t, whereas a fixed unitary matrix reads neither and an action-conditioned one reads only the action. The set $\mathcal{Q}$ is determined by the operator rather than fixed in advance: the assembly of §3.2 conserves every quadratic form that commutes with its generator, a $2m^2$-dimensional family identified in §3.3. For a concise exposition we follow three members throughout the paper,

$$\|z\|^2, \qquad \textstyle\sum_{i=1}^{N} z_i, \qquad \mathrm{Im}\big(z^\dagger \mathbf{J} z\big), \tag{2}$$

which arise from the unitary group $\mathcal{U}(d)$, the translation $z_i \mapsto z_i + \epsilon$, and the rotation generated by $\mathbf{J} = \mathrm{diag}(\mathbf{J}_2, \dots, \mathbf{J}_2)$, and which give each baseline a quantity it was designed to protect.

3.2 The Laplacian–Cayley assembly

To obtain both exactness and input dependence, we build $\mathbf{U}$ from a learned coupling per object pair, a Laplacian assembly whose pattern does not depend on the values of those couplings, and a Cayley transform that turns the assembly into a unitary step.

Learned coupling. Input dependence requires the operator to depend on the state and the action, so we let both enter through one real scalar per pair of objects,

$$\underbrace{g_{ij} = f_\theta\big(\|z_i - z_j\|^2,\ \rho_\phi(a_t)\big)}_{\binom{N}{2} \text{ scalars, the only learned part}} \qquad \underbrace{g_{ij} = g_{ji}}_{\text{one value shared per pair}} \tag{3}$$

where f_θ and ρ_ϕ are small networks.

Laplacian assembly. A free Hermitian generator built from these scalars would already give a unitary step, but its conserved set would move with θ and ϕ. To make the guarantee independent of them, we assemble the couplings into a graph Laplacian over the N object subspaces, so that the block coupling objects i and j is

$$\mathbf{H}_{ij} = \begin{cases} \underbrace{g_{ij}\, I_m}_{\text{learned}}, & i \neq j, \\ \underbrace{-\textstyle\sum_{k \neq i} g_{ik}\, I_m}_{\text{set by the row}}, & i = j, \end{cases} \qquad \text{giving} \qquad \underbrace{\mathbf{H} = \mathbf{L} \otimes I_m}_{\text{block-scalar}} \tag{4}$$

for a symmetric Laplacian $\mathbf{L} \in \mathbb{R}^{N \times N}$. Three properties follow from the pattern of Equation 4 rather than from the values placed in it, in that real entries together with a symmetric g make $\mathbf{H}$ Hermitian, while the diagonal construction makes each row of blocks sum to zero, and the factor I_m carried by every entry makes each block a multiple of the identity. Because the three hold at every (θ, ϕ), whatever they imply is conserved for all parameter values, as Equation 1 requires.

Cayley step. We turn the generator into a transition with the Cayley transform, which maps a Hermitian matrix to a unitary one (Helfrich et al., 2018; Lezcano-Casado & Martínez-Rubio, 2019),

$$\underbrace{\mathbf{U} = \mathrm{Cay}(\mathbf{H}) = \big(I + \tfrac{\mathrm{i}}{2}\mathbf{H}\big)^{-1}\big(I - \tfrac{\mathrm{i}}{2}\mathbf{H}\big)}_{\text{unitary for every } \mathbf{H} = \mathbf{H}^\dagger} \qquad \underbrace{\mathbf{U} = \mathbf{V} \otimes I_m}_{\mathbf{V} = \mathrm{Cay}(\mathbf{L}) \in \mathcal{U}(N)} \tag{5}$$

with $\mathrm{i} = \sqrt{-1}$. Since $\mathbf{H} = \mathbf{L} \otimes I_m$, the Kronecker structure passes to the transition, so that the transform acts on the $N \times N$ matrix $\mathbf{L}$ instead of the $d \times d$ matrix $\mathbf{H}$, which reduces the cost of the solve at each step from $O(d^3)$ to $O(N^3)$ without altering the result. Equation 5 therefore rotates the N objects into one another while leaving every coordinate channel untouched, and because the assembly fixes the three structural properties of $\mathbf{H}$ at every parameter value, §3.3 derives what they conserve from those properties alone. Algorithm 1 gives the rollout, in which line 1 encodes and partitions the observation while line 3 evaluates the couplings of Equation 3 and is the only line inside the loop that reads a parameter. The three lines that follow assemble those couplings into $\mathbf{L}$ by Equation 4, apply the Cayley transform by Equation 5, and advance the latent, which Theorem 1 shows leaves every member of $\mathcal{Q}$ unchanged. The training objective and the batched form are given in Appendix C.4.

Algorithm 1: Laplacian–Cayley rollout

Require: observation x_0, actions $a_{0:T-1}$, encoder Φ, networks f_θ, ρ_ϕ
1: $z_0 \leftarrow \Phi(x_0)$, partitioned as $(z_{0,1}, \dots, z_{0,N})$
2: **for** $t = 0, \dots, T-1$ **do**
3: $\quad g_{ij} \leftarrow f_\theta(\|z_{t,i} - z_{t,j}\|^2, \rho_\phi(a_t))$ ▷ Eq. 3
4: $\quad \mathbf{L}_{ij} \leftarrow g_{ij},\ \mathbf{L}_{ii} \leftarrow -\sum_{k \neq i} g_{ik}$ ▷ Eq. 4
5: $\quad \mathbf{V} \leftarrow \mathrm{Cay}(\mathbf{L})$ ▷ Eq. 5
6: $\quad z_{t+1} \leftarrow (\mathbf{V} \otimes I_m)\, z_t$ ▷ Thm. 1
7: **end for**
8: **return** $z_{1:T}$

3.3 The invariant family

To characterise the conserved set, we define the commutant of $\mathbf{H}$ as the set of matrices G with $[\mathbf{H}, G] = 0$, which $\mathrm{Cay}(\mathbf{H})$ shares because it is a rational function of $\mathbf{H}$. Each of the three structural properties then contributes one invariant, since Hermiticity makes $\mathrm{Cay}(\mathbf{H})$ unitary and so preserves $\|z\|^2$, while zero row sums place the translation direction $\mathbf{1}_N \otimes e$ in the kernel of $\mathbf{H}$ and so preserve $\sum_i z_i$, and block-scalar form places $I_N \otimes C$ in the commutant for every C, the rotation generator $\mathbf{J}$ among them. The three members of Equation 2 are therefore invariant under one step.

Theorem 1 (Exact conservation by construction)**.** *Let* $\mathbf{H}$ *satisfy Equation 4 and* $\mathbf{U} = \mathrm{Cay}(\mathbf{H})$*. Then for every* $z \in \mathbb{C}^d$ *and every action* a*,*

$$\|\mathbf{U}z\|^2 = \|z\|^2, \qquad \textstyle\sum_i (\mathbf{U}z)_i = \sum_i z_i, \qquad \mathrm{Im}\big((\mathbf{U}z)^\dagger \mathbf{J} \mathbf{U} z\big) = \mathrm{Im}(z^\dagger \mathbf{J} z). \tag{6}$$

Proof sketch. Set $\mathbf{A} = \frac{\mathrm{i}}{2}\mathbf{H}$, so $\mathbf{A}^\dagger = -\mathbf{A}$ and $\mathbf{U} = (I + \mathbf{A})^{-1}(I - \mathbf{A})$. Hermiticity yields $\mathbf{U}^\dagger \mathbf{U} = I$, since the four factors commute. Zero row sums yield $\mathbf{U}(\mathbf{1}_N \otimes e) = \mathbf{1}_N \otimes e$ for every $e \in \mathbb{C}^m$, and block-scalar form yields $[\mathbf{U}, \mathbf{J}] = 0$. Appendix C.2 gives the full proof. □

Equation 6 follows from the commutant alone, without a loss term, a projection or an integrator, and the same reasoning extends to any quadratic form built from that commutant, since $A \otimes C$ lies in the commutant of $\mathbf{H}$ for every C whenever A lies in the commutant of $\mathbf{L}$.

Corollary 2 (Invariant family)**.** *Let* $\mathbf{U}$ *be as in Theorem 1. For every* $A \in \mathrm{span}\{I_N, \mathbf{1}_N \mathbf{1}_N^\top\}$ *and every* $C \in \mathbb{C}^{m \times m}$*, the quadratic form* $Q_{A,C}(z) = z^\dagger (A \otimes C) z$ *satisfies* $Q_{A,C}(\mathbf{U}z) = Q_{A,C}(z)$*, and a quadratic form preserved by* $\mathrm{Cay}(\mathbf{L}) \otimes I_m$ *for every admissible* $\mathbf{L}$ *lies in this span.*

The family is a real vector space of dimension $2m^2$ and coincides with the set $\mathcal{Q}$ of Equation 1, because the quantifier over (θ, ϕ) in that equation is the one the converse of Corollary 2 matches. A single operator preserves more than the family, since it commutes with the spectral projectors of its own $\mathbf{L}$, whereas the family is what remains invariant at every value of the couplings. The family contains $\|z\|^2$ as $(A, C) = (I_N, I_m)$ and $\mathrm{Im}(z^\dagger \mathbf{J} z)$ as $(I_N, \mathbf{J}_m)$, while the block sum $\sum_i z_i$ is linear rather than quadratic and is conserved by the zero row sums instead (Appendix C.5). The boundary of the family is verified in §4.2 by sampling forms from the span and from its orthogonal complement, and by permuting coordinates within objects (Proposition 4). Since the family cannot cover data that leaves the level sets it fixes, and since the norm is the loosest of those level sets, §3.4 bounds what variation of the norm costs while §4.4 measures the remaining ones.

3.4 WHAT EXACTNESS COSTS

A unitary step preserves $\|z\|$, so wherever the encoded norm changes between consecutive frames the difference is an error no such step can remove, and its size bounds the one-step error from below.

Proposition 3 (Lower bound for any exact operator)**.** *For every unitary* $\mathbf{U}$ *and every pair* (z_t, z_{t+1})*,*

$$\|\mathbf{U}z_t - z_{t+1}\| \;\geq\; \big|\,\|z_{t+1}\| - \|z_t\|\,\big|. \tag{7}$$

Equation 7 is the reverse triangle inequality applied with $\|\mathbf{U}z_t\| = \|z_t\|$, and it is tight when z_{t+1} is a positive multiple of $\mathbf{U}z_t$. The bound belongs to the encoder and the data, and no choice of $\mathbf{L}$, θ or ϕ lowers it. Written on the encoder output, the per-step floor is

$$\delta = \text{median}_t \;\frac{\big|\,\|\Phi(x_{t+1})\| - \|\Phi(x_t)\|\,\big|}{\|\Phi(x_t)\|}, \tag{8}$$

computable from Φ and the data. Two designs follow from this floor, in that the Laplacian–Cayley assembly is used directly wherever the transition is free to be replaced and the encoder is trained alongside it, so that the encoder adapts to the constraint, whereas the encoder is fixed and the floor becomes a property of the host wherever the predictor must be kept intact, so that the invariants enter as the penalty of §3.5 instead. Both designs require a latent partitioned into N subspaces of equal width m, with m even in the first of them, while N itself is unconstrained, and §4.4 measures what the exact operator costs on a host latent.

3.5 RELAXATIONS

Two settings fall outside the exact operator, since a pretrained encoder need not keep the encoded norm fixed and the architecture hosting the transition is often not free to be replaced, as with the Transformer inside a JEPA world model (LeCun, 2022; Maes et al., 2026). For these we design two schemes that differ in where the constraint is placed, in that an interpolated constraint remains in the algebraic form of the operator and weakens it by a fixed amount, whereas a soft constraint leaves the operator unchanged and places the invariants in the training objective instead.

Interpolated constraint. Proposition 3 bounds the one-step error from below without relating accuracy to exactness, so we introduce one scalar that mixes the assembly with a free predictor and makes that relation measurable,

$$z_{t+1} = (1-\alpha)\,\underbrace{\mathbf{U}(z_t, a_t)\, z_t}_{\text{exact}} + \alpha\,\underbrace{F_\psi(z_t, a_t)}_{\text{free}}, \qquad \alpha \in [0,1], \tag{9}$$

so that $\alpha = 0$ recovers Theorem 1 and $\alpha = 1$ removes every guarantee. At intermediate values the violation of any member of $\mathcal{Q}$ is bounded by $\alpha\|F_\psi(z_t, a_t) - \mathbf{U}z_t\|$ (Proposition 5, Appendix C.2). The scalar is fixed before training and receives no gradient, so each value defines a separate transition and a separate model, and the resulting sequence traces the cost of exactness on a pixel latent (§4.4).

Soft constraint. Since Equation 9 still replaces the transition, it is unavailable once the predictor is fixed by the host, and the three members of Equation 2 are then imposed on the prediction $\hat{z}_{t+1} = F_\psi(z_t, a_t)$ through the objective instead of the operator,

$$\mathcal{L} = \underbrace{\mathcal{L}_{\text{base}}}_{\text{host, unchanged}} + \underbrace{\lambda_H \sum_{Q \in \mathcal{Q}_3} \left(\frac{Q(\hat{z}_{t+1}) - Q(z_t)}{|Q(z_t)| + \epsilon}\right)^2}_{\text{Noether penalty}}, \tag{10}$$

$$\mathcal{L}^{\text{LeWM}}_{\text{base}} = \mathcal{L}_{\text{pred}} + \lambda_S\, \mathcal{L}_{\text{SIGReg}}, \tag{11}$$

where $\mathcal{Q}_3$ denotes the three members and the vector-valued one enters through the squared norm of its relative difference. Since the penalty requires only z_t and $\hat{z}_{t+1}$, it transfers to any latent world model that predicts a next latent, whatever the predictor is made of, and we instantiate it on LeWM (§4.4). Conservation is then carried by the objective and holds at a minimiser of $\mathcal{L}$ rather than at every (θ, ϕ) (Krishnapriyan et al., 2021), so the assembly and the penalty cover the two ends of Equation 1. On a real latent $\text{Im}(z^\dagger \mathbf{J} z)$ vanishes identically, so the penalty acts on the other two members (Remark 6).

