HAMIFORMER: DUAL-EXPERT DIFFUSION FIELDS WITH AFFINE SYMPLECTIC MAPS

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting smooth dynamics and collisions requires modeling continuous evolution and abrupt state changes. We introduce HamiFormer, a dual-expert diffusion field combining whole-window denoising with residual-corrected Hamiltonian propagation. Their mixed-state feedback attenuates the direct contribution of inherited autoregressive error: each mixed state guides subsequent propagation within the jointly refined window. Parallel Local Affine Scan (PLAS) amortizes iterative refinement across rectified-flow steps and evaluates derivatives in parallel across physical time. PLAS's affine symplectic maps achieve lower solver error and runtime than sequential explicit Euler in our evaluation. A Regime Model Tree specializes residuals and routing to balance typical-state accuracy against large tail errors. Our analysis gives conditions for physically consistent refinement and warm-start tracking, and finite-window error bounds under diffusion feedback. In 192-step evaluations, HamiFormer reduces normalized phase-space MSE by 26.3% against PhysiFormer on HamiBalls-1 and 21.4% against DiT on HamiBalls-2, with comparable model capacities. Disjoint-interval comparisons show the lowest late-horizon position and momentum errors among baselines on both datasets.
Project page: `https://hamiformer.github.io/`.

1 INTRODUCTION

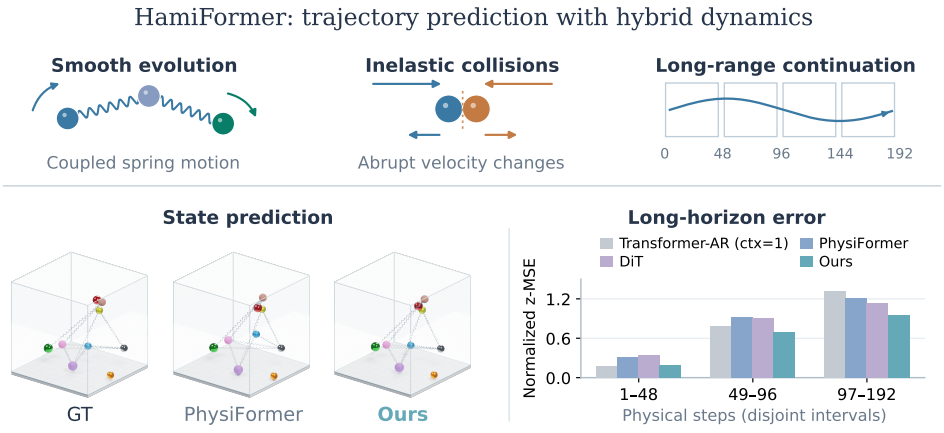


Figure 1: HamiFormer overview: schematic hybrid dynamics (top), with HamiBalls-2 state predictions at step 21 and disjoint-interval normalized phase-space MSE over 192 steps (bottom).

Learned physical simulators must reconcile local dynamics with long-range prediction. A one-step predictor can be accurate on observed states yet encounter progressively different inputs when recursively applied to its own outputs, allowing prediction errors to accumulate over time. This distribution shift makes long-horizon reliability difficult to infer from one-step prediction accuracy. Whole-window generative prediction offers a complementary approach by revising all future states together and updating distant physical times. PhysiFormer adopts this perspective for world-space mechanics (Chen et al., 2026). Combining local state propagation with whole-window denoising requires coordinating their predictions across physical time within each jointly refined future window.

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.

We parameterize local propagation through a learned scalar Hamiltonian, whose derivatives define smooth mechanical evolution (Greydanus et al., 2019). However, collisions, dissipation, and model mismatch also require a flexible prediction pathway beyond the smooth Hamiltonian model. The challenge is therefore to integrate local dynamics into a whole-window diffusion model, with its corrections fed back into subsequent propagation.

We propose **HamiFormer**, whose clean trajectory estimator contains two experts. The diffusion expert predicts the entire future window. The Hamiltonian expert combines a scalar Hamiltonian model with a learned residual, denoted $H + r$. At each physical edge, a router mixes their candidates, and this mixed state becomes the predecessor of the next Hamiltonian step. This feedback provides a *soft reset*: mixing with the whole-window diffusion candidate attenuates the direct contribution of inherited autoregressive error before subsequent Hamiltonian propagation. At each denoising step, this mixed-state propagation re-estimates the entire future trajectory.

Repeated nonlinear propagation inside a diffusion sampler introduces a second challenge. General nonseparable Hamiltonians require implicit symplectic-Euler solves throughout the physical trajectory. Building on Newton-based sequence linearization (Lim et al., 2024), we develop **Parallel Local Affine Scan (PLAS)** for closed-loop dual-expert denoising. PLAS accommodates general Hamiltonians without imposing a prescribed separable form. It linearizes implicit Hamiltonian relations at cached trajectory anchors, constructing affine maps from gradients and Hessians evaluated in parallel across physical time. These maps act on newly mixed predecessors; nonlinear residuals and routing respond to the evolving mixed history. Anchors persist across rectified-flow (RF) steps, carrying refinement progress between changing denoising conditions. Under suitable local conditions, this selective refinement tracks the evolving mixed solution, amortizing iterative work across RF evaluations. Each PLAS-SymEuler affine map preserves symplectic structure exactly in the Hamiltonian propagation step, before residual correction and diffusion mixing. In our solver evaluation, the resulting propagation achieves lower numerical error and runtime than sequential explicit Euler.

Finally, residual corrections and expert preferences vary across dynamical regimes. A shared predictor must accommodate small smooth corrections, sharp impulses, and recovery from earlier mistakes; a few large errors can dominate the objective and compete with accuracy on typical states. We introduce a **Regime Model Tree** with learned feature splits, leaf-specific linear residual readouts, and leaf-specific affine routing corrections. Motivated by potential gradient interference across regimes, conditioning corrections and routing on observable features encourages specialized responses to their different correction needs.

Our contributions are summarized as follows:

(1) We introduce a closed-loop dual-expert diffusion field, PLAS for efficient Hamiltonian propagation with affine symplectic maps, and a regime-conditioned residual/router design for regime-adaptive correction.

(2) We establish conditions for physical consistency, local refinement convergence, and warm-start tracking, and derive finite-window error bounds under diffusion feedback, with empirical support.

(3) Evaluations on HamiBalls-1 and HamiBalls-2 demonstrate lower aggregate and late-horizon prediction errors over 192 physical steps than the evaluated baselines. Component ablations examine numerical accuracy and solver efficiency, regional specialization, and training objectives.

2 Related Work

Hamiltonian learning and structured computation. Hamiltonian neural networks learn energy-based dynamics (Greydanus et al., 2019), with dissipative extensions (Sosanya & Greydanus, 2022). Neural Hamiltonian Diffusions model stochastic Hamiltonian dynamics on curved manifolds (Park, 2025). SympNets learn symplectic maps (Jin et al., 2020), complementing structure-preserving integration (Hairer et al., 2002). DEER uses Newton linearization and affine recurrences to parallelize nonlinear sequences, including HNN simulation (Lim et al., 2024); quasi-DEER and ELK address approximate Jacobians and stability (Gonzalez et al., 2024). PLAS builds on this solver lineage for Hamiltonian propagation within mixed-state denoising.

Denoising and whole-window physical prediction. Rectified flow and flow matching learn transport fields from interpolation paths (Liu et al., 2023; Lipman et al., 2023), while diffusion transform-

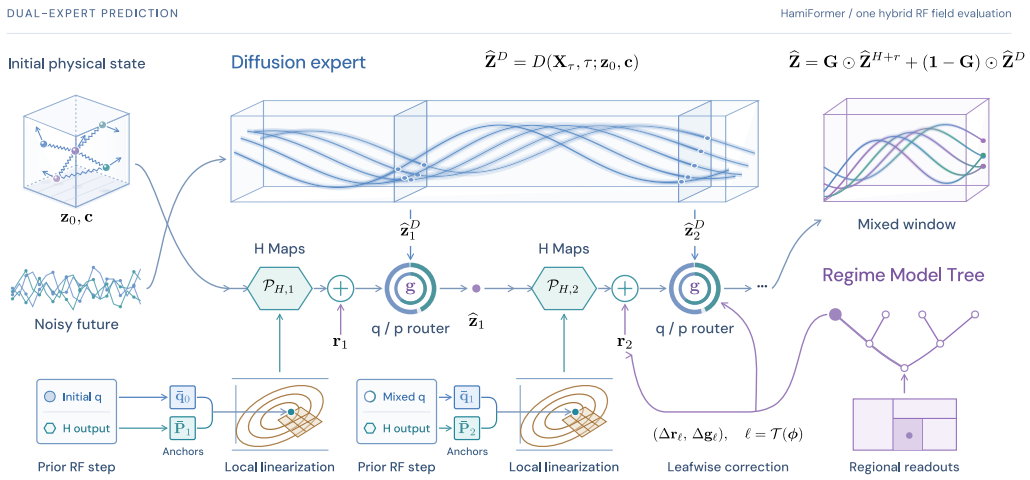


Figure 2: Overview of HamiFormer. At each RF field evaluation, a whole-window diffusion expert and a residual-corrected Hamiltonian expert produce candidates for componentwise mixing. PLAS constructs affine maps in parallel from cached anchors and scans the mixed history. The Regime Model Tree conditions the residual and router readouts.

ers enable joint denoising (Peebles & Xie, 2023). Hamiltonian Score Matching and Generative Flows introduce Hamiltonian velocity predictors for generative modeling (Holderrieth et al., 2024); port-Hamiltonian formulations interpret diffusion sampling through energy and feedback control (Darehmiraki, 2026). These approaches structure evolution in generative time. For physical prediction, PhysiFormer jointly denoises future trajectory states across physical time (Chen et al., 2026).

Combining denoising with Hamiltonian dynamics. Denoising Hamiltonian Network (DHN) denoises overlapping local Hamiltonian blocks for prediction and physical reasoning (Deng et al., 2026). Hamiltonian-Guided Diffusion Fields (HG-DPF) guides joint denoising of larger prediction windows with a separately learned Hamiltonian (Sang & Rooshenas, 2026). Both build on smooth Hamiltonian dynamics. HamiFormer couples whole-window denoising with residual-corrected propagation through routed feedback, accommodating non-Hamiltonian effects and prediction errors.

3 HamiFormer

At each denoising step, HamiFormer mixes diffusion and Hamiltonian candidates across the prediction window, feeding each mixed state into subsequent physical propagation (Figure 2).