4 EXPERIMENTS

We conduct experiments on three physics datasets and three planning environments to answer four questions. **Q1.** During a latent rollout, what does a conservation-aware transition hold fixed, to what precision, over how many quantities, and at what cost in predictive accuracy? **Q2.** Which part of the assembly is responsible, and does the guarantee remain exact under a wider coupling network, a relabelled latent, or a different object partition? **Q3.** What effect does exact conservation have on long-horizon rollouts and on evaluation under a change of force law? **Q4.** Under what conditions is the exact operator applicable, and where those conditions fail, can the same invariants be imposed on a host world model through its objective alone?

Setup. Visual N-body is rendered from a velocity-Verlet integrator, so the three invariants have analytic values, while HNN 3-body and Balls are public and follow different force laws. Nine predictors cover three unconstrained transitions (MLP, GRU, GNN), two physics priors (SymODEN, HNN), an action-conditioned unitary step (UWM), two same-parameter controls (Ctrl-free, Ctrl-cayley), and our operator (NWM), each trained under three seeds. Each latent is rolled forward 29 steps in double precision, and we read the drift $|Q(z_{29}) - Q(z_0)|/|Q(z_0)|$ for the three invariants of Equation 2 and for 100 quadratic forms, 50 drawn from the span of Corollary 2 and 50 from its orthogonal complement. Generalisation is probed in two ways, by extending the rollout to 100 steps and by evaluating zero-shot under a $1/r^2$ force law and at half the energy scale. On the planning environments Push-T, Tworoom and Reacher the host is LeWM and planning uses CEM over 200 episodes per evaluation seed (Appendix D.1).

4.1 CONSERVATION PRECISION AND THE SET OF CONSERVED QUANTITIES

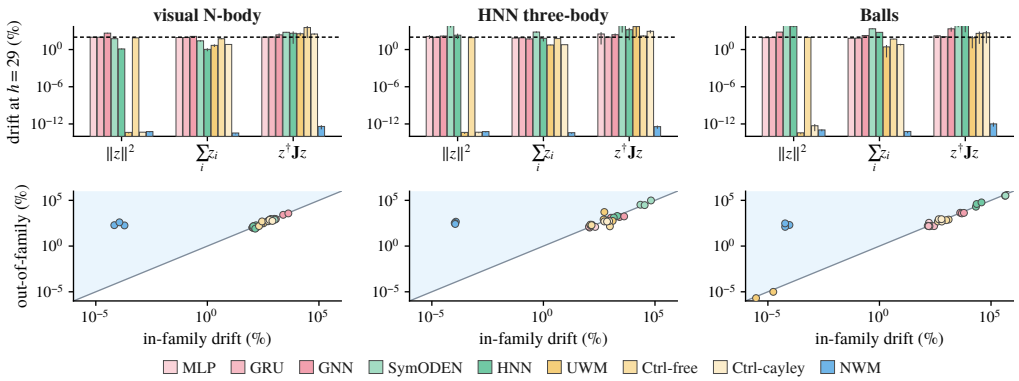

Figure 3: **Conservation across three datasets.** Top: drift of the three invariants of Equation 2 at $h = 29$, mean and standard deviation over three seeds, with the dashed line at 100%. Bottom: in-family against out-of-family drift, one point per method and seed, with the diagonal marking no separation. Open markers on Balls are frozen rollouts, not conservation (Appendix D.2).

Results. As shown in Figure 3 and Table 1, NWM is the only design that holds all three invariants at double precision, between 2.5×10^{-14} and 1.0×10^{-12} on every dataset, whereas UWM and Ctrl-cayley hold the norm alone, indicating that unitarity fixes one quantity rather than a family. The family ratio makes the same point for the whole $3{,}200$-dimensional family of Corollary 2, reaching 4.3×10^6 to 6.0×10^6 for NWM against at most 41.0 for any competitor, at an accuracy cost that leaves NWM second to fourth of nine in latent error and 0.10 dB behind the best PSNR on visual N-body (Appendix D.3). Holding a named quantity is therefore within reach of any unitary step, while holding the family requires the Laplacian assembly, and §4.2 shows that the separation is lost as soon as that assembly is removed from a generator that retains everything else.

4.2 WHERE THE FAMILY COMES FROM

Results. The guarantee follows from the assembly rather than from unitarity, since Ctrl-cayley shares f_θ, ρ_ϕ and the Cayley step with NWM and differs only in placing the $\binom{N}{2}$ scalars on a random Hermitian basis instead of a Laplacian, and yet its family ratio is 0.97 where the Laplacian reaches

Table 1: **Main results.** Family ratio is out-of-family over in-family drift at $h = 29$, the out-of-family forms orthogonalised against the span of Corollary 2. $\log|\det J|$ is zero for a volume-preserving step and includes $\partial \mathbf{U}/\partial z$. Best in **bold**, second underlined, † where the standard deviation exceeds the mean. Per-seed values in Appendix D.2, accuracy in Appendix D.3.

		Family ratio (↑)			
Method	**# Par.**	N-body	3-body	Balls	$\log\lvert\det J\rvert$
MLP	190K	0.90	1.27	2.31	-1039
GRU	234K	1.02	0.94	0.89	-424
GNN	69K	1.10	1.12†	0.78	-5.39
SymODEN	159K	1.59	1.02	1.20	$+4.5\times10^{-11}$
HNN	128K	2.16	1.42	1.18	-1.9×10^{-9}
UWM	950K	0.87	1.30	1.50†	$+3.1\times10^{-15}$
Ctrl-free	3,945	0.86	1.10	0.57	-9366
Ctrl-cayley	3,963	0.97	1.00	41.0†	-4.2×10^{-8}
NWM	3,963	$\mathbf{4.3\times10^{6}}$	$\mathbf{3.6\times10^{6}}$	$\mathbf{6.0\times10^{6}}$	$+1.0\times10^{-15}$

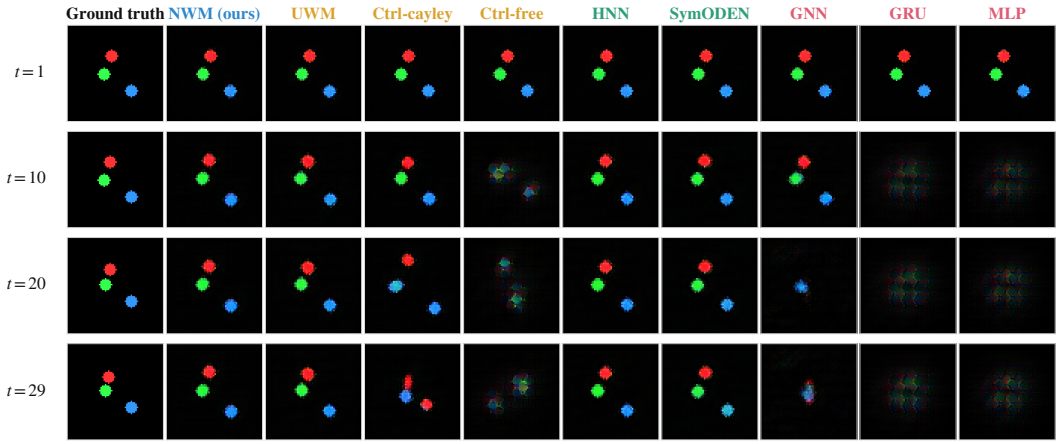

Figure 4: **Qualitative rollout on visual N-body.** One test trajectory decoded at four steps. Ctrl-cayley exchanges two objects by $t = 20$ although its latent norm has not moved, and the three unconstrained transitions collapse to a static reconstruction between $t = 10$ and $t = 20$.

4.3×10^6 at the same parameter count (Figure 5a), which Figure 4 shows in pixels as two objects exchanged by $t = 20$ while the norm has not moved. The separation is unchanged by rebuilding the partition for $N \in \{6, 12, 24\}$ or widening f_θ by a factor of 240, both of which hold the ratio within a factor of three while neither control leaves $[0.46, 1.76]$, and it is unchanged by relabelling the m coordinates inside every object, which moves the drift of $z^\dagger \mathbf{J} z$ from 7.6×10^{-5} only to 9.3×10^{-5} (Appendices D.4 and D.5). A commutator diagnostic confirms that no out-of-family form commutes with the learned $\mathbf{L}$ at any N (Appendix D.6), so the guarantee lies in the algebraic form of the generator rather than in the coupling network, its capacity, the partition or the coordinates. We next ask what the family is worth beyond the training distribution, and show in §4.3 that it flattens the degradation curve and that the two designs that are unitary without a Laplacian fall to the bottom under a change of force law.

4.3 Long Horizons and a Change of Force Law

Results. Exact conservation changes the shape of the degradation curve rather than its level, since over a rollout ten times the training horizon NWM falls by 0.68 dB against 1.38 dB for the best unconstrained transition and rises from seventh to fourth without overtaking the three above it, while HNN falls by 0.70 dB and is indistinguishable from NWM (Appendix D.7). A change of force law separates the designs more sharply, as Figure 5b shows, because under a zero-shot $1/r^2$ law NWM, HNN and SymODEN hold 20.3 to 20.4 dB while the MLP and the GRU fall to 18.5, a gap of 1.9 dB that moves NWM from third to second, whereas the two designs that are unitary without a Laplacian, Ctrl-cayley and UWM, rank last and second to last, indicating that structure of the wrong kind costs

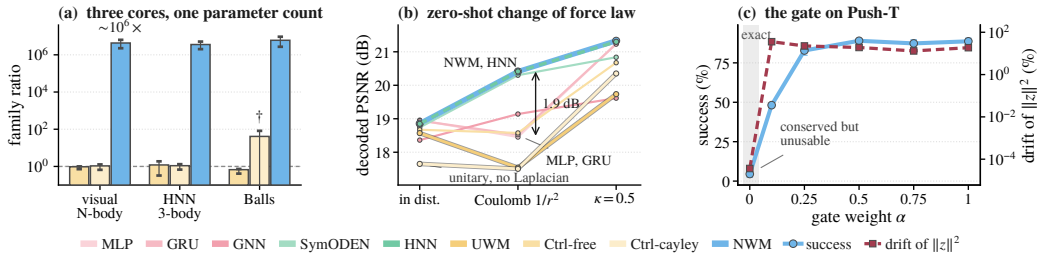

Figure 5: **Where the family comes from and what it buys.** (a) Family ratio at $h = 29$ over three seeds for three cores holding 3,945, 3,963 and 3,963 parameters, with the dashed line marking no separation and † as in Table 1. (b) Decoded PSNR in distribution and zero-shot under two shifts, the two designs that are unitary without a Laplacian outlined. (c) Planning success and drift of $\|z\|^2$ at $h = 7$ on Push-T against α in Equation 9, standard deviations over three evaluation seeds.

more than none. Rescaling the energy by $\kappa = 0.5$ leaves the force law intact and the MLP and the GRU recover to third and fourth, so what the Coulomb result measures is sensitivity to the force law rather than difficulty of evaluation (Appendix D.8). NWM therefore matches the robustness of a Hamiltonian prior without assuming a Hamiltonian, since the family of Corollary 2 is fixed by the block structure at every value of the couplings, and both evaluations replace the transition outright, so §4.4 turns to the regime in which the predictor must be kept.

4.4 Where the Operator Applies, and How the Invariants Travel

Results. The exact operator does not transfer to a JEPA latent, and capacity does not explain it, since at the host's own 10.8M parameters it plans at 3.17% while an unconstrained transition at 16,704 parameters reaches 14.83% (Appendix D.9). The cause is the constraint itself, because a unitary step confines the rollout to a sphere while the encoded norm of this latent moves by a median of 4.9% per frame, so most goals of a seven-step plan lie off that sphere before planning begins.

Table 2: **Where the penalty pays.** Drift of $\|z\|^2$ at $h = 7$ and change in planning success, Reacher paired over three training seeds.

		Drift at $h = 7$ (%)		
Env.	δ (%)	Host	+Penalty	ΔSucc.
Push-T	4.9	6.27	3.67	−0.50
Tworoom	5.2	31.51	21.03	**+8.50**
Reacher	1.5	6.78	5.14	−1.89±1.20

Figure 5c traces this boundary, since at $\alpha = 0$ the gated model holds $\|z\|^2$ to 3.6×10^{-5}% and plans at 4.5%, whereas $\alpha \geq 0.5$ saturates success near 89%, so the invariants have to enter through the objective instead. That route removes drift in every environment, by 42%, 33% and 24% on Push-T, Tworoom and Reacher under matched runs that differ in λ_H alone, yet it converts into planning only on Tworoom, where the host drifted five times more and success rises by 8.50 points, while on Push-T and Reacher the change lies inside the 1.83-point spread of retraining (Table 2). The gain therefore narrows with planning depth, from +8.50 points at $H = 5$ to +3.33 at $H = 10$ (Appendix D.10), and the two regimes divide along two criteria measurable in advance, since the per-frame change of the encoded norm of §3.4 decides whether the exact operator can replace the transition, and the host's accumulated drift decides whether the penalty is worth adding once it cannot.

5 Conclusion

We introduce the Noether World Model, a latent world model whose conservation guarantee is fixed by the algebraic form of the transition operator rather than by the objective that trains it. Because the Laplacian assembly and the Cayley step make an entire family of quadratic invariants a property of the operator, the guarantee holds at every parameter value and is unchanged by the partition, the capacity, the coordinates and the force law. Where a unitary step cannot replace the transition, the same invariants still reach a fixed predictor through its objective. Extending the commutant argument beyond quadratic forms, learning the partition jointly with the couplings, and bounding the cost of exactness for a general member of the family are the three directions the construction leaves open.

THE USE OF LARGE LANGUAGE MODELS (LLMS)

In this work, we used generative AI tools to edit the manuscript for readability. None of the tasks requiring disclosure are applicable to this work. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Martin Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. In *International Conference on Machine Learning*, pp. 1120–1128, 2016.

Mohammad Babaeizadeh, Mohammad Taghi Saffar, Suraj Nair, Sergey Levine, Chelsea Finn, and Dumitru Erhan. FitVid: Overfitting in pixel-level video prediction. *arXiv preprint arXiv:2106.13195*, 2021. doi: 10.48550/arXiv.2106.13195.

Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Yuhan Chen, Takashi Matsubara, and Takaharu Yaguchi. Neural symplectic form: Learning Hamiltonian equations on general coordinate systems. In *Advances in Neural Information Processing Systems*, volume 34, pp. 16659–16670, 2021.

Zhengdao Chen, Jianyu Zhang, Martin Arjovsky, and Léon Bottou. Symplectic recurrent neural networks. In *International Conference on Learning Representations*, 2020.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.026.

Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. Lagrangian neural networks. In *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2020.