3.1 States, time axes, and the diffusion expert

We represent physical states by positions and momenta, providing a phase-space description for Hamiltonian propagation. Given initial state $\mathbf{z}_0$ and attributes $\mathbf{c}$, we predict L states $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_L)$, where $\mathbf{z}_k = (\mathbf{q}_k, \mathbf{p}_k)$ collects the positions and momenta of n objects. Physical time is $t_k = kh$; RF time is $\tau \in [0, 1]$. We use N RF intervals and K PLAS refinements per field evaluation.

Let $\mathbf{X}$ be the trajectory in fixed normalized coordinates. The diffusion expert D reads the full noisy future block, τ, $\mathbf{z}_0$, and $\mathbf{c}$, and returns a clean estimate and object–time tokens. We train it using

$$\mathbf{X}_\tau = \tau \mathbf{X} + (1 - \tau)\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathrm{I}), \quad \tau = \sigma(\xi), \quad \xi \sim \mathcal{N}(\mu_\tau, \sigma_\tau^2), \tag{1}$$

with the velocity loss (where $d_\tau = \max(1 - \tau, \varepsilon_\tau)$)

$$\mathcal{L}_D = \mathbb{E}\left\| \frac{D(\mathbf{X}_\tau, \tau; \mathbf{z}_0, \mathbf{c}) - \mathbf{X}_\tau}{d_\tau} - \frac{\mathbf{X} - \mathbf{X}_\tau}{d_\tau} \right\|^2 = \mathbb{E}\left\| \frac{D(\mathbf{X}_\tau, \tau; \mathbf{z}_0, \mathbf{c}) - \mathbf{X}}{d_\tau} \right\|^2. \tag{2}$$

The positive denominator floor ε_τ is part of the implemented objective. The time-distribution settings and floor are specified in Appendix D.2. The backbone is a factorized diffusion transformer; architectural details are in Appendix D.1. Its clean physical-coordinate prediction is $\widehat{\mathbf{Z}}^D$.

3.2 DUAL EXPERTS AND MIXED-STATE FEEDBACK

At each RF field evaluation, diffusion supplies a whole-window candidate $\widehat{\mathbf{Z}}^D$. The Hamiltonian operator $\mathcal{P}_{H,k}$ produces a structured one-step candidate corrected by residual $\mathbf{r}_k$. Router weights $\mathbf{g}_{k,i} \in [0,1]^2$ act on its position and momentum components, broadcast across coordinates:

$$\begin{aligned} \widehat{\mathbf{z}}_k^H &= \mathcal{P}_{H,k}(\widehat{\mathbf{z}}_{k-1}), \quad \widehat{\mathbf{z}}_k^{H+r} = \widehat{\mathbf{z}}_k^H + \mathbf{r}_k, \\ \widehat{\mathbf{z}}_{k,i} &= \mathbf{g}_{k,i} \odot \widehat{\mathbf{z}}_{k,i}^{H+r} + (\mathbf{1} - \mathbf{g}_{k,i}) \odot \widehat{\mathbf{z}}_{k,i}^D. \end{aligned} \tag{3}$$

The propagation operator acts on the *current mixed predecessor*. The corrected $H + r$ candidate is formed before the router evaluates it. Crucially, the mixed output $\widehat{\mathbf{z}}_k$ immediately becomes the propagation input at the next physical edge. Thus diffusion-based corrections can alter all subsequent structured propagation. The following subsections specify the implementations of $\mathcal{P}_H$, $\mathbf{r}$, and $\mathbf{g}$.

3.3 HAMILTONIAN PROPAGATION VIA PARALLEL LOCAL AFFINE SCAN

PLAS-SymEuler instantiates $\mathcal{P}_H$ through Newton-type trajectory linearization (Lim et al., 2024). A scalar $H_\theta(\mathbf{q}, \mathbf{p}; \mathbf{c})$ defines the Hamiltonian prior. Suppressing $\theta, \mathbf{c}$, symplectic Euler gives

$$\mathbf{p} = \mathbf{P} + hH_{\mathbf{q}}(\mathbf{q}, \mathbf{P}), \qquad \mathbf{Q} = \mathbf{q} + hH_{\mathbf{p}}(\mathbf{q}, \mathbf{P}). \tag{4}$$

A mixed-coordinate anchor $(\bar{\mathbf{q}}, \bar{\mathbf{P}})$ pairs predecessor position and next-step Hamiltonian momentum. Parallel gradient and Hessian evaluation at all L fixed anchors defines full-system blocks

$$\mathbf{U} = hH_{\mathbf{qq}}(\bar{\mathbf{q}}, \bar{\mathbf{P}}), \quad \mathbf{B} = \mathrm{I} + hH_{\mathbf{qp}}(\bar{\mathbf{q}}, \bar{\mathbf{P}}), \quad \mathbf{V} = hH_{\mathbf{pp}}(\bar{\mathbf{q}}, \bar{\mathbf{P}}). \tag{5}$$

Linearizing Eq. (4) at the anchor yields an affine map $\mathcal{P}_{H,k}(\mathbf{z}) = \mathbf{A}_k\mathbf{z} + \mathbf{b}_k$. For nonsingular $\mathbf{B}$, its offset is $\mathbf{b} = \mathbf{z}_{a,+} - \mathbf{A}\mathbf{z}_a$, with linear part and anchor-compatible source–target pair

$$\mathbf{A} = \begin{bmatrix} \mathbf{B}^\top - \mathbf{V}\mathbf{B}^{-1}\mathbf{U} & \mathbf{V}\mathbf{B}^{-1} \\ -\mathbf{B}^{-1}\mathbf{U} & \mathbf{B}^{-1} \end{bmatrix}, \quad \begin{aligned} \mathbf{z}_a &= [\bar{\mathbf{q}},\ \bar{\mathbf{P}} + hH_{\mathbf{q}}(\bar{\mathbf{q}}, \bar{\mathbf{P}})]^\top \\ \mathbf{z}_{a,+} &= [\bar{\mathbf{q}} + hH_{\mathbf{p}}(\bar{\mathbf{q}}, \bar{\mathbf{P}}),\ \bar{\mathbf{P}}]^\top. \end{aligned} \tag{6}$$

Affine maps act sequentially on newly mixed predecessors; residuals and routing remain nonlinear, and $\mathbf{B}^{-1}$ is applied through linear solves.

We use one warm-started refinement ($K = 1$). Each committed rollout supplies the next anchors: mixed predecessor positions $\mathbf{q}_{k-1}$ and uncorrected H-candidate momenta $\mathbf{P}_k$. Carrying these pairs across RF intervals preserves refinement progress as the denoising conditions evolve.

3.4 REGIME-CONDITIONED RESIDUALS AND ROUTING

A shared residual $\mathbf{r}_0$ corrects $\mathbf{q}$ and $\mathbf{p}$ using diffusion tokens, noisy and initial states, mixed history, candidates, and attributes. Observable features encode candidate differences, geometry, and history. A CART-based model tree (Breiman et al., 1984; Quinlan, 1992) selects a regional linear readout:

$$\ell = \mathcal{T}(\boldsymbol{\phi}), \qquad \Delta\mathbf{r}_\ell = \mathbf{W}_\ell\boldsymbol{\psi}. \tag{7}$$

The split features and thresholds are fitted on the training split. The readout vector $\boldsymbol{\psi}$ contains standardized, clipped features and an intercept. A hierarchical ridge penalty shares information through a common linear block and leaf-specific deviations. The effective residual used in Eq. (3) is

$$\mathbf{r}_k = (\mathbf{1} - \boldsymbol{\alpha}) \odot \mathbf{r}_{0,k} + \boldsymbol{\alpha} \odot \Delta\mathbf{r}_{\ell,k}, \tag{8}$$

where $\boldsymbol{\alpha} = (\alpha_q, \alpha_p) \in [0,1]^2$ is fitted on the training split, and each coefficient is broadcast across its spatial coordinates. The same blend applies on the first edge, where the leaf readout is zero.

The router uses a physical-time GRU (Cho et al., 2014), a shared scalar logit, and nonlinear $\mathbf{q}/\mathbf{p}$ offsets. Its leaf-specific affine correction reads current causal observables $\mathbf{o}$ and an intercept:

$$\mathbf{g}_{k,i} = \sigma\big(\mathrm{logit}(\mathbf{g}_{k,i}^{\mathrm{base}}) + \boldsymbol{\alpha} \odot \mathbf{C}_\ell[\mathbf{o}_{k,i}; 1]\big)\,. \tag{9}$$

All features are available at the current causal stage. Router recurrence resets for each complete physical rollout, while mixed-coordinate anchors persist between RF intervals. Appendix D.1 specifies the features, normalization, and tree dimensions for residuals and routing.

3.5 STATEFUL RF SAMPLING AND STAGED LEARNING

Stateful sampling. Each field evaluation returns $(\widehat{\mathbf{X}} - \mathbf{X}_\tau)/d_\tau$ and updated mixed-coordinate anchors. We use $N = 20$ RF intervals with Heun updates and a final Euler step, initializing with two diffusion-only intervals. Both Heun evaluations share the incoming anchor; the left evaluation commits the next anchor (Algorithm 2).

Joint base learning. We jointly train D and H_θ for 50,000 updates under $\mathcal{L}_{\text{base}} = \mathcal{L}_D + \mathcal{L}_H$. The diffusion loss in Eq. (2) supervises whole-window denoising. The Hamiltonian loss fits local discrete Hamiltonian relations between adjacent states, with robust object–edge weights that reduce the influence of large relation errors. Both networks are then frozen during residual and router learning.

Residual and router learning. The 800 downstream updates comprise four 200-update stages. The shared residual first learns local corrections and closed-loop recovery through local replay of trajectories generated by randomly mixing the expert candidates. Two scalar-router stages then learn recurrent mixing on trajectories from the preceding policy, held fixed within each stage: the first minimizes recurrent projection regret (Appendix D.2.3); the second adds componentwise non-degradation relative to diffusion.

Regional specialization. We then fit the model tree on training trajectories and freeze its splits. The final stage freezes the scalar trunk and trains position/momentum heads and leaf-affine routing corrections using candidate preference, recurrent projection regret, and non-degradation objectives. Hierarchical ridge fitting updates leaf residuals from refreshed rollouts; blend coefficients are fitted on the training split. Appendix D.2 details all objectives, fitting procedures, and refresh schedules.

3.6 THEORETICAL PROPERTIES

We analyze PLAS propagation, single-refinement tracking, and finite-window error propagation. Symplecticity is established at the PLAS-map level, and smoothness assumptions apply to the learned Hamiltonian. Physical collision consistency is analyzed separately. Proofs are given in Appendix A. Appendix C evaluates complete deployed predictions, including tree-boundary cases.

Theorem 3.1 (Symplecticity and local accuracy)**.** *For a learned Hamiltonian $H \in C^2$, fixed anchors and attributes, and invertible $\mathbf{B}$ in Eq.* (5)*, PLAS-SymEuler satisfies $\mathbf{A}^\top \mathbf{J} \mathbf{A} = \mathbf{J}$, where $\mathbf{J}$ is canonical. For $H \in C^3$, the local symplectic-Euler map F_h through $\mathbf{z}_a$ obeys*

$$F_h(\mathbf{z}) = \mathbf{A}\mathbf{z} + \mathbf{b} + O(\|\mathbf{z} - \mathbf{z}_a\|^2). \tag{10}$$

For quadratic H, the affine map is exact wherever the implicit step is defined.

Theorem 3.2 (Self-consistency and quadratic refinement)**.** *Fix RF conditions $\mathbf{y} = (\mathbf{X}_\tau, \tau)$, diffusion outputs, and scan initialization. Let $\mathcal{R}_\mathbf{y}$ and $\mathcal{C}_\mathbf{y}$ return the next anchor and clean trajectory. For a finite rollout with $H \in C^3$ satisfying the local regularity conditions in Appendix A.2, the unique branch-consistent fixed point $\mathbf{a}^*$ satisfies, locally for some $C_L < \infty$,*

$$\|\mathcal{R}_\mathbf{y}(\mathbf{a}) - \mathbf{a}^*\| + \|\mathcal{C}_\mathbf{y}(\mathbf{a}) - \mathcal{C}_\mathbf{y}(\mathbf{a}^*)\| \le C_L \|\mathbf{a} - \mathbf{a}^*\|^2 . \tag{11}$$

This one-update bound yields quadratic local convergence and, under the tracking conditions in Appendix A.4, local $K = 1$ warm-start tracking. Fixed-point H candidates obey Eq. (4) *at mixed predecessors.*

Proposition 3.3 (Finite-window error bound)**.** *For the nonnegative physical–memory recurrence in Appendix A.5 on an L-edge window with diffusion error at most U_D, the physical error $\mathbf{e}_k$ satisfies*

$$\max_{k \le L} \|\mathbf{e}_k\|_\infty \le V_L + C_D(L) U_D. \tag{12}$$

Here $C_D(L) = \max_{k \le L} \sum_{j=1}^{k} \|\mathbf{K}_{k,j}\|_\infty$ and $V_L = \max_{k \le L} \|\mathbf{v}_k\|_\infty$. In the equality recurrence, $\mathbf{K}_{k,j}$ maps diffusion error at j to physical error at k; $\mathbf{v}_k$ is the physical response with zero diffusion forcing, retaining initial and non-diffusion errors.

Appendix E.2 connects finite refinement to RF outputs. Deployed single refinement matches the prediction accuracy of an additional-refinement reference (computed with 16 refinement updates), with 0.1903% mean positive excess loss over 192 HamiBalls-1 edges (Appendix C.4). Appendices A.3 and B establish physical-consistency bounds for smooth propagation and matched collision events.

4 EXPERIMENTS

We evaluate full-model accuracy and error accumulation across smooth motion and contacts, then examine solver performance, mixed feedback, regional specialization, and training objectives.

4.1 PROTOCOL AND COMPARISON SCOPE

We evaluate on HamiBalls-1 and HamiBalls-2. HamiBalls-1 contains five balls in a closed square, each attached by a spring to the center. Balls interact only through inelastic collisions, which also occur with the walls; gravity is absent. HamiBalls-2 contains five to ten balls in three dimensions, sparse spring connections, gravity, and inelastic ball collisions in a bounded environment. Initial positions and momenta, mass, radius, restitution, and the HamiBalls-2 spring graph and its parameters are observable. Training uses 48-edge train-split windows; validation uses 192-edge rollouts from episode initialization. Each evaluation uses 512 physical sources from four data-generation seeds and two fixed diffusion sampling-noise realizations per source.

For $c \in \{z, q, p\}$, MSE averages normalized squared errors over valid source–noise–object–edge entries and coordinates of c, excluding the initial frame and padding. Contact contains contact-involved cells; Continuous contains the rest (Appendix G).

Baselines are PhysiFormer and matched-size HG-DPF on HamiBalls-1, and PhysiFormer, DiT, DHN, and Transformer-AR with one- or four-state contexts on HamiBalls-2. Transformer-AR uses teacher forcing; DiT follows the matched RF protocol. PhysiFormer, DiT, and Transformer-AR use 50,000 updates with batch size 64; model sizes are about 1.2M/5.6M for HamiBalls-1/2 (Appendix D.1). Context-four remains worse in true-predecessor single-step evaluation and after three true transitions before free rollout (Appendix G).

Whole-window models chain four 48-edge windows from predicted endpoints. DHN and Transformer-AR roll out recursively. Ground truth is supplied only at initialization. We report full-rollout and disjoint-interval errors over edges 1–48, 49–96, and 97–192 without resets, separating initial accuracy from later prediction performance.

4.2 TRAJECTORY PREDICTION

We first test whether HamiFormer improves 192-edge prediction on both systems. Table 1 reports the lowest total MSE for Ours; position/momentum errors within Contact and Continuous subsets test whether the gains extend to contact-involved states and the remaining trajectory cells.

Table 1: Pooled 192-edge normalized MSE (↓), with ground truth supplied only at initialization. Green, yellow, and red mark the best, second-best, and third-best results per dataset and metric.

Method	Total			Continuous		Contact	
	z	q	p	q	p	q	p
HamiBalls-1							
PhysiFormer	0.34029	0.34689	0.33369	0.34710	0.32854	0.33378	0.65186
HG-DPF	1.17966	1.29854	1.06077	1.29673	1.05897	1.41008	1.17228
HamiFormer (Ours)	**0.25071**	**0.25596**	**0.24546**	**0.25613**	**0.23985**	**0.24529**	**0.59206**
HamiBalls-2							
DHN	0.99190	0.83993	1.14387	0.82841	1.12771	1.02251	1.39999
Transformer-AR$_{\text{ctx}=1}$	0.90340	0.56438	1.24241	0.55663	1.21678	0.68721	1.64851
Transformer-AR$_{\text{ctx}=4}$	1.51165	0.94036	2.08295	0.92730	2.06619	1.14717	2.34849
DiT	0.88309	0.60233	1.16386	0.59430	1.14362	0.72957	1.48460
PhysiFormer	0.91397	0.59681	1.23113	0.58827	1.20712	0.73211	1.61155
HamiFormer (Ours)	**0.69398**	**0.49281**	**0.89516**	**0.48644**	**0.87631**	**0.59378**	**1.19375**

On HamiBalls-1, Ours achieves total MSE 0.25071 versus 0.34029 for PhysiFormer and 1.17966 for HG-DPF, reducing error by 26.3% relative to PhysiFormer. On HamiBalls-2, Ours achieves 0.69398 versus DiT's 0.88309, a 21.4% reduction over the best baseline. Figure 3 complements these aggregate results with visual comparisons of object positions and spring configurations.



Figure 3: Two HamiBalls-2 trajectories: ground truth, PhysiFormer, and Ours at selected steps. Higher gloss denotes higher restitution; red circles mark large object-wise prediction errors.

4.3 Error Accumulation

We distinguish initial from later prediction accuracy using disjoint intervals without resets (Table 2) and per-edge errors (Figure 4). Ours leads across all HamiBalls-1 intervals; its HamiBalls-2 advantage emerges during continued prediction, which carries forward earlier errors.

Table 2: Disjoint-interval normalized MSE (↓) without resets; colors follow Table 1.

Method	1–48 edges		49–96 edges		97–192 edges	
	q	p	q	p	q	p
HamiBalls-1						
PhysiFormer	0.02076	0.10887	0.23468	0.29940	0.56606	0.46324
HG-DPF	0.23083	0.70319	1.29546	1.04699	1.83393	1.24646
HamiFormer (Ours)	**0.01451**	**0.07719**	**0.16078**	**0.21496**	**0.42427**	**0.34484**
HamiBalls-2						
DHN	0.18593	0.72067	0.82700	1.43810	1.17339	1.20836
Transformer-AR$_{\text{ctx}=1}$	**0.02919**	**0.32268**	**0.33786**	1.24024	0.94524	1.70335
Transformer-AR$_{\text{ctx}=4}$	0.16930	1.04295	0.87638	2.33487	1.35787	2.47699
DiT	0.07969	0.60970	0.50147	1.32340	0.91407	1.36118
PhysiFormer	0.06255	0.54874	0.46712	1.36271	0.92879	1.50653
HamiFormer (Ours)	0.04090	0.34356	0.35779	**1.02909**	**0.78628**	**1.10399**

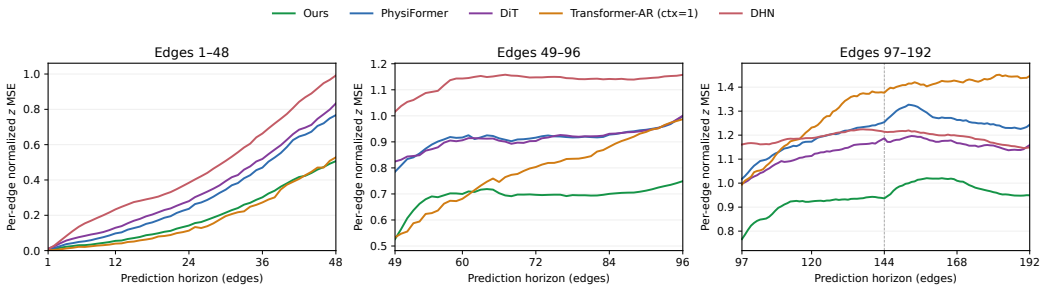


Figure 4: HamiBalls-2 phase-space MSE averaged over valid objects and realizations at each edge. Axes use separate linear ranges; the dashed line marks the diffusion window boundary at edge 144. Transformer-AR is competitive initially but exhibits stronger error growth later. Ours maintains lower error across most later edges, consistent with the interval averages over the evaluated horizon.

On HamiBalls-2, context-one Transformer-AR leads initially (MSE 0.17594 versus 0.19223 for Ours), but the ordering reverses over edges 49–96 (0.69344 for Ours versus 0.78905) and 97–192 (0.94513 versus 1.32430). In the final interval, Ours also improves on DiT's 1.13762 and PhysiFormer's 1.21766, supporting sustained prediction as a source of its full-rollout advantage.