Tal Daniel, Carl Qi, Dan Haramati, Amir Zadeh, Chuan Li, Aviv Tamar, Deepak Pathak, and David Held. Latent particle world models: Self-supervised object-centric stochastic dynamics modeling. In *International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=lTaPtGiUUc`.

Emily Denton and Rob Fergus. Stochastic video generation with a learned prior. In *International Conference on Machine Learning*, pp. 1174–1183, 2018.

Daniel DiPietro, Shiying Xiong, and Bo Zhu. Sparse symplectically integrated neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6074–6085, 2020.

Herbert Goldstein, Charles P. Poole, and John L. Safko. *Classical Mechanics*. Addison Wesley, 3rd edition, 2002.

Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. MaskViT: Masked visual pre-training for video prediction. In *International Conference on Learning Representations*, 2023.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

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

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, pp. 47376–47405, 2024.

Kyle Helfrich, Devin Willmott, and Qiang Ye. Orthogonal recurrent neural networks with scaled Cayley transform. In *International Conference on Machine Learning*, pp. 1969–1978, 2018.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. In *Advances in Neural Information Processing Systems*, volume 32, pp. 12519–12530, 2019.

Giancarlo Kerg, Kyle Goyette, Maximilian Puelma Touzel, Gauthier Gidel, Eugene Vorontsov, Yoshua Bengio, and Guillaume Lajoie. Non-normal recurrent neural network (nnRNN): learning long time dependencies while improving expressivity with transient dynamics. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Aditi S. Krishnapriyan, Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 26548–26560, 2021.

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

Alex X. Lee, Richard Zhang, Frederik Ebert, Pieter Abbeel, Chelsea Finn, and Sergey Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018.

Mario Lezcano-Casado and David Martínez-Rubio. Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group. In *International Conference on Machine Learning*, pp. 3794–3803, 2019.

Zhixuan Lin, Yi-Fu Wu, Skand Peri, Bofeng Fu, Jindong Jiang, and Sungjin Ahn. Improving generative imagination in object-centric world models. In *International Conference on Machine Learning*, pp. 6140–6149, 2020. URL `https://proceedings.mlr.press/v119/lin20f.html`.

Peter Y. Lu, Rumen Dangovski, and Marin Soljačić. Discovering conservation laws using optimal transport and manifold learning. *Nature Communications*, 14:4744, 2023. doi: 10.1038/s41467-023-40325-7.

Michael Lutter, Christian Ritter, and Jan Peters. Deep Lagrangian networks: Using physics as model prior for deep learning. In *International Conference on Learning Representations*, 2019.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. LeWorld-Model: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026. doi: 10.48550/arXiv.2603.19312.

Takashi Matsubara and Takaharu Yaguchi. FINDE: Neural differential equations for finding and preserving invariant quantities. In *International Conference on Learning Representations*, 2023.

Takashi Matsubara, Ai Ishikawa, and Takaharu Yaguchi. Deep energy-based modeling of discrete-time physics. In *Advances in Neural Information Processing Systems*, volume 33, pp. 13100–13111, 2020.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *International Conference on Learning Representations*, 2023.

Emmy Noether. Invariante Variationsprobleme. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, pp. 235–257, 1918.

Christian Offen and Sina Ober-Blöbaum. Symplectic integration of learned Hamiltonian systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 32(1):013122, 2022. doi: 10.1063/5.0065913.

Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: 10.1016/j.jcp.2018.10.045.

Reuven Y. Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 1(2):127–190, 1999.

Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *International Conference on Machine Learning*, pp. 9323–9332, 2021.

Uladzislau Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim G. J. Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Erik Talvitie. Self-correcting models for model-based reinforcement learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 2597–2603, 2017. doi: 10.1609/aaai.v31i1.10850. URL https://ojs.aaai.org/index.php/AAAI/article/view/10850.

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

Peter Toth, Danilo J. Rezende, Andrew Jaegle, Sébastien Racanière, Aleksandar Botev, and Irina Higgins. Hamiltonian generative networks. In *International Conference on Learning Representations*, 2020.

Tycho F. A. van der Ouderaa, Mark van der Wilk, and Pim de Haan. Noether's razor: Learning conserved quantities. In *Advances in Neural Information Processing Systems*, volume 37, pp. 135943–135965, 2024. URL https://openreview.net/forum?id=dpvqBkEp1f.

Eugene Vorontsov, Chiheb Trabelsi, Samuel Kadoury, and Chris Pal. On orthogonality and learning recurrent networks with long term dependencies. In *International Conference on Machine Learning*, pp. 3570–3578, 2017.

Hongbo Wang. When do conservation laws survive learned representations? Certified Horizons for latent world models. *arXiv preprint arXiv:2606.24945*, 2026.

Yaofeng Desmond Zhong, Biswadip Dey, and Amit Chakraborty. Symplectic ODE-Net: Learning Hamiltonian dynamics with control. In *International Conference on Learning Representations*, 2020.

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

Appendix

Table of Contents

A Extended Related Work **14**

B Datasets, implementation details, theorems, and notation **15**

C Proofs and Method Details **19**

C.1 Assumptions 19

C.2 Proofs 19

C.3 Remark on Real-Valued Latents 22

C.4 Method in Detail 22

C.5 What Exactly Is Conserved 25

D Experimental Details **25**

D.1 Evaluation protocol 25

D.2 Per-seed conservation measurements 26

D.3 Predictive accuracy 26

D.4 Perturbing the operator 27

D.5 Relabelling the coordinates 28

D.6 The boundary of the family 28

D.7 Long horizons 30

D.8 Transfer under a change of force law 30

D.9 Reachability in the host latent 31

D.10 The penalty weight 31

D.11 Numerical precision 32

D.12 The perceptual term 32

A Extended Related Work

Lines omitted from the main text. Dreamer has been scaled to a single agent across domains (Hafner et al., 2025) and IRIS (Micheli et al., 2023) predicts tokenised observations, and both learn the latent through reconstruction or reward prediction, which is the dependence the joint-embedding line of §2 removes. The Hamiltonian recipe has been extended by the Lagrangian form (Cranmer et al., 2020; Lutter et al., 2019), recurrent integration (Chen et al., 2020) and sparse symplectic identification (DiPietro et al., 2020), each of which keeps the guarantee in the continuous-time field, while methods that discover conserved quantities from data (Lu et al., 2023) make the quantity an output of learning that holds only to the accuracy of the fit.

Table 3: Conservation guarantees by design. *Exact* holds at every value of the weights. *Input* is what the update reads. *Family* counts what is conserved by construction.

Design	Conserved	Guarantee in	Exact	Input	Family
HNN, LNN, SymODEN, SRNN, HGN	learned H	continuous field	✗	state	1
Discrete gradient	learned H	implicit update	✓	state	1
FINDE	K learned V_k	implicit projection	✓	state	K
PINN	known residual	loss weight	✗	state	1
Noether's Razor	learned Q	Bayesian objective	✗	state	learned
Unitary RNN	$\|z\|^2$	fixed matrix	✓	none	1
UWM	$\|z\|^2$	unitary step	✓	action	1
EGNN	none	message structure	–	state	0
NWM (ours)	quadratic forms	algebraic form	✓	state, action	$2m^2$

Where the guarantee lives. The two designs of Table 3 that conserve exactly in discrete time do so through an implicit update, since the discrete-gradient method finds u^{s+1} from $H(u^{s+1}) - H(u^s) = \bar{\nabla} H^\top (u^{s+1} - u^s)$ and FINDE projects onto the level sets of K learned first integrals with discrete gradients, and in both the conserved quantities are scalars fixed in number before training. The three unitary designs differ in what the step reads, because a unitary recurrent network applies one fixed matrix and UWM a matrix that follows the action alone, so for both any invariant beyond the norm belongs to the matrix learned rather than to the design. Our operator builds its generator from the current state and the action, so the rotation differs at every step, and the conserved family is nonetheless the same at every step because the block structure fixes it, at the cost of one explicit matrix product.

Equivariance and conservation. Equivariance, $f(g \cdot z) = g \cdot f(z)$, constrains how the output moves with the input, whereas conservation, $Q(f(z)) = Q(z)$, constrains what stays fixed, and neither implies the other, since a translation-equivariant map may change $\sum_i f_i(z)$ and a generic unitary matrix commutes with no group action on the coordinates. Noether's theorem links the two only for Hamiltonian flows. Our conservation instead comes from the commutant of $\mathbf{H} = \mathbf{L} \otimes I_m$ (§3.3), which the block structure fixes at every value of the couplings, and the permutation equivariance the Laplacian also provides is not what it rests on.

Pixel-level video prediction. SVG′ (Denton & Fergus, 2018), SAVP (Lee et al., 2018) and FitVid (Babaeizadeh et al., 2021) predict frames through stochastic recurrent latents, MaskViT (Gupta et al., 2023) through masked tokens, and diffusion models (Ho et al., 2020) raise sample quality. They are evaluated over ten to thirty frames, where the drift of an integrated quantity is not yet visible, and are not baselines here because a pixel decoder is not part of the transition we replace, although the operator could be placed inside any of them whose latent can be partitioned into objects.

B DATASETS, IMPLEMENTATION DETAILS, THEOREMS, AND NOTATION

Table 4: **Glossary and notation.**

Symbol	Description
Latent and partition	
$d,\ N,\ m$	Latent dimension, subspace count, and subspace width, $d{=}Nm$
$z{=}(z_1,\dots,z_N)\in\mathbb{C}^d$	Latent state; $z_i{=}(q_i,p_i)\in\mathbb{C}^m$ with m even
$x_t,\ a_t,\ t,\ h$	Observation, action, time index, and rollout step
$\Phi,\ \Psi$	Encoder $x\mapsto z$ and decoder $z\mapsto\hat{x}$
Latent invariants and the family	
$\|z\|^2$, $\sum_i z_i$, $z^\dagger\mathbf{J}z$	Latent norm, block sum, and symplectic pairing, the invariants of the unitary, translation, and rotation groups
$\mathbf{J}{=}\operatorname{diag}(\mathbf{J}_2,\dots,\mathbf{J}_2)$	Symplectic form, $\mathbf{J}_2{=}\left(\begin{smallmatrix}0&-1\\1&0\end{smallmatrix}\right)$
$Q_{A,C}(z){=}z^\dagger(A\otimes C)z$	Family member; in-family means $A\in\operatorname{span}\{I_N,\mathbf{1}\mathbf{1}^\top\}$, out-of-family means A in the orthogonal complement of that span
$\mathcal{M}_0$, π	Set on which every family member keeps its value at z_0; relabelling of the m coordinates, applied identically to every object
physical charges	Energy, momentum, and angular momentum from simulator ground truth (q,p,m)
The operator	
$g_{ij}{=}f_\theta(\|z_i{-}z_j\|^2,\rho_\phi(a))$	Learned real coupling, symmetric in (i,j); f_θ,ρ_ϕ are the coupling network and action embedding
$\mathbf{L}\in\mathbb{R}^{N\times N}$	Graph Laplacian of the couplings, symmetric with zero row sums
$\mathbf{H}{=}\mathbf{L}\otimes I_m$	Generator: Hermitian, zero row sums, blocks proportional to I_m
$\operatorname{Cay}(\mathbf{H}){=}(I{+}\frac{i}{2}\mathbf{H})^{-1}(I{-}\frac{i}{2}\mathbf{H})$	Cayley transform, unitary for every Hermitian $\mathbf{H}$
$\mathbf{U}{=}\operatorname{Cay}(\mathbf{L})\otimes I_m$	Transition, applied as $z_{t+1}{=}\mathbf{U}(z_t,a_t)\,z_t$
Relaxations	
F_ψ, $\hat{z}_{t+1}{=}F_\psi(z_t,a_t)$	Free predictor of a host model and its prediction
$\alpha,\ \beta$	Gate between the operator at $\alpha{=}0$ and F_ψ at $\alpha{=}1$; weight on α when the gate is learned
$\mathcal{L}_Q,\ \lambda_H,\ \epsilon$	Relative-violation penalty of invariant Q, its weight, and the denominator offset
$\mathcal{L}_{\text{base}},\ \lambda_S$	Objective of the host model and the weight of its regulariser
δ	Median of $\big\|\,\|\Phi(x_{t+1})\|-\|\Phi(x_t)\|\,\big|/\|\Phi(x_t)\|$ over ground-truth trajectories, measured on the trained encoder
Evaluation	
$z_h^{\text{gt}}{=}\Phi(x_h)$	Ground-truth latent, the true frame encoded by the same model
norm deviation@h	$100\,\big\|\,\|\hat{z}_h\|-\|z_h^{\text{gt}}\|\,\big|/\|z_h^{\text{gt}}\|$. Compares a rollout against the truth, not against its own start
$\text{drift}_Q@h$	$100\,\|Q(\hat{z}_h)-Q(z_0)\|/(\|Q(z_0)\|+\epsilon)$, in per cent. Compares one rollout against its own start
family ratio	Mean drift of the out-of-family forms divided by mean drift of the in-family forms, per seed. The out-of-family forms are orthogonalised against the span before the ratio is taken
Cos@h, rel. MSE@h, PSNR@h	Latent-direction coherence, latent squared error over the variance of z_h^{gt}, and decoded-frame PSNR at step h
CEM success, r, R^2	Successful planning episodes out of 200 per seed; Pearson correlation and coefficient of determination against a physical quantity
$\kappa,\ \sigma,\ T,\ H,\ s$	Energy scale, action noise, rollout length, planning horizon, and seed, $s\in\{42,43,44\}$

Table 5: **Source of datasets used in this research.** Public datasets are used unmodified except for resizing to the stated resolution. Synthetic datasets come from our simulator, which stores the physical charges per frame.