4.4 Integrator ablation

We test PLAS's implicit-step accuracy and sampling cost on HamiBalls-1 against Explicit Euler and two-iteration symplectic Euler (SymEuler2). PLAS-S sparsely refreshes Hessians; PLAS-DC adds defect correction, with frozen-jet (PLAS-DC-F) and sparse-Hessian (PLAS-DC-S) variants.

Solver errors are measured against converged FP64 Newton references at each method's inputs, using 512 sources, two noises, eight uniformly selected edges per 48-edge window, and all 18 accepted Hamiltonian-bearing RF fields. This measures implicit-step agreement, separately from ground-truth prediction MSE. Timing uses synchronized B64×48 sampling with a shared interface. All variants retain the same RF schedule, model weights, residual corrections, and routing rules, with only the numerical solver changed.

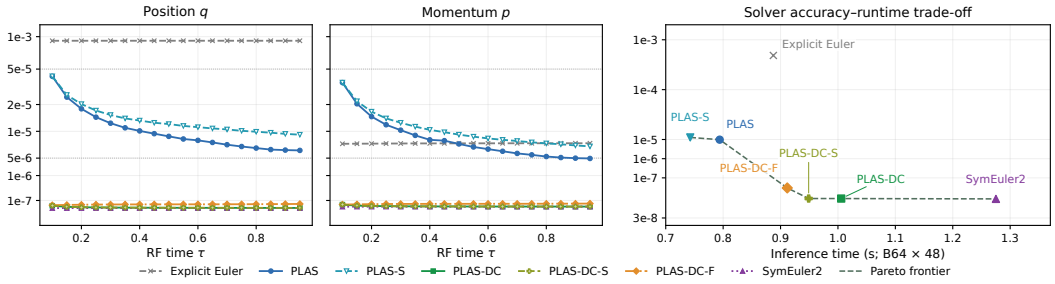


Figure 5: HamiBalls-1 integrator ablation. Left/center: cumulative-mean q/p RMSE through each RF field. Right: pooled phase-space RMSE versus synchronized B64×48 sampling time; the dashed line connects nondominated configurations. See Table 3 and Appendix E.1 for values and protocol.

Table 3: Solver RMSE against FP64 Newton on HamiBalls-1 (512 sources, two noises). Time: median of 11 warmed, synchronized B64×48 runs with a shared comparison interface (Appendix E.1).

Solver	RMSE ($\times 10^{-6}$) ↓			Time (s) ↓
	q	p	z	
Explicit Euler	685.261	7.34700	484.580	0.8875
SymEuler2	**0.04955**	**0.05661**	**0.05319**	1.2752
PLAS	10.6093	8.92038	9.80127	0.7940
PLAS-S	12.1360	9.74284	11.0047	**0.7427**
PLAS-DC	0.05031	0.05694	0.05373	1.0057
PLAS-DC-F	0.07266	0.07510	0.07389	0.9117
PLAS-DC-S	0.05042	0.05695	0.05378	0.9490

PLAS's cumulative-mean errors decrease along RF (Figure 5). It achieves phase-space RMSE 9.80×10^{-6} in 0.794 seconds, versus Euler's 4.85×10^{-4} in 0.887 seconds. Thus PLAS attains lower solver error and shorter sampling time than Explicit Euler in this comparison. PLAS-S takes 0.743 seconds with modestly increased error; PLAS-DC-S approaches SymEuler2's accuracy in 0.949 versus 1.275 seconds (Table 3). The variants provide additional accuracy–runtime choices, while Ours uses full-Hessian PLAS.

PLAS achieves near-SymEuler2 prediction MSE at lower runtime than either sequential solver (Appendix E.1). Its window-parallel derivative evaluation and lightweight affine propagation amortize refinement across RF fields while preserving the symplectic structure of its affine maps. For larger systems, PLAS-DLR reduces curvature and solve costs through diagonal–low-rank structure; Appendix F reports HamiBalls-1 accuracy and runtime/memory scaling with increasing object count.

4.5 EXPERT FEEDBACK ABLATION

We isolate the contribution of mixed-state feedback on 64 sources per dataset with two noises over the initial 48 edges. H+r candidate only outputs and propagates H+r; output-only mixing propagates H+r but outputs the blend; Ours propagates and outputs the blend.

Table 4: Expert feedback ablation: normalized MSE (↓), ranked by color within each column.

Method	HamiBalls-1			HamiBalls-2		
	q	p	z	q	p	z
H+r candidate only	0.022601	0.098554	0.060577	0.391673	1.069561	0.730617
Output-only mixing	0.020802	0.084092	0.052447	0.079374	0.426752	0.253063
Ours	**0.014472**	**0.077240**	**0.045856**	**0.043307**	**0.378203**	**0.210755**

Output mixing improves on H+r alone; mixed feedback further reduces phase-space MSE by 12.6%/16.7% on HamiBalls-1/2, improving both position and momentum (Table 4).

Against internal D-only on 512 HamiBalls-1 sources with two sampling noises, Ours reduces phase-space MSE from 0.06629 to 0.04585 over 48 edges (30.8%) and from 0.33863 to 0.25071 over 192 edges (26.0%; Appendix G.5). Both methods continue from their own predicted endpoints.

4.6 MODEL-TREE AND OBJECTIVE ABLATION

We cross shared/tree-conditioned residual-router structures with basic squared-error/full objectives on HamiBalls-1 (Table 5; Appendix D.2). Trees add regional readouts to shared r_0/router models; GL uses one global readout. All share frozen H/D models and 200 residual plus three 200-update router stages, evaluated on 512 sources with two noises over the initial 48 edges.

Table 5: HamiBalls-1, 48 edges (%, ↑). Error reductions are relative to PhysiFormer. Cell wins compare noise-mean phase-space MSE per source–edge–object against the same baseline.

ID	Structure	Objective	Error reduction (%) ↑			Cell win (%) ↑
			z	q	p	
A	Shared	Basic	−55.96	−61.89	−54.82	18.47
B	Tree	Basic	11.69	12.02	11.63	54.14
C	Shared	Full	8.42	8.83	8.34	61.60
D	Tree	Full	**29.27**	**30.19**	**29.09**	**64.41**
GL	Global leaf	Full	7.03	8.65	6.72	46.63

Tree conditioning and the full objective each improve phase-space error under either setting of the other factor. Tree+Full reduces MSE by 29.27% relative to PhysiFormer, versus 11.69% for Tree+Basic and 8.42% for Shared+Full; its 64.41% cell win rate indicates broadly distributed gains.

GL yields a 7.03% reduction under the same update budget, supporting regional readouts; the structure contrasts also change sharing and capacity (Appendix G.6).

5 CONCLUSION

HamiFormer couples whole-window diffusion and residual-corrected Hamiltonian propagation, improving full-rollout accuracy on both HamiBalls systems. Ablations support mixed feedback, regional specialization, training objectives, and PLAS's accuracy–runtime balance. The conditional analysis establishes the symplecticity of PLAS maps, local refinement and tracking results, and a finite-window error bound.

The evaluation uses simulated systems with observable physical attributes and state supervision. Future work includes broader interaction laws and observation modalities, inference of physical context from limited observations, and adaptive refinement budgets for larger systems.

AI USAGE STATEMENT

Generative AI tools, under the authors' direction and judgment, provided support for the development and refinement of hypotheses, theoretical and conceptual formulations, mathematical claims, mathematical reasoning and proof preparation, research methodology and experimental design, method implementation, synthetic-data generation, dataset reformatting, result interpretation, and translation. Additional support included literature search and summarization, manuscript editing, and preparation of code and documentation.

Across these activities, the authors guided the work, reviewed the resulting AI-assisted material, and made the final decisions regarding its incorporation. The authors take responsibility for the final text, scientific claims, results, code, and accompanying materials.

REPRODUCIBILITY STATEMENT

Appendices D.1–G describe the forward computation, sampling schedule, training settings and evaluation aggregation. Appendix A provides proofs with explicit hypotheses. Code, datasets and pretrained weights are publicly available.[1]

ETHICS STATEMENT

This study concerns synthetic mechanical trajectory prediction using simulated trajectories free of human-subject data. Approximate learned dynamics can be unreliable at rare contacts, as the contact-momentum results illustrate. Safety-critical simulation or control requires application-specific verification beyond the present experimental scope.

REFERENCES

Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. *Classification and Regression Trees*. Wadsworth International Group, 1984.

Yiming Chen, Yushi Lan, and Andrea Vedaldi. PhysiFormer: Learning to simulate mechanics in world space. *arXiv preprint arXiv:2606.27364*, 2026.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1724–1734, 2014.

Majid Darehmiraki. Port–hamiltonian diffusion models: A control-theoretic perspective on generative modeling. *arXiv preprint arXiv:2601.06071*, 2026.

Congyue Deng, Brandon Y. Feng, Cecilia Garraffo, Alan Garbarz, Robin Walters, William T. Freeman, Leonidas Guibas, and Kaiming He. Denoising hamiltonian network for physical reasoning. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=KublEgx7Hv.

Xavier Gonzalez, Andrew Warrington, Jimmy T. H. Smith, and Scott W. Linderman. Towards scalable and stable parallelization of nonlinear RNNs. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2407.19115.

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

Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, volume 31 of *Springer Series in Computational Mathematics*. Springer, 2002.

[1]Project page: https://hamiformer.github.io/; code: https://github.com/HamiFormer/HamiFormer; datasets: https://huggingface.co/datasets/HamiFormer/Hamiballs; weights: https://huggingface.co/HamiFormer/HamiFormer-Assets.

Peter Holderrieth, Yilun Xu, and Tommi Jaakkola. Hamiltonian score matching and generative flows. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Pengzhan Jin, Zhen Zhang, Aiqing Zhu, Yifa Tang, and George Em Karniadakis. SympNets: Intrinsic structure-preserving symplectic networks for identifying hamiltonian systems. *Neural Networks*, 132:166–179, 2020.

Y. H. Lim, Q. Zhu, J. Selfridge, and M. F. Kasim. Parallelizing non-linear sequential models over the sequence length. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2309.12252.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Sungwoo Park. Neural hamiltonian diffusions for modeling structured geometric dynamics. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

J. Ross Quinlan. Learning with continuous classes. In *Proceedings of the Fifth Australian Joint Conference on Artificial Intelligence*, pp. 343–348, 1992.

Guorui Sang and Pedram Rooshenas. Hamiltonian-guided diffusion fields for variable-length rigid-arm trajectory generation. In *ReALM-GEN Workshop at ICLR*, 2026.

Andrew Sosanya and Sam Greydanus. Dissipative hamiltonian neural networks: Learning dissipative and conservative dynamics separately. *arXiv preprint arXiv:2201.10085*, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

A Proofs and Scope of the Scientific Claims

Observable conditioning and model parameters are fixed throughout each physical rollout. The analysis uses exact arithmetic, with canonical symplectic matrix $\mathbf{J} = \begin{bmatrix} \mathbf{0} & \mathrm{I} \\ -\mathrm{I} & \mathbf{0} \end{bmatrix}$. Norms, derivative bounds, and implicit branches are fixed on neighborhoods containing all compared points and connecting segments. The local refinement analysis is complemented by finite-precision and boundary-tail measurements in Appendix C.

A.1 Symplecticity and local accuracy of PLAS maps

Full statement of Theorem 3.1. For $H \in C^2$, fixed anchors and attributes, and invertible $\mathbf{B}$ in Eq. (5), PLAS-SymEuler is symplectic: $\mathbf{A}^\top \mathbf{J} \mathbf{A} = \mathbf{J}$, with $\mathbf{J}$ the canonical symplectic matrix. If $H \in C^3$, the local symplectic-Euler map F_h through $\mathbf{z}_a$ satisfies

$$F_h(\mathbf{z}) = \mathbf{A}\mathbf{z} + \mathbf{b} + O(\|\mathbf{z} - \mathbf{z}_a\|^2). \tag{13}$$

For quadratic H, the affine representation is exact wherever the implicit step is defined.

Proof of Theorem 3.1. Schwarz symmetry gives $\mathbf{U}^\top = \mathbf{U}$ and $\mathbf{V}^\top = \mathbf{V}$. Factor the matrix as

$$\mathbf{A} = \underbrace{\begin{bmatrix} \mathrm{I} & \mathbf{V} \\ \mathbf{0} & \mathrm{I} \end{bmatrix}}_{\mathbf{S_V}} \underbrace{\begin{bmatrix} \mathbf{B}^\top & \mathbf{0} \\ \mathbf{0} & \mathbf{B}^{-1} \end{bmatrix}}_{\mathbf{D_B}} \underbrace{\begin{bmatrix} \mathrm{I} & \mathbf{0} \\ -\mathbf{U} & \mathrm{I} \end{bmatrix}}_{\mathbf{S_{-U}}}. \tag{14}$$

Direct block multiplication gives $\mathbf{S}_\mathbf{V}^\top \mathbf{J} \mathbf{S}_\mathbf{V} = \mathbf{J}$ and $\mathbf{S}_{-\mathbf{U}}^\top \mathbf{J} \mathbf{S}_{-\mathbf{U}} = \mathbf{J}$ because $\mathbf{U}, \mathbf{V}$ are symmetric. It also gives $\mathbf{D}_\mathbf{B}^\top \mathbf{J} \mathbf{D}_\mathbf{B} = \mathbf{J}$ because its diagonal blocks are inverse transposes of each other. The product is therefore symplectic. A constant translation has identity derivative, so the affine map retains this property.

For the approximation claim, consider the equation

$$\mathbf{\Phi}(\mathbf{q}, \mathbf{p}, \mathbf{P}) = \mathbf{P} + h H_\mathbf{q}(\mathbf{q}, \mathbf{P}) - \mathbf{p} = \mathbf{0}. \tag{15}$$

At $(\mathbf{q}, \mathbf{p}) = \mathbf{z}_a$, $\mathbf{P} = \bar{\mathbf{P}}$ is a solution and $\partial_\mathbf{P} \mathbf{\Phi} = \mathbf{B}$ is invertible. The implicit function theorem provides a local branch $\mathbf{P} = \mathbf{P}(\mathbf{q}, \mathbf{p})$; with $H \in C^3$, this branch is C^2. Define $\mathbf{Q} = \mathbf{q} + h H_\mathbf{p}(\mathbf{q}, \mathbf{P}(\mathbf{q}, \mathbf{p}))$. At the anchor-compatible source,

$$d\mathbf{P} = \mathbf{B}^{-1}(d\mathbf{p} - \mathbf{U}\, d\mathbf{q}), \qquad d\mathbf{Q} = \mathbf{B}^\top d\mathbf{q} + \mathbf{V}\, d\mathbf{P}. \tag{16}$$

Thus $DF_h(\mathbf{z}_a) = \mathbf{A}$ and $F_h(\mathbf{z}_a) = \mathbf{z}_{a,+}$. Taylor's theorem on a sufficiently small convex neighborhood yields

$$F_h(\mathbf{z}) - \mathbf{A}\mathbf{z} - \mathbf{b} = \int_0^1 (1-t) D^2 F_h(\mathbf{z}_a + t(\mathbf{z} - \mathbf{z}_a))[\mathbf{z} - \mathbf{z}_a, \mathbf{z} - \mathbf{z}_a] dt, \tag{17}$$

and hence a remainder bounded by $\frac{1}{2} \sup \left\| D^2 F_h \right\| \|\mathbf{z} - \mathbf{z}_a\|^2$. If H is quadratic, Eq. (4) is affine in $(\mathbf{q}, \mathbf{p}, \mathbf{Q}, \mathbf{P})$, its Hessian blocks are constant, and the solved map equals $\mathbf{A}\mathbf{z} + \mathbf{b}$ exactly.

Finally, for $\mathbf{z} = \mathbf{S}\mathbf{x} + \boldsymbol{\mu}$ with constant nonsingular $\mathbf{S}$, the transformed linear part is $\mathbf{A}_\mathbf{x} = \mathbf{S}^{-1}\mathbf{A}\mathbf{S}$. Consequently,

$$\mathbf{A}_\mathbf{x}^\top \mathbf{J}_\mathbf{x} \mathbf{A}_\mathbf{x} = \mathbf{S}^\top \mathbf{A}^\top \mathbf{J} \mathbf{A} \mathbf{S} = \mathbf{S}^\top \mathbf{J} \mathbf{S} = \mathbf{J}_\mathbf{x}, \qquad \mathbf{J}_\mathbf{x} = \mathbf{S}^\top \mathbf{J} \mathbf{S}. \tag{18}$$

The transformed translation again leaves the identity unchanged. □

Relation to physical-step accuracy. The theorem's remainder is second order in *anchor mismatch*. More specifically, if second and third derivatives of H and the implicit inverses are uniformly bounded for $0 < h \le h_0$ on a common neighborhood, differentiation of the implicit relation twice gives $\left\| D^2 F_h \right\| \le Ch$. The local affine error is then $O(h \|\mathbf{z} - \mathbf{z}_a\|^2)$. For a smooth Hamiltonian vector field, symplectic Euler has one-step error $O(h^2)$ relative to the continuous flow (Hairer et al., 2002); stability bounds propagate these local errors along the trajectory. For quadratic H, the affine representation exactly reproduces the discrete symplectic-Euler map.

Role of the frozen anchor. The symplectic identity applies to the derivative of each affine propagation map with its anchor held fixed. For a state-dependent anchor, differentiating $\mathbf{A}(\mathbf{a}(\mathbf{z}))\mathbf{z}+\mathbf{b}(\mathbf{a}(\mathbf{z}))$ includes the anchor's derivative terms. The complete predictor also includes residual and routing derivatives. Equation (14) characterizes the fixed-anchor Hamiltonian propagation component within this predictor.

Solves and finite precision. Invertibility of $\mathbf{B}$ defines the affine solve, and $\left\|\mathbf{B}^{-1}\right\|$ controls its sensitivity to derivative errors and the local approximation constants. The algebraic proof establishes the exact symplectic identity; the finite-precision residual $\left\|\mathbf{A}^{\top}\mathbf{J}\mathbf{A}-\mathbf{J}\right\|$ measures its numerical accuracy.

A.2 Mixed fixed points and local quadratic refinement

Fix RF conditions $\mathbf{y}=(\mathbf{X}_\tau,\tau)$ and diffusion outputs. Let $\mathcal{R}_{\mathbf{y}}(\mathbf{a})$ and $\mathcal{C}_{\mathbf{y}}(\mathbf{a})$ return the next anchor and clean trajectory, resetting the initial state and memory each scan. A rollout is *regular* if its implicit H branches are unique and nonsingular, and its residual/router/memory maps are locally Lipschitz in candidates and causal history, depending on anchors only through H candidates.

Full statement of Theorem 3.2. For a regular finite-horizon rollout with $H\in C^3$, $\mathcal{R}_{\mathbf{y}}$ has a unique fixed point $\mathbf{a}^*$ on the prescribed branches. Its H candidates satisfy Eq. (4) at the mixed predecessors. Locally, for some $C_L<\infty$,

$$\|\mathcal{R}_{\mathbf{y}}(\mathbf{a})-\mathbf{a}^*\|+\|\mathcal{C}_{\mathbf{y}}(\mathbf{a})-\mathcal{C}_{\mathbf{y}}(\mathbf{a}^*)\|\le C_L\,\|\mathbf{a}-\mathbf{a}^*\|^2\,, \tag{19}$$

so sufficiently close anchors converge quadratically under repeated refinement.

Anchor, readout, and regularity. Fix $\mathbf{y}=(\mathbf{X}_\tau,\tau)$, diffusion predictions and tokens, observable attributes, and model parameters. Write the uncorrected Hamiltonian candidate as $(\mathbf{Q}_k,\mathbf{P}_k)$ and the mixed output as $(\mathbf{q}_k,\mathbf{p}_k)$. The scan returns

$$\mathcal{R}_{\mathbf{y}}(\mathbf{a})=\big((\mathbf{q}_{k-1})_{k=2}^{L},(\mathbf{P}_k)_{k=1}^{L}\big),\qquad \mathcal{C}_{\mathbf{y}}(\mathbf{a})=\big((\mathbf{q}_k,\mathbf{p}_k)\big)_{k=1}^{L}. \tag{20}$$

The initial $\mathbf{q}_0$ is fixed. Each scan initializes physical-time memory identically and follows Eq. (3). An exact mixed rollout evaluates the prescribed implicit H branches sequentially, applying the same deterministic residual, router, and memory updates after each H candidate.

For $H\in C^3$ and nonsingular implicit blocks, the implicit function theorem supplies smooth H maps. Smaller closed neighborhoods and the finite horizon give common derivative and inverse bounds. Residual/router/memory maps, including features and selectors, must be locally Lipschitz: neighborhoods stay in one continuity region, or adjacent formulas share a Lipschitz bound.

In Algorithm 1, anchors enter only the H construction; features and GRU updates read candidates, mixed predecessors, and causal history, with memory reset each scan. Recurrent dependence is therefore mediated by the H candidates and included in the analysis. With fixed attributes and masks, the Hamiltonian networks' SiLU, softmax, and positive-offset normalization are smooth, including at physical contacts.