Dataset	Details
Synthetic, structured physics	
visual N-body	Planar gravitational N-body, N=3, 64×64 RGB, 30-step training rollouts, 256 test trajectories, centre-of-mass frame so the physical momentum is identically zero. The primary in-distribution benchmark, also used with 100-step test rollouts for the long-horizon column and with the latent partitioned into $N \in \{3, 6, 12, 24\}$ subspaces.
Coulomb	Same renderer with $1/r^2$ electrostatic interactions, used for zero-shot transfer and as a separately trained benchmark.
Shifted evaluation sets	Test-time energy scale $\kappa \in \{0.5, 0.75, 1, 1.5, 2, 2.5, 3, 4\}$; object count $N \in \{2, 4, 5, 6\}$ zero-shot from N=3, with mixed-N training on $N \in \{3, 4, 5\}$; four-body elastic and inelastic collisions; action noise $\sigma \in \{0, 0.01, 0.1, 0.5\}$.
Public, structured physics	
HNN 3-body	Planar three-body problem from the HNN release (Greydanus et al., 2019), official script, 1000/100/100 trajectories of 100 steps, 64×64 RGB, per-frame (q, p, m). Non-zero centre-of-mass momentum, maximum 0.257. Also re-rendered with positions taken relative to the per-frame centre of mass.
Balls-Interaction	Bouncing balls from G-SWM (Lin et al., 2020) as distributed with LPWM (Daniel et al., 2026). Native 64×64, up to 15 objects, per-frame positions, velocities and masses, non-zero net momentum.
Public, control and planning	
Push-T	Planar block pushing from Diffusion Policy (Chi et al., 2023) under the LeWM protocol (Maes et al., 2026): 224×224 RGB, ViT-Tiny/14 encoder, CEM planning.
Tworoom, DMC Reacher	Two-room navigation as released with LeWM (Maes et al., 2026), and Reacher from the DeepMind Control Suite (Tassa et al., 2018), both under the same protocol.

Table 6: **Implementation details.** Every hyperparameter is fixed in advance and none is selected on an evaluation split. Each control is matched to the arm it tests, in parameter count or in warm start, and the scope column names what the control isolates.

Setting	Value	Scope
Structured latents, exact operator		
Latent, encoder, decoder	$z \in \mathbb{C}^{120}$, N=3 subspaces of width m=40; 4-layer CNN encoder, channels $[64, 128, 256, 512]$, single frame; U-Net decoder, linear skip decay, K=10	All structured latents
Coupling network, Cayley step	MLP $17 \to 64 \to 32 \to 1$ on $(\|z_i - z_j\|^2, \rho_\phi(a))$, real and symmetric in (i, j), 3,963 parameters; one float64 linear solve on the $N \times N$ matrix (float32 in Appendix D.11)	NWM, UWM
Optimiser, objective	Adam, 2×10^{-4}, batch 16, 30 epochs; $\mathcal{L}_{\mathrm{MSE}} + 0.01\,\mathcal{L}_{\mathrm{lat}} + 0.01\,\mathcal{L}_{\mathrm{VGG}}$, last term ablated in Appendix D.12	All structured latents
Structured latents, controls		
UWM	Unconstrained Hermitian $\mathbf{H} \in \mathbb{C}^{120 \times 120}$ from a 951K network reading the action alone, Cayley step on the full matrix	Isolates unitarity
CTRL-CAYLEY-3.9K	Same coupling network and Cayley step, Laplacian basis replaced by a fixed random Hermitian basis: entries from $\mathcal{N}(0, 1/d)$ under seed 0, symmetrised, frozen, shared across training seeds	Isolates the Laplacian structure
CTRL-FREE-3.9K	Unconstrained MLP transition, 3,945 parameters, otherwise the NWM recipe	Isolates the parameter count
Widened NWM	Coupling network widened to $\{50\text{K}, 200\text{K}, 950\text{K}\}$ parameters, the three structural residuals verified before training	Isolates capacity
Partition sweep	$N \in \{3, 6, 12, 24\}$, form bank rebuilt for each N	Isolates the partition
Permutation test	20 random relabellings of the m coordinates inside every subspace, same permutation seed across methods	Isolates the coordinate labelling
Pixel latents		
Host model	LEWM unchanged: ViT-Tiny/14 encoder, 12 layers, 192 dimensions, 5.5M; DiT-style predictor, 6 layers, 8 heads, AdaLN action conditioning, 10.8M; 18.0M in total	Push-T, Tworoom, Reacher
Noether penalty	λ_H=0.1 over N=3 equal-width blocks of the 192-dimensional token, added to the host objective with no architectural change (sensitivity in Appendix D.10)	NOETHER-LEWM
Warm start, optimiser	Both arms continue from the same LEWM 5-epoch checkpoint for 5, 15, 55 or 95 further epochs. AdamW, 5×10^{-5}, weight decay 10^{-3}, bf16, gradient clip 1.0, batch 256, seed 3072, inherited from LEWM	Push-T, Tworoom, Reacher
Gate	$z_{t+1} = (1 - \alpha)\,\mathbf{U}(z_t, a_t) z_t + \alpha F_\psi(z_t, a_t)$, $\alpha \in \{0, 0.1, 0.25, 0.5, 0.75, 1\}$ fixed; $\mathbf{U}$ from the widened Cayley branch, F_ψ the LEWM predictor, 10 epochs from the shared warm start	Isolates the cost of exactness
Capacity-matched arms	HARD-NATIVE 17K, HARD-WIDE 10.8M, FREE-SMALL 17K unconstrained, same warm start and budget as LEWM at 10 epochs	Isolates capacity on a pixel latent
Evaluation		
Rollout, form bank	30 steps in the main table, 100 for the long-horizon column, 200 on HNN 3-body; 100 quadratic forms per method and seed, 50 in-family and 50 out-of-family, the latter orthogonalised against the span by Gram–Schmidt before use, bank seed 12345, rebuilt for each partition	All experiments
CEM planner	300 candidates, 30 iterations, 30 elites, initial σ=1, horizon H=5, frame skip 5, inherited from LEWM (Rubinstein, 1999); 200 episodes per evaluation seed, seeds $\{42, 43, 44\}$ paired across arms	Pixel-latent planning

Table 7: **Formal results.** Every statement is proved in Appendix C.2 under the number it carries here and in the main text.

Result	Statement	Used for
What the operator conserves		
Theorem 1	For $\mathbf{H} = \mathbf{L} \otimes I_m$ assembled as in Equation 4 and $\mathbf{U} = \mathrm{Cay}(\mathbf{H})$, the three invariants $\|z\|^2$, $\sum_i z_i$ and $z^\dagger \mathbf{J} z$ are preserved for every z and every action. Hermiticity gives the first, zero row sums the second, and block-scalar structure the third, with no hypothesis shared between them	§3.3
Corollary 2	$\mathbf{U}$ preserves every quadratic form $z^\dagger(A \otimes C)z$ with $A \in \mathrm{span}\{I_N, \mathbf{1}\mathbf{1}^\top\}$ and C arbitrary, a family of $2m^2$ real dimensions, and a form preserved for every admissible $\mathbf{L}$ lies in that span. A single fixed $\mathbf{L}$ preserves more, but those forms move when the couplings move	§3.3, family ratio
Proposition 4	Relabelling the m coordinates inside every subspace keeps the invariants inside the family, so they remain exact. A unitary operator preserves every relabelled $z^\dagger \mathbf{J} z$ if and only if it has the block-scalar form $\mathbf{V} \otimes I_m$	Permutation test
When exactness applies		
Proposition 3	For every unitary $\mathbf{U}$ and every pair (z_t, z_{t+1}), $\|\mathbf{U}z_t - z_{t+1}\| \geq \big|\, \|z_{t+1}\| - \|z_t\| \,\big|$. The bound is tight, and it turns the encoded norm variation δ into a floor on the one-step error of any exact operator	§3.4
Relaxations		
Proposition 5	For $z' = (1-\alpha)\mathbf{U}z + \alpha F_\psi(z)$, the violation of any family invariant is bounded by α times the disagreement $\|F_\psi(z) - \mathbf{U}z\|$, up to a constant depending only on the form and on $\|z\|$	§3.5, gate sweep
Remark 6	On a real-valued latent, $z^\dagger \mathbf{J} z$ vanishes identically, so the corresponding penalty contributes no gradient and the soft variant enforces the norm and the block sum only	§3.5

C Proofs and Method Details

This appendix states the assumptions behind Section 3 (§C.1), restates and proves each formal result under its main-text number (§C.2), records a remark on real-valued latents (§C.3), gives the implementation behind each component (§C.4), and answers what the operator conserves (§C.5). Throughout, $\mathbf{H} = \mathbf{L} \otimes I_m$ denotes a generator of the form in Equation 4, $\mathbf{U} = \mathrm{Cay}(\mathbf{H})$, and $\mathbf{A} = \frac{\mathrm{i}}{2}\mathbf{H}$. For the three members of Equation 2 we write $E(z) = \|z\|^2$, $\mathbf{P}(z) = \sum_i z_i$, and $L(z) = \mathrm{Im}(z^\dagger \mathbf{J} z)$.

C.1 Assumptions

The results of Section 3 rest on the five assumptions of Table 8.

Table 8: Assumptions behind the results of Section 3.

Assumption	Statement	Used in
Partition	The latent has $d = Nm$ coordinates, partitioned into N object subspaces of equal width m. The exact regime requires m even, so that each subspace splits into $z_i = (q_i, p_i)$ and $\mathbf{J}_m = \mathrm{diag}(\mathbf{J}_2, \ldots, \mathbf{J}_2)$ is defined; the soft regime requires only the partition.	Eq. 4, Thm. 1
Couplings	The scalars $g_{ij} = f_\theta(\|z_i - z_j\|^2, \rho_\phi(a))$ are real and symmetric in (i, j); symmetry by evaluating each pair once, reality by the output layer of f_θ.	Hermiticity of $\mathbf{H}$
Invertibility	$I + \mathbf{A}$ is invertible for every Hermitian $\mathbf{H}$, since the spectrum of $\mathbf{A}$ is purely imaginary; no value of (θ, ϕ) makes the step undefined.	Eq. 5
Quantifier	The converse statements characterise the forms preserved for *every* admissible $\mathbf{L}$, that is, for every symmetric zero-row-sum matrix the assembly can produce. A single fixed $\mathbf{L}$ preserves more, since it commutes with the spectral projectors of its own spectrum, but those forms move when the couplings move.	Cor. 2, Prop. 4 (converse parts)
Size	$N \geq 3$ and $m \geq 4$; all experiments satisfy both (Table 6).	Cor. 2, Prop. 4

C.2 Proofs

Preliminaries. All proofs use two properties of the Cayley transform, which we isolate as lemmas. Throughout, $\mathbf{H}$ is Hermitian, $\mathbf{A} = \frac{\mathrm{i}}{2}\mathbf{H}$, and $\mathbf{U} = \mathrm{Cay}(\mathbf{H}) = (I + \mathbf{A})^{-1}(I - \mathbf{A})$.

Lemma C.1 (Unitarity). For every Hermitian $\mathbf{H}$, the matrix $I + \mathbf{A}$ is invertible and $\mathbf{U}$ is unitary.

Proof. Since $\mathbf{H}^\dagger = \mathbf{H}$, we have $\mathbf{A}^\dagger = -\frac{\mathrm{i}}{2}\mathbf{H} = -\mathbf{A}$, so $\mathbf{A}$ is skew-Hermitian and its spectrum lies on the imaginary axis. Hence -1 is not an eigenvalue of $\mathbf{A}$ and $I + \mathbf{A}$ is invertible. Skew-Hermiticity also gives $(I - \mathbf{A})^\dagger = I + \mathbf{A}$ and $\big((I + \mathbf{A})^{-1}\big)^\dagger = (I - \mathbf{A})^{-1}$, so $\mathbf{U}^\dagger = (I + \mathbf{A})(I - \mathbf{A})^{-1}$. The four factors $I \pm \mathbf{A}$ and $(I \pm \mathbf{A})^{-1}$ are rational functions of $\mathbf{A}$ and therefore commute, which yields

$$\mathbf{U}^\dagger \mathbf{U} = (I + \mathbf{A})(I - \mathbf{A})^{-1}(I + \mathbf{A})^{-1}(I - \mathbf{A}) = (I + \mathbf{A})(I + \mathbf{A})^{-1}\,(I - \mathbf{A})^{-1}(I - \mathbf{A}) = \boxed{I}.$$

□

Lemma C.2 (Commutant). If a matrix $\mathbf{B}$ satisfies $[\mathbf{H}, \mathbf{B}] = 0$, then $[\mathbf{U}, \mathbf{B}] = 0$.

Proof. $\mathbf{B}$ commutes with $\mathbf{A}$, hence with $I \pm \mathbf{A}$, hence with $(I + \mathbf{A})^{-1}$ (multiply $[\mathbf{B}, I + \mathbf{A}] = 0$ on both sides by $(I + \mathbf{A})^{-1}$), and therefore with the product $\mathbf{U}$. □

Lemma C.2 is the mechanism behind every conservation statement below: a quadratic form $z^\dagger \mathbf{B} z$ is preserved by a unitary $\mathbf{U}$ exactly when $\mathbf{U}^\dagger \mathbf{B} \mathbf{U} = \mathbf{B}$, that is, when $[\mathbf{U}, \mathbf{B}] = 0$, and the lemma transfers that condition from $\mathbf{U}$ to the generator $\mathbf{H}$, whose structure is what the assembly fixes.

Theorem 1 (Exact conservation by construction). Let $\mathbf{H}$ satisfy Equation 4 and $\mathbf{U} = \mathrm{Cay}(\mathbf{H})$. Then for every $z \in \mathbb{C}^d$ and every action a,

$$E(\mathbf{U}z) = E(z), \qquad \mathbf{P}(\mathbf{U}z) = \mathbf{P}(z), \qquad L(\mathbf{U}z) = L(z).$$

Proof. Each of the three structural properties of Equation 4 yields one identity.

Step 1 (Hermiticity gives the norm). Real symmetric couplings make $\mathbf{H}$ Hermitian, so Lemma C.1 applies and

$$E(\mathbf{U}z) = z^\dagger \mathbf{U}^\dagger \mathbf{U} z = z^\dagger z = E(z).$$

Step 2 (Zero row sums give the block sum). Fix $e \in \mathbb{C}^m$ and set $v = \mathbf{1}_N \otimes e$. Because every block row of $\mathbf{H}$ sums to the zero block,

$$(\mathbf{H}v)_i = \sum_j \mathbf{H}_{ij}\, e = \Big(\sum_j g_{ij}\Big) e = 0,$$

so $\mathbf{H}v = 0$, $\mathbf{A}v = 0$, and $(I \pm \mathbf{A})v = v$. Applying $(I + \mathbf{A})^{-1}$ gives $\mathbf{U}v = v$, and the same argument for $\mathbf{U}^{-1} = (I - \mathbf{A})^{-1}(I + \mathbf{A})$ gives $\mathbf{U}^{-1}v = v$. Since $\mathbf{U}$ is unitary, $\mathbf{U}^\dagger v = \mathbf{U}^{-1} v = v$, and for every z

$$\langle v, \mathbf{U}z \rangle = \langle \mathbf{U}^\dagger v, z \rangle = \langle v, z \rangle.$$

Choosing e to be the k-th standard basis vector turns $\langle v, z \rangle$ into $\sum_i z_{i,k} = \mathbf{P}_k(z)$, so every component of the block sum is preserved: $\mathbf{P}(\mathbf{U}z) = \mathbf{P}(z)$.

Step 3 (Block-scalar form gives the symplectic form). Write $\mathbf{J} = I_N \otimes \mathbf{J}_m$. Because $\mathbf{H} = \mathbf{L} \otimes I_m$,

$$[\mathbf{H}, \mathbf{J}] = (\mathbf{L} \otimes I_m)(I_N \otimes \mathbf{J}_m) - (I_N \otimes \mathbf{J}_m)(\mathbf{L} \otimes I_m) = \mathbf{L} \otimes \mathbf{J}_m - \mathbf{L} \otimes \mathbf{J}_m = 0.$$

Lemma C.2 gives $[\mathbf{U}, \mathbf{J}] = 0$, and with the unitarity of Step 1,

$$L(\mathbf{U}z) = \mathrm{Im}\big(z^\dagger \mathbf{U}^\dagger \mathbf{J} \mathbf{U} z\big) = \mathrm{Im}\big(z^\dagger \mathbf{U}^\dagger \mathbf{U} \mathbf{J} z\big) = \mathrm{Im}\big(z^\dagger \mathbf{J} z\big) = L(z).$$

Step 3 uses the unitarity of Step 1; otherwise the three steps rest on three different properties of $\mathbf{H}$, and all three hold for every value of the couplings, so the three identities hold at once and for every (θ, ϕ). □

Corollary 2 (Invariant family). Let $\mathbf{U}$ be as in Theorem 1. For every $A \in \mathrm{span}\{I_N, \mathbf{1}_N \mathbf{1}_N^\top\}$ and every $C \in \mathbb{C}^{m \times m}$, the quadratic form $Q_{A,C}(z) = z^\dagger (A \otimes C) z$ satisfies

$$Q_{A,C}(\mathbf{U}z) = Q_{A,C}(z),$$

and a quadratic form preserved by $\mathrm{Cay}(\mathbf{L}) \otimes I_m$ for every admissible $\mathbf{L}$ lies in this span.

Proof. Step 1 (Every member is preserved). The Laplacian satisfies $\mathbf{L}\mathbf{1} = 0$ and, by symmetry, $\mathbf{1}^\top \mathbf{L} = 0$, so $\mathbf{L}\,\mathbf{1}\mathbf{1}^\top = 0 = \mathbf{1}\mathbf{1}^\top \mathbf{L}$, and $[\mathbf{L}, A] = 0$ for every A in the span. Then

$$[\mathbf{H}, A \otimes C] = (\mathbf{L}A - A\mathbf{L}) \otimes C = 0,$$

so $[\mathbf{U}, A \otimes C] = 0$ by Lemma C.2 and

$$Q_{A,C}(\mathbf{U}z) = z^\dagger \mathbf{U}^\dagger (A \otimes C) \mathbf{U} z = z^\dagger (A \otimes C) \mathbf{U}^\dagger \mathbf{U} z = Q_{A,C}(z).$$

Step 2 (Where the three members sit). E is the member (I_N, I_m) and $L = \mathrm{Im}\, Q_{I_N, \mathbf{J}_m}$. The block sum $\mathbf{P}$ is linear, so it is not itself a member; it is preserved by Step 2 of Theorem 1, and its quadratic image $\mathbf{P}(z)^\dagger C\, \mathbf{P}(z) = z^\dagger (\mathbf{1}\mathbf{1}^\top \otimes C) z$ is the member $(\mathbf{1}\mathbf{1}^\top, C)$. Counting real dimensions, the Hermitian C contribute m^2 real-valued invariants for each of the two choices of A, which gives the dimension $2m^2$ stated in the main text.

Step 3 (The converse). Let $\mathcal{S}_0$ be the space of real symmetric $N \times N$ matrices with zero row sums, which is the set of admissible $\mathbf{L}$. A quadratic form $z^\dagger \mathbf{B} z$ is preserved by $\mathbf{U}$ for all z if and only if $\mathbf{U}^\dagger \mathbf{B} \mathbf{U} = \mathbf{B}$, that is, if and only if $[\mathbf{U}, \mathbf{B}] = 0$, because $\mathbf{U}$ is unitary. The Cayley map is invertible on its image through

$$\mathbf{H} = -2\mathrm{i}\,(I - \mathbf{U})(I + \mathbf{U})^{-1},$$

so $\mathbf{B}$ commutes with $\mathrm{Cay}(\mathbf{L}) \otimes I_m$ if and only if it commutes with $\mathbf{L} \otimes I_m$. A form preserved for every admissible $\mathbf{L}$ must therefore commute with $\mathbf{L} \otimes I_m$ for every $\mathbf{L} \in \mathcal{S}_0$, and the commutant of $\{\mathbf{L} \otimes I_m : \mathbf{L} \in \mathcal{S}_0\}$ equals $\mathrm{Comm}(\mathcal{S}_0) \otimes \mathbb{C}^{m\times m}$. In the orthogonal decomposition $\mathbb{R}^N = \mathrm{span}\{\mathbf{1}\} \oplus \mathbf{1}^\perp$, every $\mathbf{L} \in \mathcal{S}_0$ acts as $0 \oplus M$ with M an arbitrary symmetric $(N-1)\times(N-1)$ matrix, and symmetric matrices of size at least two generate the full matrix algebra on $\mathbf{1}^\perp$. Hence

$$\mathrm{Comm}(\mathcal{S}_0) = \mathrm{span}\{P_{\mathbf{1}}, P_\perp\} = \mathrm{span}\{I_N, \mathbf{1}\mathbf{1}^\top\} \qquad (N \geq 3),$$

and the forms preserved for every admissible $\mathbf{L}$ are exactly the family. A single fixed $\mathbf{L}$ preserves more, since it commutes with the spectral projectors of its own spectrum and with every polynomial in itself, but those forms change when the couplings change. Only the family survives every value of (θ, ϕ), which is the quantifier the statement carries. □

Proposition 4 (Permutation stability)**.** *Let Π be an $m\times m$ permutation matrix and $\pi(z) = (I_N \otimes \Pi)z$ the corresponding relabelling of coordinates inside every object subspace.*

(a) For $\mathbf{U}$ as in Theorem 1, every $Q_{A,C} \circ \pi$ is preserved exactly; in particular $E \circ \pi$, $\mathbf{P} \circ \pi$, and $L \circ \pi$ are preserved for every Π.

(b) Let $m \geq 4$ be even and let $\mathbf{W}$ be any unitary operator. Then $\mathbf{W}$ preserves $L \circ \pi$ for every permutation Π if and only if $\mathbf{W} = \mathbf{V} \otimes I_m$ for some unitary $\mathbf{V}$.

Proof. Part (a). Since $\Pi^\top = \Pi^{-1}$,

$$Q_{A,C}(\pi(z)) = z^\dagger(I_N \otimes \Pi^\top)(A \otimes C)(I_N \otimes \Pi)z = z^\dagger\big(A \otimes \Pi^\top C \Pi\big)z = Q_{A,\Pi^\top C\Pi}(z),$$

which is again a member of the family, so Corollary 2 applies.

Part (b), necessity. The permuted form is $L \circ \pi(z) = \mathrm{Im}\big(z^\dagger(I_N \otimes \Pi^\top \mathbf{J}_m \Pi)z\big)$. By the argument of Step 3 above, $\mathbf{W}$ preserves it for all z if and only if $[\mathbf{W}, I_N \otimes \Pi^\top \mathbf{J}_m \Pi] = 0$. Preservation for every Π therefore requires $\mathbf{W}$ to commute with $I_N \otimes S$ for every S in the linear span of $\{\Pi^\top \mathbf{J}_m \Pi\}$. Each $\Pi^\top \mathbf{J}_m \Pi$ is a sum of $m/2$ disjoint elementary skew matrices $\pm(E_{ab} - E_{ba})$ over a perfect matching of $\{1, \ldots, m\}$, and swapping the two elements of one pair flips the sign of that term alone. Subtracting the two matrices obtained from one matching with and without such a swap isolates a single $E_{ab} - E_{ba}$, and varying the matching reaches every pair (a, b) whenever $m \geq 4$. Hence the span is the full space $\mathfrak{so}(m)$ of real skew-symmetric matrices. Because $\mathfrak{so}(m)$ acts irreducibly on $\mathbb{C}^m$ for $m \geq 3$, the commutant of $I_N \otimes \mathfrak{so}(m)$ in $\mathbb{C}^{d\times d}$ is $\mathbb{C}^{N\times N} \otimes \mathbb{C}I_m$. Therefore $\mathbf{W} = \mathbf{V} \otimes I_m$, and $\mathbf{V}$ is unitary because $\mathbf{W}$ is.

Part (b), sufficiency. An operator of the form $\mathbf{V} \otimes I_m$ commutes with $I_N \otimes S$ for every S, so part (a) applies to it. □

Part (b) is the statement behind the permutation test of §4.2. An operator that conserves the labelled L but not its relabelled versions has no block-scalar structure, so its conservation of L is a property of the coordinate labelling rather than of the operator.

Proposition 3 (Lower bound for any exact operator). For every unitary $\mathbf{U}$ and every pair (z_t, z_{t+1}),

$$\|\mathbf{U}z_t - z_{t+1}\| \;\geq\; \big|\, \|z_{t+1}\| - \|z_t\| \,\big|.$$

Proof. The reverse triangle inequality gives $\|\mathbf{U}z_t - z_{t+1}\| \geq \big|\, \|\mathbf{U}z_t\| - \|z_{t+1}\| \,\big|$, and $\|\mathbf{U}z_t\| = \|z_t\|$ by Lemma C.1. The bound is attained when z_{t+1} is a positive multiple of $\mathbf{U}z_t$, so it is tight. Applied to each encoded pair $(\Phi(x_t), \Phi(x_{t+1}))$ along a trajectory, it gives a per-step error floor whose median is $\delta\,\|\Phi(x_t)\|$, with δ as in Equation 8. The floor is a property of the encoder and the data, and no choice of $\mathbf{L}$, θ, or ϕ lowers it. □

Proposition 5 (Violation bound for the interpolated operator). *Let $z' = (1-\alpha)\,\mathbf{U}z + \alpha\,F_\psi(z)$ with $\mathbf{U}$ as in Theorem 1, and write $\Delta = F_\psi(z) - \mathbf{U}z$. For every quadratic member $Q(w) = w^\dagger \mathbf{B} w$ of the family,*

$$|Q(z') - Q(z)| \le \alpha\,\|\mathbf{B}\|\,\|\Delta\|\big(2\|z\| + \alpha\|\Delta\|\big),$$

and for every linear member $\ell(w) = \langle v, w\rangle$, $|\ell(z') - \ell(z)| \le \alpha\,\|v\|\,\|\Delta\|$.

Proof. Since $z' = \mathbf{U}z + \alpha\Delta$ and $Q(\mathbf{U}z) = Q(z)$ by Corollary 2,

$$Q(z') - Q(z) = \alpha\big(\Delta^\dagger \mathbf{B}\mathbf{U}z + (\mathbf{U}z)^\dagger \mathbf{B}\Delta\big) + \alpha^2\,\Delta^\dagger \mathbf{B}\Delta.$$

The bracket is bounded in modulus by $2\|\mathbf{B}\|\|\Delta\|\|\mathbf{U}z\|$ with $\|\mathbf{U}z\| = \|z\|$, and the last term by $\alpha^2\|\mathbf{B}\|\|\Delta\|^2$. The linear case follows from $\ell(\mathbf{U}z) = \ell(z)$ and $\ell(z') - \ell(\mathbf{U}z) = \alpha\langle v, \Delta\rangle$. At $\alpha = 0$ both bounds vanish and Theorem 1 is recovered; at $\alpha = 1$ they reduce to the unconstrained violation of F_ψ. □

C.3 Remark on Real-Valued Latents

Remark 6 (Degeneracy of L on real latents). The charge $L(z) = \mathrm{Im}(z^\dagger \mathbf{J} z)$ is defined for complex z. JEPA encoders such as the ViT of LeWM produce real-valued embeddings. For real z every imaginary part vanishes, so $L(z) = \mathrm{Im}(z^\top \mathbf{J} z) = 0$ identically, and the L term of Equation 10 contributes no gradient. The Noether penalty on pixel latents therefore acts on E and $\mathbf{P}$ alone (§4.4).

C.4 Method in Detail

The subsections below follow §3.2, the controls used in §4, §3.4, and §3.5 of the main text, and give the implementation behind each.

C.4.1 The Laplacian–Cayley operator

Computing the Cayley step. Because $\mathbf{H} = \mathbf{L} \otimes I_m$, the Cayley transform reduces to an $N \times N$ problem, $\mathbf{U} = \mathrm{Cay}(\mathbf{L}) \otimes I_m$. We form $\mathrm{Cay}(\mathbf{L})$ by a linear solve rather than an explicit inverse, in float64, and apply it to the N object blocks of z by broadcasting (Algorithm 1, lines 5–6). The arithmetic cost is $O(N^3 + d)$ per step, but the float64 solve carries a fixed kernel overhead that a FLOP count does not capture, and at $N = 3$ that overhead dominates: one forward and backward pass at batch 64 takes 18.6 ms against 1.9 ms for a multilayer perceptron. The same solve on a different basis is slower still, at 21.0 ms for Ctrl-cayley, so the overhead belongs to the solve and not to the assembly. A closed-form inverse of the 3×3 matrix removes it.

The coupling network. For N objects the generator needs $\binom{N}{2}$ real scalars g_{ij}. Each is produced by f_θ from the 17-dimensional input $(\|z_i - z_j\|^2,\ \rho_\phi(a))$, where ρ_ϕ embeds the action into 16 dimensions. The default f_θ is a $17 \to 64 \to 32 \to 1$ MLP, giving the 3,963 predictor parameters reported in Table 1. Because g_{ij} depends on $\|z_i - z_j\|^2$ only, the coupling is invariant to a common rotation of the two objects, which is what makes g_{ij} a scalar rather than an $m\times m$ block.