Proof of Theorem 3.2. At edge k, put $\bar{\mathbf{v}}_k=(\bar{\mathbf{q}}_{k-1},\bar{\mathbf{P}}_k)$ and $\mathbf{v}_k=(\mathbf{q}_{k-1},\mathbf{P}_k)$. The affine construction solves

$$\mathbf{p}_{k-1}=\mathbf{P}_k+h\big[H_{\mathbf{q}}(\bar{\mathbf{v}}_k)+H_{\mathbf{qq}}(\bar{\mathbf{v}}_k)(\mathbf{q}_{k-1}-\bar{\mathbf{q}}_{k-1})+H_{\mathbf{qp}}(\bar{\mathbf{v}}_k)(\mathbf{P}_k-\bar{\mathbf{P}}_k)\big], \tag{21}$$

$$\mathbf{Q}_k=\mathbf{q}_{k-1}+h\big[H_{\mathbf{p}}(\bar{\mathbf{v}}_k)+H_{\mathbf{pq}}(\bar{\mathbf{v}}_k)(\mathbf{q}_{k-1}-\bar{\mathbf{q}}_{k-1})+H_{\mathbf{pp}}(\bar{\mathbf{v}}_k)(\mathbf{P}_k-\bar{\mathbf{P}}_k)\big]. \tag{22}$$

For $\mathbf{f}=(H_{\mathbf{q}},H_{\mathbf{p}})$, suppose $D\mathbf{f}$ is M_H-Lipschitz on the segment from $\bar{\mathbf{v}}_k$ to $\mathbf{v}_k$. Taylor's integral remainder bounds the implicit-equation defect

$$\mathbf{d}_k=\begin{bmatrix}\mathbf{P}_k+hH_{\mathbf{q}}(\mathbf{v}_k)-\mathbf{p}_{k-1}\\ \mathbf{Q}_k-\mathbf{q}_{k-1}-hH_{\mathbf{p}}(\mathbf{v}_k)\end{bmatrix},\qquad \|\mathbf{d}_k\|\le\frac{hM_H}{2}\,\|\mathbf{v}_k-\bar{\mathbf{v}}_k\|^2\,. \tag{23}$$

At a fixed point of Eq. (20), the anchor mismatch vanishes at every edge, including the fixed initial position. Thus each H candidate satisfies the exact learned-H implicit equations at the actual mixed predecessor.

Conversely, run the prescribed unique implicit branches sequentially with the same residual, router, memory reset, and initial state. Its H candidate and mixed predecessor are the target and source of the corresponding affine jet. Supplying the resulting anchor to the scan therefore reproduces the H candidate at the first edge, then the residual, gate, mixed state, and memory. Induction gives the entire rollout and a fixed point. The same induction shows uniqueness among fixed points in these branches.

For local convergence, let F_k be the smooth exact implicit H map near the fixed rollout, and let $\mathbf{s}_k(\mathbf{a})$ be the anchor-compatible source. Theorem 3.1 gives

$$\mathbf{A}_k(\mathbf{a})\mathbf{z} + \mathbf{b}_k(\mathbf{a}) = F_k(\mathbf{s}_k(\mathbf{a})) + DF_k(\mathbf{s}_k(\mathbf{a}))(\mathbf{z} - \mathbf{s}_k(\mathbf{a})), \tag{24}$$

whose error relative to $F_k(\mathbf{z})$ is bounded by a constant times $\|\mathbf{z} - \mathbf{s}_k(\mathbf{a})\|^2$. Set $e = \|\mathbf{a} - \mathbf{a}^*\|$. Smoothness of the compatible source gives $\left\|\mathbf{s}_k(\mathbf{a}) - \mathbf{z}^*_{k-1}\right\| \le c_k e$ locally. The first predecessor and initialized memory are exact, so its H-candidate error is $O(e^2)$. Local Lipschitzness transfers this error to the residual, gate, mixed output, and updated memory.

Inductively suppose the predecessor and memory errors entering edge k are $O(e^2)$. Then $\|\mathbf{z}_{k-1} - \mathbf{s}_k(\mathbf{a})\| = O(e)$, the affine remainder is $O(e^2)$, and the exact implicit map transfers the predecessor error by a bounded factor. The H-candidate, mixed-state, and memory errors at edge k are consequently $O(e^2)$. Taking a maximum over the finite horizon, and using norm equivalence in its finite-dimensional state spaces, proves Eq. (19). Choosing a common neighborhood for all updates and shrinking its radius until $C_L r < 1$ makes it invariant and proves quadratic convergence. □

Why mixing preserves the local argument. Local Lipschitz downstream maps transfer quadratic H-candidate perturbations without a first-order anchor term. Where differentiable, Eq. (24) gives

$$D_{\mathbf{a}}(\mathbf{A}_k(\mathbf{a})\mathbf{z} + \mathbf{b}_k(\mathbf{a}))[\mathbf{u}] = D_{\mathbf{a}}\mathbf{A}_k(\mathbf{a})[\mathbf{u}]\,(\mathbf{z} - \mathbf{s}_k(\mathbf{a})). \tag{25}$$

Indeed $F_k(\mathbf{s}_k(\mathbf{a}))$ differentiates to $DF_k(\mathbf{s}_k(\mathbf{a}))D_{\mathbf{a}}\mathbf{s}_k(\mathbf{a})[\mathbf{u}]$, which cancels the derivative of the subtracted source term. The physical propagation matrix can have expanding directions while direct anchor forcing vanishes at consistency. All residual, router, and memory paths remain in the subsequent propagation.

Regularity and finite precision. Hard-split jumps are treated in Appendix E.2. A finite soft mixture of locally Lipschitz leaves, weights, and features satisfies selector regularity, but still requires regular implicit branches and local initialization. Deployment evaluation error ϵ_{fp} changes the estimate to $e^+ \le C_L e^2 + \epsilon_{\mathrm{fp}}$; it permits an invariant error band when $C_L r^2 + \epsilon_{\mathrm{fp}} \le r$, with ϵ_{fp} bounding evaluation error throughout that neighborhood. Appendix C measures point derivatives and finite-scale perturbation responses.

A.3 Physical consistency of the converged mixed trajectory

Corollary A.1 (Consistency with smooth physical dynamics)**.** *Compare a consistent mixed rollout $\mathbf{z}^*_k$ with samples $\mathbf{z}^\dagger_k$ of a smooth Hamiltonian system $H^\dagger$, sharing the initial state and fixed duration $Lh = T$. On a common convex neighborhood of trajectories and anchors, assume h-uniform bounds on the C^3 norms of $H, H^\dagger$ and implicit inverses, and ℓ_∞ vector-field error at most ϵ_H. Set $\mathbf{G}_k = \mathrm{diag}(\mathbf{g}_k)$ for coordinate-expanded gates $\mathbf{0} \preceq \mathbf{g}_k \preceq \mathbf{1}$. If the correction $\mathbf{c}_k = \mathbf{G}_k\mathbf{r}_k + (\mathrm{I} - \mathbf{G}_k)(\widehat{\mathbf{z}}^D_k - \mathbf{z}^\dagger_k)$ satisfies $\|\mathbf{c}_k\|_\infty \le h\eta_h$ at every edge, then, with C_T independent of h,*

$$\max_{k \le L} \left\|\mathbf{z}^*_k - \mathbf{z}^\dagger_k\right\|_\infty \le C_T(h + \epsilon_H + \eta_h). \tag{26}$$

Uniform physical-step assumptions. For $0 < h \le h_0$ and $Lh = T$, the common convex neighborhood contains the true and mixed trajectories and the mixed-coordinate anchors used by the H equations. Derivatives of $H, H^\dagger$ through order three and inverses of the implicit blocks are bounded independently of h. The vector-field error is $\sup_{\mathbf{z}} \left\|\mathbf{J}\nabla H(\mathbf{z}) - \mathbf{J}\nabla H^\dagger(\mathbf{z})\right\|_\infty \le \epsilon_H$ on this neighborhood. The realized residual/diffusion correction satisfies

$$\left\|\mathbf{G}_k\mathbf{r}_k + (\mathrm{I} - \mathbf{G}_k)(\widehat{\mathbf{z}}^D_k - \mathbf{z}^\dagger_k)\right\|_\infty \le h\eta_h \quad \text{for every } k. \tag{27}$$

The physical-consistency bound vanishes as h, ϵ_H, η_h tend to zero. Collision impulses need not vanish with h: Appendix B instead bounds their approximation defects.

Proof of Corollary A.1. Let $\Phi_h^\dagger$ denote the exact physical flow. Uniform smoothness and regular implicit branches give the familiar estimates

$$\|F_h(\mathbf{z}) - F_h(\mathbf{z}')\|_\infty \le (1+ch)\,\|\mathbf{z}-\mathbf{z}'\|_\infty\,, \qquad \left\|F_h(\mathbf{z}) - \Phi_h^\dagger(\mathbf{z})\right\|_\infty \le c(h^2 + h\epsilon_H) \tag{28}$$

on the common neighborhood, for sufficiently small h. The second estimate combines symplectic Euler's local truncation error with the vector-field approximation error; bounded derivatives and implicit inverses make c uniform.

At the fixed point, Theorem 3.2 identifies the H candidate with $F_h(\mathbf{z}^*_{k-1})$. Let $\mathbf{c}_k = \mathbf{G}_k\mathbf{r}_k + (\mathrm{I} - \mathbf{G}_k)(\widehat{\mathbf{z}}_k^D - \mathbf{z}_k^\dagger)$ and $E_k = \left\|\mathbf{z}_k^* - \mathbf{z}_k^\dagger\right\|_\infty$. Subtracting the true state from the actual mixture gives

$$\mathbf{z}_k^* - \mathbf{z}_k^\dagger = \mathbf{G}_k\big(F_h(\mathbf{z}^*_{k-1}) - \mathbf{z}_k^\dagger\big) + \mathbf{c}_k, \qquad E_k \le (1+ch)E_{k-1} + c(h^2+h\epsilon_H) + h\eta_h. \tag{29}$$

Here $\|\mathbf{G}_k\|_\infty \le 1$, regardless of how the gate depends on the candidates or history. With $E_0 = 0$ and $Lh = T$, discrete Gronwall yields, uniformly over all prefixes,

$$\left\|\mathbf{Z}_h^* - \mathbf{Z}_h^\dagger\right\|_\infty \le Te^{cT}\big(ch + c\epsilon_H + \eta_h\big) \le C_T(h + \epsilon_H + \eta_h). \tag{30}$$

For a fixed affine normalization $\mathcal{N}$ with linear-part norm bounded by $C_\mathcal{N}$, set $\mathbf{v}_h^* = (\mathcal{N}(\mathbf{Z}_h^*) - \mathbf{X}_\tau)/d_\tau$ and $\mathbf{v}_h^\dagger = (\mathcal{N}(\mathbf{Z}_h^\dagger) - \mathbf{X}_\tau)/d_\tau$. Since $d_\tau \ge \varepsilon_\tau > 0$,

$$\left\|\mathbf{v}_h^* - \mathbf{v}_h^\dagger\right\|_\infty \le \frac{C_\mathcal{N} C_T}{\varepsilon_\tau}(h + \epsilon_H + \eta_h). \tag{31}$$

This proves the claimed regularized-field consistency. □

Interpretation of the assumptions. The correction condition is imposed on the realized fixed-point rollout and accounts for both experts' contributions. Zero residual and exact D give $\eta_h = 0$ for every gate; $\mathbf{G}_k = \mathrm{I}$ and zero residual also give $\eta_h = 0$ independently of D. Appendix C.1 evaluates zero and vanishing model/correction budgets in analytic controls, complemented by finite-step implicit-residual and prediction measurements on the learned model.

For a finite refinement with equation defects $\mathbf{d}_k$, stable implicit inversion adds $c\,\|\mathbf{d}_k\|$ to Eq. (29). Thus, for $Lh = T$,

$$\max_{k\le L} E_k \le e^{cT}\left[E_0 + cT(h+\epsilon_H) + \sum_{k=1}^{L}\big(c\,\|\mathbf{d}_k\| + \|\mathbf{c}_k\|_\infty\big)\right]. \tag{32}$$

This separates solver error, learned-H error, and correction error. Appendix B incorporates collision-update defects and event amplification into an event-aligned extension.

A.4 Single-refinement and warm-start tracking

Proposition A.2 (Warm-start tracking)**.** *Let $\mathbf{a}^*(\mathbf{y})$ be a continuous fixed-point branch. On common-radius neighborhoods, assume $\mathcal{R}_\mathbf{y}$ contracts distance to $\mathbf{a}^*(\mathbf{y})$ by a uniform factor $\rho \in (0,1)$. Along committed RF conditions $\mathbf{y}_s$, define $\mathbf{a}_s = \mathcal{R}_{\mathbf{y}_s}^{K_s}(\mathbf{a}_{s-1})$, $K_s \ge 1$, $e_s = \|\mathbf{a}_s - \mathbf{a}^*(\mathbf{y}_s)\|$, and $d_s = \|\mathbf{a}^*(\mathbf{y}_s) - \mathbf{a}^*(\mathbf{y}_{s-1})\|$. For sufficiently small e_0 and $\sup_s d_s$, refinements stay in these neighborhoods and*

$$e_s \le \rho^{K_s}(e_{s-1} + d_s). \tag{33}$$

If $\mathbf{y}_s \to \mathbf{y}_*$ within the branch domain, then $\mathbf{a}_s \to \mathbf{a}^*(\mathbf{y}_*)$.

Proof of Proposition A.2. The local radial contraction hypothesis is

$$\|\mathcal{R}_\mathbf{y}(\mathbf{a}) - \mathbf{a}^*(\mathbf{y})\| \le \rho\,\|\mathbf{a} - \mathbf{a}^*(\mathbf{y})\|\,, \qquad \|\mathbf{a} - \mathbf{a}^*(\mathbf{y})\| \le r, \tag{34}$$

with common $r > 0$ and $0 < \rho < 1$. Choose $\delta = r\min\{\rho, 1-\rho\}$. If $e_0 \le \delta$ and $\sup_s d_s \le \delta$, then $e_{s-1} \le \rho r$ implies $e_{s-1} + d_s \le r$. Each refinement remains in the ball and contracts distance by ρ. Applying it K_s times proves Eq. (33) and $e_s \le \rho r$, hence invariance by induction.

Since $K_s \ge 1$, for $s > s_0$ the recurrence gives

$$e_s \le \rho^{s-s_0}e_{s_0} + \frac{\rho}{1-\rho}\sup_{j>s_0} d_j. \tag{35}$$

If $\mathbf{y}_s \to \mathbf{y}_*$ within the branch domain, continuity gives $\mathbf{a}^*(\mathbf{y}_s) \to \mathbf{a}^*(\mathbf{y}_*)$ and hence $d_s \to 0$. Letting $s \to \infty$ and then $s_0 \to \infty$ proves $e_s \to 0$, and the triangle inequality gives $\mathbf{a}_s \to \mathbf{a}^*(\mathbf{y}_*)$. □

Deployed single refinement. Let $\mathbf{a}_s^* = \mathbf{a}^*(\mathbf{y}_s)$ and $\widehat{\mathbf{Z}}_s = \mathcal{C}_{\mathbf{y}_s}(\mathbf{a}_{s-1})$. For $K_s = 1$, Theorem 3.2 and the triangle inequality give

$$e_s + \left\| \widehat{\mathbf{Z}}_s - \mathcal{C}_{\mathbf{y}_s}(\mathbf{a}_s^*) \right\| \le C_s(e_{s-1} + d_s)^2, \qquad e_{s-1} + d_s \le r.$$

Here r lies within the regular neighborhoods and C_s is their quadratic bound. If $\sup_s C_s r \le \rho < 1$, this implies Eq. (33) with $K_s = 1$. A bound ϵ_s on the sum of anchor and readout evaluation errors adds ϵ_s to the right-hand side. If the branch is locally L_a-Lipschitz, $d_s \le L_a \|\mathbf{y}_s - \mathbf{y}_{s-1}\|$. Thus one update per RF condition tracks the moving target across successive conditions. In Heun sampling this recurrence follows committed left anchors; each right readout uses the same incoming anchor at its own RF condition (Appendix D.1.4). Appendix E.2 propagates field errors through the finite schedule and accounts for discrete transitions.

A.5 Componentwise diffusion resets and observable memory

Full statement of Proposition 3.3. Fix an L-edge window. Let $\mathbf{E}_k$ collect physical and memory error bounds, with physical part $\mathbf{e}_k$. Assume $\mathbf{E}_k \preceq \mathbf{M}_k \mathbf{E}_{k-1} + \mathbf{B}_k \boldsymbol{\delta}_k + \mathbf{b}_k$, all terms nonnegative, where $\boldsymbol{\delta}_k$ bounds diffusion error. In the equality recurrence, let $\mathbf{K}_{k,j}$ map $\boldsymbol{\delta}_j$ to $\mathbf{e}_k$, and $\mathbf{v}_k$ be the physical response with all $\boldsymbol{\delta}_j = \mathbf{0}$. Set $C_D(L) = \max_{k \le L} \sum_{j=1}^{k} \|\mathbf{K}_{k,j}\|_\infty$ and $V_L = \max_{k \le L} \|\mathbf{v}_k\|_\infty$. If $\max_{j \le L} \|\boldsymbol{\delta}_j\|_\infty \le U_D$, then

$$\max_{k \le L} \|\mathbf{e}_k\|_\infty \le V_L + C_D(L) U_D. \tag{36}$$

Physical-output response matrices. Fix the prediction window $1 \le k \le L$ and its conditioning variables. Let $\mathbf{P} = [\mathrm{I}\ \mathbf{0}]$ select the physical part of the augmented error vector. For the equality comparison system $\mathbf{x}_k = \mathbf{M}_k \mathbf{x}_{k-1} + \mathbf{B}_k \boldsymbol{\delta}_k + \mathbf{b}_k$, with $\mathbf{x}_0 = \mathbf{E}_0$, define $\boldsymbol{\Pi}_{b:a} = \mathbf{M}_b \cdots \mathbf{M}_a$, with empty product I. Its diffusion-response matrices and remaining physical response are

$$\mathbf{K}_{k,j} = \mathbf{P} \boldsymbol{\Pi}_{k:j+1} \mathbf{B}_j, \qquad \mathbf{v}_k = \mathbf{P} \boldsymbol{\Pi}_{k:1} \mathbf{E}_0 + \sum_{j=1}^{k} \mathbf{P} \boldsymbol{\Pi}_{k:j+1} \mathbf{b}_j. \tag{37}$$

Here $\mathbf{v}_k$ retains incoming-state errors and all non-diffusion forcing, including residual/model errors, memory errors, and boundary jumps. For finite comparison matrices and forcing, the window-specific quantities

$$C_D(L) = \max_{1 \le k \le L} \sum_{j=1}^{k} \|\mathbf{K}_{k,j}\|_\infty, \qquad V_L = \max_{1 \le k \le L} \|\mathbf{v}_k\|_\infty \tag{38}$$

separate the propagated diffusion-error budget from these remaining contributions. Their dependence on L, realized gates, and the coupled state-memory dynamics is retained.

Proof of Proposition 3.3. Nonnegativity permits repeated substitution in the assumed error recurrence, giving

$$\mathbf{e}_k = \mathbf{P} \mathbf{E}_k \preceq \mathbf{v}_k + \sum_{j=1}^{k} \mathbf{K}_{k,j} \boldsymbol{\delta}_j, \qquad 1 \le k \le L. \tag{39}$$

Apply the induced sup norm, use $\|\boldsymbol{\delta}_j\|_\infty \le U_D$ over this window, and maximize over its prefixes. Equation (38) gives Eq. (36). □

Complete state and realized gates. Let $\mathbf{e}_k = |\widehat{\mathbf{z}}_k - \mathbf{z}_k^\dagger|$ and let $\mathbf{m}_k$ collect errors in all propagated physical-time memory, including accumulators. For nonnegative candidate-error bounds

$$|\widehat{\mathbf{z}}_k^{H+r} - \mathbf{z}_k^\dagger| \preceq \mathbf{F}_k^{zz} \mathbf{e}_{k-1} + \mathbf{F}_k^{zm} \mathbf{m}_{k-1} + \mathbf{a}_k, \tag{40}$$

$$|\widehat{\mathbf{z}}_k^{D} - \mathbf{z}_k^\dagger| \preceq \boldsymbol{\delta}_k, \qquad \mathbf{m}_k \preceq \mathbf{F}_k^{mz} \mathbf{e}_{k-1} + \mathbf{F}_k^{mm} \mathbf{m}_{k-1} + \mathbf{F}_k^{mD} \boldsymbol{\delta}_k + \mathbf{d}_k, \tag{41}$$

the complete mixed-state inequality has

$$\mathbf{M}_k = \begin{bmatrix} \mathbf{G}_k \mathbf{F}_k^{zz} & \mathbf{G}_k \mathbf{F}_k^{zm} \\ \mathbf{F}_k^{mz} & \mathbf{F}_k^{mm} \end{bmatrix}, \qquad \mathbf{B}_k = \begin{bmatrix} \mathrm{I} - \mathbf{G}_k \\ \mathbf{F}_k^{mD} \end{bmatrix}, \qquad \mathbf{b}_k = \begin{bmatrix} \mathbf{G}_k \mathbf{a}_k \\ \mathbf{d}_k \end{bmatrix}. \tag{42}$$

The memory bounds include dependence on the current mixed state and gate; $\mathbf{F}_k^{mD}$ retains diffusion-to-memory transfer after substituting the current-state bound. Candidate modeling errors, anchor errors, and boundary jumps enter the additive terms or their full region bounds. This coordinatewise error decomposition evaluates the gate at its realized value. The off-diagonal blocks preserve cross-object and memory-to-state error transfer.