Widened operators. To vary predictor capacity while keeping every constraint, we widen f_θ and ρ_ϕ to reach $\{50\text{K}, 200\text{K}, 950\text{K}\}$ parameters. Widening changes only how the scalars g_{ij} are computed, so Hermiticity, zero row sums, and block-scalar structure are unaffected. Before training each widened model we verify the three residuals numerically (Table 9). The same residuals on the Cayley control, which keeps the couplings and drops the Laplacian, show that the assembly and not the Cayley step is what supplies the structure.

C.4.2 Controls

The unitary control (UWM). UWM is the special case of §3.2 with the Laplacian structure removed. Its generator is an unconstrained Hermitian $\mathbf{H}(a_t) \in \mathbb{C}^{d\times d}$ produced from the action alone by a network with 950K parameters, and the transition is $\mathrm{Cay}(\mathbf{H})$ computed on the full $d\times d$ matrix. Because the generator does not read z_t, we have $\partial\mathbf{U}/\partial z = 0$ and the Jacobian of the step is $\mathbf{U}$ itself. The step is unitary, so E is preserved exactly, and no other structure is imposed. We report UWM as

Table 9: Structural residuals of the generator before training, in float32. Hermitian residual $\|\mathbf{H} - \mathbf{H}^\dagger\|_F$, row-sum residual $\|\mathbf{H}\mathbf{1}\|_F$, block-scalar residual $\|\mathbf{H} - \mathbf{L} \otimes I_m\|_F$ with $\mathbf{L}$ the block trace. Mean over three seeds. The widened operators inherit the structure of the assembly at every width; the Cayley control, which places the same scalars in a fixed random Hermitian basis, has no zero-row-sum or block-scalar structure, and its residuals grow with N.

Operator	Hermitian	Row sum	Block-scalar
NWM-wide-50K	0	1.2×10^{-7}	0
NWM-wide-200K	0	2.0×10^{-7}	0
NWM-wide-950K	0	9.9×10^{-8}	0
Ctrl-cayley, $N = 3$	0	1.45	0.87
Ctrl-cayley, $N = 6$	0	3.81	2.43
Ctrl-cayley, $N = 12$	0	6.72	3.90
Ctrl-cayley, $N = 24$	0	15.9	8.12

a control for one question: whether unitarity alone accounts for the conservation of NWM. It is the action-conditioned form of a unitary RNN and not a method of its own. The family-boundary test of §4.1 returns a ratio near one for UWM, as for every non-Laplacian operator, which answers the question in the negative.

Same-parameter controls. Two operators with the same 3,963 parameters as NWM but without the Laplacian structure serve as controls in Table 1 and Table 13. CTRL-CAYLEY-3.9K reuses f_θ and ρ_ϕ unchanged and produces the same $\binom{N}{2}$ scalars, but assembles the generator as $\mathbf{H} = \sum_k g_k \mathbf{B}_k$ with $\mathbf{B}_k$ a fixed random Hermitian basis: entries drawn once from $\mathcal{N}(0, 1/d)$ under seed 0, symmetrised to $\frac{1}{2}(\mathbf{B} + \mathbf{B}^\dagger)$, and frozen, with the same basis shared across all training seeds. The Cayley step then still preserves E, while the zero-row-sum and block-scalar properties are absent (Table 9). The only difference from NWM is therefore the basis in which the scalars are placed. CTRL-FREE-3.9K replaces the operator by an unconstrained MLP transition $z_{t+1} = F_\psi(z_t, a_t)$ with two hidden layers of widths $(9, 6)$, which is the smallest two-layer configuration within 1% of the target count.

C.4.3 THE ONE-STEP FLOOR

Computing δ. Equation 8 is evaluated on the encoder alone, after training. For each dataset or environment we take the encoder the host model uses, encode 256 ground-truth trajectories of $\mathrm{hist} + 8$ frames with the history length of the host, compute the per-step relative change of the encoded norm, and take the median over all steps and trajectories. The procedure is repeated for three sample seeds, and the reported δ is the mean over seeds. On the physics datasets the encoder is the one trained jointly with the operator; on the planning environments it is the LeWM encoder of the host, which is the encoder the penalty acts on. On visual N-body the median falls on a pair of frames in which nothing moves, since 55% of consecutive frames are identical, and over the frames in which something does move it is 2.1%. The measured values are 2.1%, 4.9% and 3.3% on the three physics datasets and 4.9%, 5.2% and 1.5% on Push-T, Tworoom and Reacher. The six values are of the same order, and what distinguishes the two settings is whether the encoder is trained with the operator, which §4.3 compares directly.

C.4.4 GATE AND NOETHER PENALTY

The gated operator. Equation 9 uses the widened Cayley branch for $\underbrace{\mathbf{U}}_{\text{exact}}$ and the LeWM predictor for $\underbrace{F_\psi}_{\text{free}}$. The two branches receive the same (z_t, a_t) and their outputs are mixed by the scalar α, which is a hyperparameter swept over $\{0, 0.1, 0.25, 0.5, 0.75, 1\}$. Each value is trained as a separate model from the same warm start, so the six points of §4.3 are six trainings rather than one model evaluated at six settings.

The Noether penalty on pixel latents. The LeWM predictor emits a d-dimensional embedding, which we partition into $N = 3$ subspaces of width $m = d/3$ before computing E and $\mathbf{P}$; L vanishes

on the real embedding (Remark 6). The penalty of Equation 10 is computed on the one-step prediction and averaged over the batch, with $\underbrace{\lambda_H = 0.1}_{\text{penalty weight}}$. The host objective is the LeWM objective without change,

$$\mathcal{L}_{\text{base}}^{\text{LeWM}} = \underbrace{\mathcal{L}_{\text{pred}}}_{\text{next-embedding MSE}} + \lambda_S \underbrace{\mathcal{L}_{\text{SIGReg}}}_{\text{anti-collapse}},$$

with λ_S the SIGReg weight of the host configuration. The optimiser, the schedule, and the SIGReg regulariser are inherited from LeWM without change; the penalty adds no parameters.

C.4.5 Training procedure

The two regimes are trained differently, and Algorithms 2 and 3 give one step of each. On the physics datasets the encoder, the operator, and a decoder are trained jointly from scratch under a pixel objective on a 10-step rollout; the same recipe is used for all nine transitions, and only the core changes. On the planning environments the host is LeWM, its objective is kept unchanged, and the penalty is the only addition.

Algorithm 2 One training step on a physics dataset (all nine transitions)

Require: frame x_0, target frames $x_{1:10}$, actions $a_{0:9}$, encoder Φ, transition core, decoder Ψ

$z_0 \leftarrow \Phi(x_0)$, partitioned into N subspaces
for $t = 0, \dots, 9$ **do**
  $z_{t+1} \leftarrow \text{core}(z_t, a_t)$ ▷ Algorithm 1 for NWM; a free map for the controls
  $\hat{x}_{t+1} \leftarrow \Psi(z_{t+1})$
end for
$\mathcal{L}_{\text{pix}} \leftarrow \frac{1}{10}\sum_t \|\hat{x}_{t+1} - x_{t+1}\|^2$
$\mathcal{L}_{\text{vgg}} \leftarrow \frac{1}{10}\sum_t \|\text{VGG}(\hat{x}_{t+1}) - \text{VGG}(x_{t+1})\|^2$ ▷ VGG19 features up to layer 16, frozen
$\mathcal{L}_{\text{scale}} \leftarrow \frac{1}{10}\sum_t \text{mean}\,|z_{t+1}|^2$ ▷ latent scale regulariser
$\mathcal{L} \leftarrow \mathcal{L}_{\text{pix}} + 0.01\,\mathcal{L}_{\text{vgg}} + 0.01\,\mathcal{L}_{\text{scale}}$
update $(\Phi, \text{core}, \Psi)$ by a clipped gradient step on $\mathcal{L}$

No term of Algorithm 2 refers to a latent invariant. The pixel and perceptual terms act on decoded frames, and the scale term keeps the latent magnitude bounded without fixing it along a trajectory, so whatever a transition conserves on these datasets it conserves by its own structure and not by the objective. The encoder reads a single frame; the rollout supplies the temporal signal. Training uses Adam with learning rate 2×10^{-4}, cosine decay over 30 epochs, batch size 16, and gradient clipping at norm 1. The three residuals of Table 9 are verified on random inputs before training begins.

Algorithm 3 One training step with the Noether penalty on a host world model

Require: batch of observations $x_{0:T}$ and actions $a_{0:T-1}$, encoder Φ, host predictor F_ψ, weights λ_H, λ_S

$z_t \leftarrow \Phi(x_t)$ for $t = 0, \dots, T$, each partitioned into N subspaces
$\hat{z}_{t+1} \leftarrow F_\psi(z_t, a_t)$ for $t = 0, \dots, T-1$ ▷ one-step prediction
$\mathcal{L}_{\text{pred}} \leftarrow \frac{1}{T}\sum_t \|\hat{z}_{t+1} - z_{t+1}\|^2$
$\mathcal{L}_{\text{SIGReg}} \leftarrow \text{SIGReg}(z_{0:T})$ ▷ host regulariser, unchanged
$\mathcal{L}_{\text{cons}} \leftarrow \frac{1}{T}\sum_t \sum_{Q \in \{E, \mathbf{P}\}} \left(\frac{Q(\hat{z}_{t+1}) - Q(z_t)}{|Q(z_t)| + \epsilon}\right)^2$ ▷ Eq. 10
$\mathcal{L} \leftarrow \mathcal{L}_{\text{pred}} + \lambda_H \mathcal{L}_{\text{cons}} + \lambda_S \mathcal{L}_{\text{SIGReg}}$
update (Φ, F_ψ) by a gradient step on $\mathcal{L}$

Line 5 of Algorithm 3 is the only line the penalty adds. The host is warm-started from the LeWM checkpoint after five epochs and trained for the same number of further epochs with and without the penalty. The two arms of §4.4 share the checkpoint, the data order, the optimiser state, and every hyperparameter; λ_H is the only difference between them.

C.5 What Exactly Is Conserved

The three members inside the family. Corollary 2 places E and L inside the family as the members (I_N, I_m) and $(I_N, \mathbf{J}_m)$. The block sum $\mathbf{P}$ is linear and lies outside the span of quadratic forms; it is conserved by the zero row sums instead, and its squared norm $\|\mathbf{P}\|^2$ is the member $(\mathbf{1}_N\mathbf{1}_N^\top, I_m)$. Conserving the vector is the stronger of the two statements, since a fixed vector has a fixed norm while a fixed norm still allows the vector to rotate, so Theorem 1 says more about $\mathbf{P}$ than the family alone does. The boundary of the family is tested in §4.1 in two ways, by sampling quadratic forms from the span and from its orthogonal complement, and by permuting the m coordinates inside every object, which maps C to $\Pi^\top C\Pi$ and stays in the span (Proposition 4).

What the latent invariants encode. Corollary 2 fixes the algebraic shape of the invariants; the encoder fixes their content. A probe over trajectory constants regresses each physical constant of a scene on the latent invariant at $t = 0$ and reports R^2 over three seeds. The latent E encodes the initial spatial extent of the scene, with R^2 of 0.61 ± 0.04 on visual N-body and 0.64 ± 0.25 on HNN 3-body, which the operator then holds fixed along the rollout. The latent $\|\mathbf{P}\|$ encodes the same extent on visual N-body (R^2 0.44) and nothing we tested on HNN 3-body, and L encodes no tested constant on either. The free control does not encode the extent in its latent norm on visual N-body (R^2 -0.02) and encodes it more strongly than NWM on HNN 3-body (0.87), but in neither case does it hold it fixed. What the operator supplies is an exactly conserved family of latent quantities; whether a member of that family also carries a physical charge is a property of the encoder, and the probe above is what measures it. The results of §4 are measured on the latent invariants and on downstream planning.

D Experimental Details

The subsections below follow the four questions of §4 in order. Training is described in Appendix C.4 and is not repeated here, so what follows concerns evaluation alone.

D.1 Evaluation protocol

On the three physics datasets every predictor is rolled forward 29 steps in double precision from a single encoded frame, and the drift of a quantity Q is read as $|Q(z_{29}) - Q(z_0)|/|Q(z_0)|$ and reported in %. The three seeds are 42, 43 and 44, and a seed governs the initialisation, the data order and the evaluation split together, so the three runs of a method share nothing beyond the architecture. The quadratic-form bank holds 100 Hermitian forms drawn once from a fixed generator seed and reused across methods, 50 of them from the span of Corollary 2 and 50 from its orthogonal complement, with each form normalised to unit Frobenius norm so that the two halves are comparable. The family ratio is formed per seed and then averaged over the three, rather than being formed from the averaged drifts, because the in-family drift of NWM is small enough that the two orders of operation differ by a factor of several.

On the planning environments Push-T, Tworoom and Reacher the host is LeWM and its checkpoint, data order and optimiser state are shared between the two arms of every comparison, with λ_H the only difference. Planning uses CEM with 200 episodes per evaluation seed at each environment's own planning horizon, which is seven steps on Push-T and five on Tworoom, while the drift is read at $h = 7$ in all three so that the environments share one scale. The evaluation seeds are again 42, 43 and 44. Push-T and Tworoom are trained from one seed and evaluated under three, so their reported spread is over evaluation only, whereas Reacher is trained under three seeds and its difference is paired across them. The weight is $\lambda_H = 0.1$ in every environment and the host regulariser keeps its own weight $\lambda_S = 0.09$, and §D.10 reports the sweep that fixes the former. Because a penalty can be present in the objective and still contribute nothing, the realised weight was recovered from the loss components of three validation batches rather than read from the configuration, and it agrees with 0.1 to within single-precision round-off, at a conservation term of the same order as the prediction term. The volume column of Table 1 is the log-determinant of the full one-step Jacobian, including $\partial\mathbf{U}/\partial z$, averaged over 64 anchor states on visual N-body at seed 42 in double precision, with SymODEN and HNN evaluated by central differences at step 10^{-6} because both cores detach their own input and autograd returns exactly zero and exactly the identity for them.

D.2 PER-SEED CONSERVATION MEASUREMENTS

Table 10 lists every bar in the top row of Figure 3, and Table 11 every point in the bottom row, both per seed. The main text quotes the mean over the three seeds, so a value there is the average of the three rows for that method here. Both tables are read from the same rollouts and the same checkpoints as the figure.

Table 10: **Invariant drift at $h = 29$ (%), per seed.**

		visual N-body			HNN 3-body			Balls		
Method	**seed**	$\|z\|^2$	$\sum_i z_i$	$z^\dagger \mathbf{J} z$	$\|z\|^2$	$\sum_i z_i$	$z^\dagger \mathbf{J} z$	$\|z\|^2$	$\sum_i z_i$	$z^\dagger \mathbf{J} z$
MLP	42	97.6	87.41	105.6	84.33	68.8	100.3	82.72	65.04	148.3
	43	97.86	87.84	99.6	85.06	66.53	74.08	85.31	69.71	148.3
	44	97.99	86.71	103.3	227.1	80.98	802.6	80.23	62.48	198.2
GRU	42	97.64	87.03	102	96.69	82.39	102.7	87.13	66.94	96.34
	43	97.76	85.93	116.6	96.19	80.77	109.6	86.44	70.01	118.2
	44	97.81	87.66	98.33	92.06	75.35	88.61	87.27	69.46	133.1
GNN	42	488.8	143	176.2	157.9	54.14	397.3	640.8	154.7	1041
	43	363.7	125.5	111.9	127.7	43.65	129.7	679	205.4	1771
	44	478.4	132.5	416.5	169.5	60.49	219.7	648.7	184.3	3869
SymODEN	42	63.35	31.25	538.2	3816	466.7	2.0e4	7.5e4	2004	6.3e4
	43	50.71	18.02	583.5	1.2e4	816.1	3828	9.0e4	2537	6.2e4
	44	56.27	27.74	713.9	6992	731.7	5.3e4	7.9e4	2209	4.8e5
HNN	42	1.18	1.631	1095	360.2	108.9	4147	4443	532.8	1.3e4
	43	1.097	0.539	67.7	217.4	64.73	342.2	5269	572.2	9188
	44	1.502	1.072	117.1	43.65	20.32	292.3	8349	688.1	7.1e4
UWM	42	3.3e−14	4.235	266.9	4.3e−14	5.487	1.5e4	2.4e−14	3.5e−8	1.9e−6
	43	3.7e−14	7.099	569.4	3.0e−14	6.061	153.1	3.3e−14	8.149	237.2
	44	3.0e−14	2.92	230.2	2.7e−14	4.948	90.83	2.4e−14	1.4e−7	4.9e−6
Ctrl-free	42	77.06	49.99	204.7	80.87	55.76	142.1	79.12	54.69	102
	43	87.32	65.46	1.0e4	86.81	64.35	214.3	103.9	45.68	915
	44	71.91	51.78	117.5	79.93	57.88	112	91.78	48.09	209.9
Ctrl-cayley	42	3.6e−14	7.022	251.9	4.2e−14	6.189	586.4	9.0e−13	4.844	1183
	43	3.8e−14	6.97	410.2	3.2e−14	5.963	631.9	5.8e−13	8.466	92.33
	44	3.7e−14	6	299.5	3.6e−14	6.046	1426	3.6e−14	6.18	226.9
NWM	42	5.1e−14	2.7e−14	6.7e−13	5.5e−14	3.3e−14	5.8e−13	1.3e−13	6.8e−14	1.7e−12
	43	5.2e−14	2.7e−14	2.6e−13	4.3e−14	2.6e−14	2.6e−13	1.1e−13	5.9e−14	9.6e−13
	44	4.3e−14	2.1e−14	2.3e−13	4.8e−14	2.8e−14	2.4e−13	6.1e−14	2.9e−14	4.0e−13

The nine dataset-invariant means of the NWM block span 2.491×10^{-14} to 1.013×10^{-12}, which §4.1 rounds to two significant figures. The two designs that reach the same resolution on the norm, UWM and Ctrl-cayley, are separated from NWM by the other two columns rather than by that one, since both leave the block sum near 6% and the symplectic pairing in the hundreds on every dataset.

One block of this table must not be read as conservation. On Balls, UWM at seeds 42 and 44 produces a rollout that does not move, with $\|z_{29} - z_0\|/\|z_0\|$ equal to 3.0×10^{-9} and 1.1×10^{-8}, so the values near 10^{-6} in those two rows record a frozen latent rather than a quantity held fixed under a non-trivial step, and the same two runs carry 10^{-6} in both halves of Table 11 for the same reason. The corresponding points are marked in Figure 3, and seed 43 of the same cell, which does move, gives the drift of 8.149 and 237.2 that the other two rows would otherwise contradict.

The symplectic column also carries a scale that the other two do not. Its reference value $|Q(z_0)|$ has a median of 1.45×10^{-2} against 0.99 for the norm and 1.14 for the block sum and falls as low as 4×10^{-6}, and the rank correlation between $|Q(z_0)|$ and the drift is -0.315 at $p = 7.0 \times 10^{-3}$ over the 72 baseline cells of that column, so part of its apparent size across the unconstrained rows is the small denominator rather than a larger absolute movement. This affects the baselines alone, since the NWM rows sit at 10^{-13} under the same denominators.

D.3 PREDICTIVE ACCURACY

Exactness is imposed on the transition rather than on the objective, so it can only be paid for in predictive accuracy, and Table 12 reports that cost under the two measures available. Relative latent error is read on each model's own latent and is therefore not comparable across rows, since every predictor carries its own encoder and the latents consequently differ in scale, whereas decoded PSNR passes through each model's decoder to a common pixel scale and is comparable. NWM ranks

Table 11: **Drift of the quadratic-form bank at $h = 29$ (%), per seed.** *in* is the mean over the 50 forms drawn from the span of Corollary 2 and *out* the mean over the 50 outside it, the latter orthogonalised against that span before the measurement, so the two halves share no component.

		visual N-body		HNN 3-body		Balls	
Method	**seed**	in	out	in	out	in	out
MLP	42	159.4	110.4	149.5	133.6	165.2	629.5
	43	121.4	115.2	209	139.4	145.9	211.8
	44	106.6	112.1	1016	2272	205.6	344.8
GRU	42	104.8	110.4	118.6	108.7	279.8	236.8
	43	107.7	108.4	110.2	112.9	174.2	152.8
	44	107.7	106.5	128.6	111.2	158.4	150.4
GNN	42	1172	2194	2562	1664	4156	4544
	43	2493	1232	1068	2630	4904	3599
	44	4256	3959	4144	1079	6109	3056
SymODEN	42	1009	955.5	2.2e4	1.7e4	4.2e5	3.6e5
	43	698.7	1053	6.6e4	7.3e4	5.3e5	7.0e5
	44	1038	2411	3.4e4	4.1e4	4.4e5	6.3e5
HNN	42	151.4	103.3	2039	2588	2.1e4	2.4e4
	43	114.3	163.6	1419	3053	2.3e4	3.0e4
	44	139.6	608.1	647.4	546.4	3.7e4	4.0e4
UWM	42	498.9	343.6	498.3	680.6	2.9e−6	2.1e−6
	43	582.4	543.2	1268	615.8	495.1	1655
	44	326.2	320.4	537.5	1101	1.7e−5	7.6e−6
Ctrl-free	42	264.5	202.7	954.2	302.8	416.3	314.5
	43	206.2	161.9	127.7	142.7	1260	613.7
	44	282.6	290.4	144.1	270.2	983.6	463
Ctrl-cayley	42	543.9	702.9	880.4	883.5	617.8	7.5e4
	43	614.2	582	536.5	706.9	454.3	567.9
	44	823.3	550.4	711.4	478.6	603.6	700
NWM	42	1.9e−4	460.1	1.2e−4	367.7	9.2e−5	232.4
	43	6.8e−5	438.4	1.1e−4	564.3	5.9e−5	384.9
	44	1.1e−4	468.2	1.1e−4	260.3	5.9e−5	541.7

second, third and fourth of nine in latent error, behind HNN on visual N-body, behind Ctrl-free and the GRU on HNN 3-body, and behind the MLP, the GRU and Ctrl-free on Balls.

Table 12: **Relative latent error and decoded PSNR at $h = 29$**, mean and standard deviation over three seeds. Latent error is read on each model's own latent and is not comparable across rows, whereas PSNR is decoded to pixels and is.

	visual N-body		HNN 3-body		Balls	
Method	latent err.	PSNR (dB)	latent err.	PSNR (dB)	latent err.	PSNR (dB)
MLP	1.032 ±0.0032	18.9578 ±0.0081	2.396 ±1.51	18.6330 ±0.0063	0.9018 ±0.0101	12.3831 ±0.0014
GRU	1.038 ±0.0069	18.9441 ±0.0038	1.183 ±0.0315	18.6271 ±0.0056	1.06 ±0.0103	12.3773 ±0.0089
GNN	10.7 ±0.973	18.3623 ±0.3317	2.352 ±0.19	18.5798 ±0.0168	8.702 ±0.572	12.4119 ±0.0078
SymODEN	2.33 ±0.212	18.7403 ±0.0233	8.531 ±3.84	18.6081 ±0.0114	808.7 ±92.3	12.3139 ±0.0078
HNN	0.7684 ±0.0286	18.8501 ±0.0154	3.198 ±0.586	18.5698 ±0.0154	66.23 ±24.4	12.1906 ±0.0814
UWM	1.462 ±0.586	18.5714 ±0.6511	1.287 ±0.0243	18.4532 ±0.0617	1.548 ±0.205	12.3801 ±0.0758
Ctrl-free	1.011 ±0.0232	18.6719 ±0.1046	0.6999 ±0.176	18.6319 ±0.0109	1.298 ±0.221	12.3786 ±0.0058
Ctrl-cayley	1.709 ±0.399	17.6512 ±0.8507	1.395 ±0.131	18.5842 ±0.0480	2.07 ±0.24	12.2603 ±0.0542
NWM	0.8568 ±0.0566	18.8562 ±0.0246	1.278 ±0.0307	18.5582 ±0.0724	1.363 ±0.0627	12.4169 ±0.0070

The PSNR column separates the nine designs on one dataset out of three. Its spread is 1.31 dB on visual N-body, from the MLP down to Ctrl-cayley, against 0.18 dB on HNN 3-body and 0.226 dB on Balls, both of which are at or below the seed deviation of several rows, so on those two datasets decoded PSNR does not order the methods at all. The PSNR ranking is moreover close to decoupled from the latent ranking, since NWM is first on Balls PSNR while fourth on Balls latent error and eighth of nine on HNN 3-body PSNR while third on latent error, which is why the main text quotes both measures rather than either alone, and on visual N-body, where the spread is real, NWM is third at 0.10 dB behind the MLP.

D.4 Perturbing the operator

Table 13 gives the family ratio at $h = 29$ on visual N-body under the two perturbations quantified over in Equation 1 that can be applied to this dataset, namely a rebuilt object partition and a wider

coupling network. The three cores hold between 3,945 and 3,963 parameters and differ only in how the $\binom{N}{2}$ scalars are assembled. Each value is the mean over seeds 42 to 44 of the per-seed ratio, the out-of-family forms are orthogonalised against the span of Corollary 2 before each ratio is formed, and the bank is redrawn for each partition, because a bank built for one partition would not span the commutant of another. The reference frame is not a perturbation here, since the visual N-body generator places its initial conditions in the centre-of-mass frame already, so the measurement below is in that frame throughout.

Table 13: **Family ratio at $h = 29$ under perturbation** (↑), tightened bank, mean and standard deviation over three seeds. No cell has a standard deviation exceeding its mean.

	Object partition			
Core	N=3	N=6	N=12	N=24
Ctrl-free	0.86 $\pm$0.15	1.00 $\pm$0.09	1.40 $\pm$0.64	0.46 $\pm$0.16
Ctrl-cayley	0.97 $\pm$0.31	1.18 $\pm$0.64	0.87 $\pm$0.48	1.76 $\pm$0.53
NWM	$\mathbf{4.3\times10^6}$	$\mathbf{9.4\times10^6}$	$\mathbf{1.0\times10^7}$	$\mathbf{9.6\times10^6}$
$\pm$	2.1×10^6	1.5×10^6	7.1×10^6	8.9×10^6
widening the coupling network at $N = 3$				
NWM-50K				4.9×10^6 $\pm1.3\times10^6$
NWM-200K				4.0×10^6 $\pm2.8\times10^6$
NWM-950K				3.8×10^6 $\pm3.0\times10^6$

Every NWM cell lies between 3.8×10^6 and 1.0×10^7, which is within a factor of three of the value at the default partition, whereas neither control leaves the interval $[0.46, 1.76]$ at any partition, so the separation is a property of the assembly and not of the particular grouping of coordinates into objects. The variation among the NWM cells should not be read as a trend in N, because the standard deviation reaches 0.93 of the mean at $N = 24$ and 0.70 at $N = 12$, so the ordering across those columns is within the seed spread. Widening the coupling network by a factor of 240 changes only how the scalars g_{ij} are computed and leaves the assembly untouched, which is why the three widened rows move within that same spread rather than with capacity, and the three structural residuals of Table 9 are verified before each of those trainings.

D.5 Relabelling the coordinates

Table 14 gives the drift of all three invariants under the labelled coordinates and under 20 random permutations applied inside every object. The norm is invariant to any relabelling by construction, so its column is constant for every predictor and carries no information, whereas the block sum and the symplectic pairing are the two that can move. The permutation is applied to the m coordinates within each object and not to the objects themselves, because Proposition 4 concerns the former and the latter is already covered by the partition column of Table 13.

D.6 The boundary of the family

Two objections would deprive the separation of Table 1 of its meaning, and this subsection addresses each in turn. The first is that the separation might rest on a single fortunate seed, which Table 15 answers by giving the ratio per seed and per dataset, where the smallest of the nine NWM entries is 2.3×10^6 and the largest competing entry across the same 81 rows is 120.6, which is five orders below it and is itself an outlier that no other row of its own method approaches. The second is that the forms called out-of-family might lie outside the span in name only, which the commutator diagnostic answers.

For the second objection we measure, for every form $A \otimes C$ of the out-of-family half, the relative commutator

$$\kappa(A) = \frac{\|[A,\, \mathbf{L}]\|_F}{\|A\|_F \, \|\mathbf{L}\|_F}, \tag{12}$$

which is read on the $N\times N$ factor rather than on the $d\times d$ generator, because $\mathbf{H} = \mathbf{L} \otimes I_m$ and $B = A\otimes C$ give $[B, \mathbf{H}] = [A, \mathbf{L}]\otimes C$, so the two versions differ by the constant factor $\|I_m\|_F = \sqrt{m}$ and the $N\times N$ one is the tighter statement. It is evaluated at the trained couplings and reported as the

Table 14: **Invariant drift at $h = 29$ (%) under 20 random relabellings.**

Method	Invariant	labelled	mean	min	max
MLP	$\|z\|^2$	86.61	86.61	86.61	86.61
	$\sum_i z_i$	61.83	63.24	58.75	68.02
	$z^\dagger \mathbf{J} z$	117.4	429.5	96.44	2.3×10^{3}
GRU	$\|z\|^2$	86.75	86.75	86.75	86.75
	$\sum_i z_i$	63.07	64.11	59.90	68.94
	$z^\dagger \mathbf{J} z$	108.2	386.1	88.71	1.9×10^{3}
GNN	$\|z\|^2$	143.2	143.2	143.2	143.2
	$\sum_i z_i$	354.1	339.6	316.4	359.8
	$z^\dagger \mathbf{J} z$	176.3	1.1×10^{3}	143.9	1.0×10^{4}
SymODEN	$\|z\|^2$	27.63	27.63	27.63	27.63
	$\sum_i z_i$	130.2	151.4	129.6	176.6
	$z^\dagger \mathbf{J} z$	538.2	1.0×10^{3}	456.7	2.8×10^{3}
HNN	$\|z\|^2$	0.590	0.590	0.590	0.590
	$\sum_i z_i$	19.94	17.34	12.85	23.30
	$z^\dagger \mathbf{J} z$	1.4×10^{3}	200.4	56.11	872.4
UWM	$\|z\|^2$	2.0×10^{-6}	2.0×10^{-6}	2.0×10^{-6}	2.0×10^{-6}
	$\sum_i z_i$	60.76	59.75	57.20	62.04
	$z^\dagger \mathbf{J} z$	266.9	401.3	176.7	1.8×10^{3}
NWM	$\|z\|^2$	2.0×10^{-6}	2.0×10^{-6}	2.0×10^{-6}	3.0×10^{-6}
	$\sum_i z_i$	1.6×10^{-5}	1.6×10^{-5}	1.6×10^{-5}	1.6×10^{-5}
	$z^\dagger \mathbf{J} z$	7.6×10^{-5}	9.3×10^{-5}	7.3×10^{-5}	5.4×10^{-4}

Table 15: **Family ratio at $h = 29$, per seed and per dataset** (↑). Each entry is the mean drift of the 50 out-of-family forms over the mean drift of the 50 in-family forms for that run, so the table is the per-seed resolution of the bottom row of Figure 3.

Method	seed	visual N-body	HNN 3-body	Balls
MLP	42	0.69	0.89	3.81
	43	0.95	0.67	1.45
	44	1.05	2.24	1.68
GRU	42	1.05	0.92	0.85
	43	1.01	1.03	0.88
	44	0.99	0.86	0.95
GNN	42	1.87	0.65	1.09
	43	0.49	2.46	0.73
	44	0.93	0.26	0.50
SymODEN	42	0.95	0.76	0.86
	43	1.51	1.10	1.32
	44	2.32	1.20	1.42
HNN	42	0.68	1.27	1.14
	43	1.43	2.15	1.31
	44	4.35	0.84	1.08
UWM	42	0.69	1.37	0.71
	43	0.93	0.49	3.34
	44	0.98	2.05	0.44
Ctrl-free	42	0.77	0.32	0.76
	43	0.79	1.12	0.49
	44	1.03	1.88	0.47
Ctrl-cayley	42	1.29	1.00	120.63
	43	0.95	1.32	1.25
	44	0.67	0.67	1.16
NWM	42	2.4×10^{6}	3.1×10^{6}	2.5×10^{6}
	43	6.5×10^{6}	5.3×10^{6}	6.5×10^{6}
	44	4.1×10^{6}	2.3×10^{6}	9.1×10^{6}

median over the 50 out-of-family forms and the three seeds. A form lying outside the span in name only would commute with the model's own Laplacian and would therefore return zero, and the test is accordingly against zero rather than against the value at another N. The median is 0.5166 at $N = 3$, with the tenth and ninetieth percentiles at 0.2134 and 0.6791, and it is 0.2300, 0.1103 and 0.0576 at $N = 6$, 12 and 24, while the fraction of forms with $\kappa < 10^{-6}$ is exactly zero at every N tested. The generic Hermitian bank used as the loose control gives 0.0745, 0.0528, 0.0373 and 0.0263 over the same four partitions, so the tightened forms are further from the commutant than the generic ones are, which is the direction the construction intends. The out-of-family half is therefore outside the commutant in substance, so the ratios of Table 15 compare two genuinely different sets of quantities.

One entry of Table 15 should be read with its origin in view. Ctrl-cayley on Balls at seed 42 reaches 120.6 because its out-of-family drift for that run is $7.5 \times 10^4\%$ against a few hundred for its other two seeds, so the entry records an unstable rollout rather than a conserved family, which is why the corresponding cell of Table 1 carries a standard deviation above its mean.

D.7 Long horizons

Rollouts are extended to 100 steps, which is ten times the training horizon, and decoded PSNR is read at six horizons from the same checkpoints as Table 10.

Table 16: **Decoded PSNR (dB) against rollout length on visual N-body**, mean over three seeds. The last two columns give the change from $h = 10$ to $h = 100$ and the rank at those two horizons.

Method	h=10	h=20	h=30	h=50	h=70	h=100	Δ	rank
MLP	22.53	21.34	21.58	21.42	21.31	21.15	-1.3825	$2 \to 1$
GRU	22.49	21.27	21.52	21.35	21.21	21.08	-1.4043	$3 \to 2$
GNN	19.73	19.75	20.62	19.34	18.48	18.02	-1.7045	$9 \to 8$
SymODEN	20.46	20.25	20.30	19.32	18.55	17.97	-2.4959	$5 \to 9$
HNN	20.45	20.30	20.45	20.10	20.17	19.75	-0.6984	$6 \to 5$
UWM	20.05	20.27	20.12	19.31	19.05	18.81	-1.2362	$8 \to 7$
Ctrl-free	22.55	21.40	21.51	21.20	21.12	20.98	-1.5726	$1 \to 3$
Ctrl-cayley	20.63	19.08	20.04	19.76	19.36	19.10	-1.5331	$4 \to 6$
NWM	20.45	20.27	20.42	20.09	20.20	19.76	-0.6829	$7 \to 4$

NWM falls by 0.6829 dB over that interval against 1.3825 dB for the MLP, which is the strongest unconstrained transition and holds first place at $h = 100$, and its rank rises from seventh to fourth without overtaking any of the three transitions above it, so the operator exchanges a lower starting point for a slower decline. HNN falls by 0.6984 dB over the same interval, which is within the seed deviation of NWM, so the two designs that carry structure degrade at the same rate and the comparison supports no claim of uniqueness. The two slowest-degrading rows of the table are therefore NWM and HNN together, and the gap to the third slowest, UWM at 1.2362 dB, is larger than the gap between them.

D.8 Transfer under a change of force law

Two shifts are applied at evaluation time to models trained on gravity, and neither involves any retraining. The first replaces the force law by a $1/r^2$ Coulomb law, and the second rescales the energy by $\kappa = 0.5$ while leaving the force law intact.

Table 17: **Decoded PSNR (dB) in distribution and under two zero-shot shifts**, at $h = 29$, mean and standard deviation over three seeds, with the rank within each column.

	in distribution		**$1/r^2$ Coulomb**		**$\kappa = 0.5$**	
Method	PSNR	rk	PSNR	rk	PSNR	rk
MLP	18.9578 $\pm$0.0081	1	18.4539 $\pm$0.2227	7	21.2439 $\pm$0.0463	3
GRU	18.9441 $\pm$0.0038	2	18.5247 $\pm$0.2238	6	21.2300 $\pm$0.0188	4
GNN	18.3623 $\pm$0.3317	8	19.1408 $\pm$1.0827	4	19.6149 $\pm$0.5267	9
SymODEN	18.7403 $\pm$0.0233	5	20.3009 $\pm$0.1035	3	20.8372 $\pm$0.3490	5
HNN	18.8501 $\pm$0.0154	4	20.4139 $\pm$0.0386	1	21.2993 $\pm$0.0307	2
UWM	18.5714 $\pm$0.6511	7	17.5397 $\pm$0.2158	8	19.7351 $\pm$0.8187	8
Ctrl-free	18.6719 $\pm$0.1046	6	18.5762 $\pm$0.1455	5	20.6743 $\pm$0.1819	6
Ctrl-cayley	17.6512 $\pm$0.8507	9	17.5042 $\pm$0.3072	9	20.3516 $\pm$1.0164	7
NWM	18.8562 $\pm$0.0246	3	20.4116 $\pm$0.0164	2	21.3298 $\pm$0.0277	1

Under the Coulomb law NWM, HNN and SymODEN reach 20.41, 20.41 and 20.30 dB whereas the MLP and the GRU fall to 18.45 and 18.52 dB, a gap of 1.9 dB that is an order larger than the seed deviation of either group, and NWM moves from third in distribution to second. The second shift is the milder of the two, since rescaling the energy leaves the force law intact, and the methods that fall furthest under the Coulomb law recover their place under it, with the MLP returning from seventh to third and the GRU from sixth to fourth. Four methods finish within 0.10 dB of one another at the top of that column, namely NWM, HNN, the MLP and the GRU, while the full spread across the nine remains 1.71 dB, so the shift orders the designs less sharply than the force law does rather than not at all.

The two structured designs reach the same accuracy under the first shift by different means, which is the distinction the main text draws rather than a ranking. HNN requires the dynamics to be Hamiltonian and learns a scalar H whose gradient defines the flow, so a change of force law changes what it has learned, whereas the family of Corollary 2 is fixed by the block structure of the generator at every value of the couplings, so the same change alters the couplings g_{ij} without altering what is conserved.

D.9 Reachability in the host latent

The exact operator does not transfer to the JEPA latent of the host, and the two explanations one would reach for first are ruled out by the same sweep. Capacity is not the cause, since the operator at the host's own 10.8M parameters plans at 3.17% against 3.83% at $16{,}657$ parameters, so enlarging it does not help. The size of the model is not the cause either, since an unconstrained transition at $16{,}704$ parameters, which is within 0.3% of the smaller constrained one, reaches 14.83%. What remains is the constraint, because a unitary step fixes $\|z_h\| = \|z_0\|$ and confines the rollout to a sphere through the initial state, while in this latent the encoded norm moves by a median of 4.9% between adjacent frames and by more than 10% in the upper quartile, so most goals of a seven-step plan lie off that sphere before planning begins.

The drift this latent carries is wider than the norm alone. Between a Push-T state and a goal twenty-five steps ahead the Gram matrix of the whole family moves by 126% against 30% for the block sum and 24% for the norm, so the penalty of Equation 10, which acts on the latter two, addresses part of a drift that is family-wide, and the part it leaves untouched is the larger one. This is the measurement that sets the ceiling on what the soft constraint can be expected to recover.

D.10 The penalty weight

The weight λ_H of Equation 10 is swept on Tworoom with every other hyperparameter held at the host's value, and $\lambda_H = 0.1$ is used in all three environments. Table 18 gives the drift of the two quantities the penalty acts on at three horizons, and the sweep is informative because the two do not respond in the same way.

Table 18: **Drift under the penalty weight on Tworoom** (%, ↓). E is $\|z\|^2$ and P the block sum. The value at $\lambda_H = 0$ is the host.

	***E* drift**			***P* drift**	
$\boldsymbol{\lambda_H}$	h=1	h=7	h=15	h=7	h=15
0.0	12.96	31.51	33.21	39.03	36.91
0.03	9.12	26.74	30.39	34.24	33.73
0.1	5.94	**21.03**	28.12	27.75	30.71
0.3	3.51	15.55	24.08	**23.78**	**30.43**
1.0	**2.50**	11.31	**19.63**	33.46	47.32

The drift of the norm falls monotonically in λ_H and by a factor of 5.2 at $h = 1$, whereas the drift of the block sum is not monotone, since it improves as far as $\lambda_H = 0.3$ and then reverses at $\lambda_H = 1.0$, ending at 47.32 against 36.91 for the host at $h = 15$. A weight large enough to dominate the prediction term therefore worsens the very quantity it constrains, which is one reason the chosen value sits at the lower end of the grid rather than at the end that minimises the norm.

Two costs accompany the penalty and both grow with the weight. One-step prediction error rises monotonically across the same grid, from 8.980×10^{-3} at $\lambda_H = 0$ through 9.745×10^{-3}, 1.007×10^{-2} and 1.082×10^{-2} to 1.139×10^{-2} at $\lambda_H = 1.0$, which is a rise of 26.8% and is the direct price of constraining a predictor that was not built to satisfy the constraint. The advantage in planning also narrows with depth, from $+8.50$ points at a planning horizon of $H = 5$ to $+3.33$ at $H = 10$ on the same environment at a fixed replanning interval, which is what a penalty applied to one step should do, since it constrains the transition between adjacent latents and not the composition of several.

D.11 NUMERICAL PRECISION

Every measurement reported on the physics datasets is read in double precision, because the quantities being compared reach 10^{-14} and a single-precision rollout cannot resolve them. The unitarity residual of the assembled step is 9.6×10^{-16} per block in float64 against 5.7×10^{-7} in float32, so a float32 evaluation would place NWM, UWM and Ctrl-cayley at the same apparent precision and would erase the distinction that Table 10 rests on. Training itself runs in single precision, and the $N \times N$ solve of Equation 5 is the one operation carried in double precision throughout, since it is the step at which a loss of precision would enter the operator rather than the objective and its cost is negligible at $N \leq 24$.

The closed-form reduction of the Cayley step, which assumes $\mathbf{H} = \mathbf{L} \otimes I_m$, reproduces the unreduced output of NWM and of its widened variants to 2.7×10^{-7} in float32, whereas applied to Ctrl-cayley, whose generator is built on a random Hermitian basis rather than a Laplacian, the two outputs differ by 4.372. The reduction is exact where the Kronecker structure is present and fails where it is absent, which is a second and independent signature of the same structure.

D.12 THE PERCEPTUAL TERM

The pixel objective of Algorithm 2 carries a perceptual term at weight 0.01, which acts on decoded frames and refers to no latent quantity, so it cannot be the source of any conservation reported here. It is applied identically to all nine transitions, so no comparison in this paper depends on it, and the argument that it does not affect conservation rests on the objective containing no latent invariant rather than on an ablation.