Feedback and finite response budgets. The factors $\mathbf{G}_k$ attenuate inherited physical-candidate contributions, while $\mathrm{I} - \mathbf{G}_k$ admits the diffusion candidate's error. Their subsequent effects are transported by the full products in Eq. (37). This bounds the finite-window prediction error by the diffusion expert's error budget scaled by the window response, together with the propagated initial, model, and memory errors. The projection $\mathbf{P}$ measures the resulting physical response, including memory-to-state transfer. Finite-scale response measurements and complete-sampling memory interventions are reported in Appendix C.5.

Scope across prediction windows. Within a fixed-condition field, D jointly predicts the whole window. The budget U_D is conditioned on that window's incoming state and RF conditions. Chained prediction applies the same finite-window statement with the corresponding incoming-state, diffusion, and memory budgets at each window.

B Consistency Across Regular Collisions

The smooth estimate in Corollary A.1 extends by composing smooth propagation with collision updates. Errors are measured after aligning corresponding event times and matching pre- and post-event states. The comparison state includes recurrent memory and conditioning variables whenever they affect subsequent updates.

Proposition B.1 (Finite-event consistency)**.** *Fix a duration T and trajectories with the same ordered sequence of M isolated events. On common neighborhoods, assume each smooth arc of duration t transfers incoming error E to at most $e^{ct}(E + Ct\xi_h)$, where $c, C \geq 0$ and $\xi_h = h + \epsilon_H + \eta_h$. At event j, assume the exact aligned update Ψ_j is Λ_j-Lipschitz and its approximation satisfies $\sup_{\mathbf{x}} \left\| \widetilde{\Psi}_j(\mathbf{x}) - \Psi_j(\mathbf{x}) \right\| \leq \delta_j$. All compared states remain in these neighborhoods. Writing $A_j = \max\{1, \Lambda_j\}$, the aligned terminal error obeys*

$$E_T \leq e^{cT} \left[\left(\prod_{j=1}^{M} A_j \right) (E_0 + CT\xi_h) + \sum_{j=1}^{M} \left(\prod_{i=j+1}^{M} A_i \right) \delta_j \right]. \tag{43}$$

The corresponding prefix bound holds at every aligned time. If the event count and constants are uniform, the initial error, ξ_h, and all event defects tending to zero imply event-aligned convergence.

Proof. At event j, add and subtract Ψ_j evaluated at the approximate incoming state to obtain $E_j^+ \leq \Lambda_j E_j^- + \delta_j$. Compose this recurrence with the assumed smooth-arc bounds. Every smooth amplification is bounded by e^{cT}; replacing each event factor by $A_j \geq 1$ bounds all earlier error contributions by the full product. Smooth arc durations sum to at most T, giving Eq. (43). Truncation gives the prefix statement, and uniform bounds give convergence. □

This formulation allows inelastic event updates. Regularity means isolated, uniquely resolved events with a fixed local ordering and bounded aligned sensitivity. Event alignment compares both sides of each momentum jump; on smooth portions, a timing discrepancy θ adds at most $V\theta$ to the physical-state error when the flow speed is bounded by V. The event defects measure accuracy of the effective mixed update relative to the physical collision law.

B.1 Collision-update and perturbation evidence

We evaluate the first contact onset within the first 145 edges of each eligible HamiBalls-1 source, giving 510 predecessor states and two sampling noises. Physical one-edge perturbations use three directions per state and normalized-state RMS radii $10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}$. Event ordering changes

in 0, 1, 11, and 64 of the 1,530 source–direction cases at these radii. Finite perturbations can therefore cross event boundaries; we assess their prediction cost against separately simulated perturbed targets.

For all 510 sources, one paired direction is evaluated over complete 48-edge windows at the three larger radii, retaining both signs and both noises. The unperturbed simulator replay agrees with the stored 48-edge truth to MSE 1.30×10^{-14}. Each perturbed prediction is evaluated against its own simulated trajectory. Table 6 takes the positive part of each paired MSE increase before averaging, so improvements cannot cancel adverse changes. The unperturbed window MSE is 0.080099.

Table 6: Matched-target 48-edge collision perturbations on 510 sources. Positive cost is the mean casewise positive MSE increase divided by baseline MSE; intervals resample sources with paired signs and noises retained.

Radius	Positive cost (%)	95% interval (%)	Maximum MSE increase
10^{-5}	0.0416	[0.0221, 0.0684]	0.00770
10^{-4}	0.2098	[0.1528, 0.2786]	0.02501
10^{-3}	0.9998	[0.8310, 1.1957]	0.04176

We additionally evaluate every condition across all three directions and four radii with changed event ordering or physical paired-response gain at least 10: 139 conditions, yielding 556 signed, two-noise predictions. Matched 48-edge targets give a positive cost of 0.9113% of this subset's baseline MSE (95% interval [0.5001%, 1.7037%]); the maximum individual MSE increase is 0.04542. Within the event-order-changing subset alone, positive cost is 1.2491% [0.7046%, 2.4421%]. These conditional tail measurements quantify effects outside the unchanged-order comparisons without extending Proposition B.1 to arbitrary event sequences.

C Empirical Checks of the Theoretical Mechanisms

These HamiBalls-1 checks connect local solver properties to deployed $K = 1$ sampling: implicit regularity, refinement from deployed anchors, and complete predictions after boundary interventions. They assess Theorem 3.2 and Appendix E.2's finite-schedule effects.

Table 7: Correspondence between theoretical ingredients and supporting evidence. Analytic controls test smooth-system hypotheses; deployed-model diagnostics assess finite-step behavior and complete-sampling effects.

Theoretical ingredient	Proof or direct check	Complementary empirical evidence
PLAS-map symplecticity and implicit regularity	Symplecticity proof; measured $\mathbf{B}$ spectra and affine defects	6,881,280 deployed edge maps; learned-H local approximation order
Regular local refinement	Fixed-point and quadratic-remainder proof; nonlinear mixed control	Multi-start refinement and independently checked point Jacobians
Smooth-model and correction budgets	Conditional consistency proof; physical-step refinement control	First-order convergence with exact H, vanishing model error, and quadratic corrections
Single-refinement deployment	$K = 1$ tracking bound; finite-schedule output decomposition	$N = 8, 12, 20$ checks; 512-source, 192-edge refinement comparisons
Finite-window feedback and memory	Nonnegative response expansion with a conditional D budget	Physical-coordinate and internal-memory responses; complete RF interventions
Regular collision sequences	Finite-event composition bound	510 contact onsets; matched perturbed targets and event-boundary tails

C.1 Implicit-map health and smooth consistency controls

We measure the implicit block $\mathbf{B}$ along deployed predictions for 512 sources and two noise realizations over 192 edges. Four successive windows continue from their own predicted endpoints, covering

6,881,280 edge-map evaluations. The minimum singular value is 0.995916 and the maximum condition number is 1.005239. FP64 evaluation of the FP32-generated matrices gives a maximum entrywise symplectic defect of 2.80×10^{-7}. Instrumentation leaves the predictions unchanged. These measurements support implicit regularity at the evaluated anchors and the symplectic construction of PLAS maps.

We also test the local affine remainder using the frozen learned H. From 64 sources, two noises, and four positions per predicted window, we obtain 512 states. In FP64, we compare the affine tangent at the anchor-compatible source with an independently iterated implicit step at normalized-state RMS radii $10^{-1}, 3 \times 10^{-2}, 10^{-2}, 3 \times 10^{-3}, 10^{-3}$. The maximum implicit-equation residual is 3.14×10^{-16}; fitted error orders range from 1.9492 to 2.0545, with median 2.0020, consistent with Theorem 3.1's quadratic remainder.

An independent physical-step control uses the nonlinear nonseparable Hamiltonian

$$H^{\dagger}(q,p) = \tfrac{1}{2}(q^2 + p^2) + 0.1 \sin(q) \sin(p). \tag{44}$$

Over duration $T = 1.6$, we use 24, 48, 96, 192, and 384 physical steps from three initial states, $(0.7, -0.4)$, $(-1, 0.8)$, and $(1.4, 1)$. Each implicit step is solved by affine refinement to an anchor-increment tolerance of 10^{-13}, with a high-accuracy DOP853 solution as the physical reference. Besides exact H, we test gradient error $h(0.1, -0.07)$ and an additive per-step correction $h^2(0.03, -0.02)$. These instantiate the model-error and correction budgets in Corollary A.1.

Table 8: Physical-step convergence in the analytic Hamiltonian control. Slopes fit the maximum trajectory error against h over five resolutions; ranges span all three initial states.

Setting	Budget scaling	Observed order
Exact Hamiltonian	$\epsilon_H = \eta_h = 0$	0.9956–1.0060
Vanishing gradient error	$\epsilon_H = O(h)$	0.9973–1.0065
Quadratic step correction	$\eta_h = O(h)$	0.9953–1.0066

These controls instantiate Corollary A.1's small-step hypotheses and exhibit first-order consistency. The fixed-step learned-model checks assess implicit residuals, refinement, and complete predictions.

C.2 Protocol and physical-equation consistency

All model weights, tree structure, and calibrated coefficients are unchanged. We use Ours with current gradients and full Hessians at every anchor. Fixed-condition refinement freezes the RF state and time, diffusion predictions and tokens, observable attributes, and model parameters, and resets physical-time memory for every complete scan. It repeatedly updates the actual anchor in Eq. (20). Physical anchors are passed directly between iterations.

The $N = 20$ checks cover 512 sources, two noises, and seven first-window RF fields: left/right evaluations at intervals 2, 10, and 18 (zero-based), and the final field. Interval 2 is the first H-bearing interval after diffusion-only initialization. Each uses its deployed anchor and two opposite perturbations of normalized-anchor L2 radius 10^{-4}. The $N = 8, 12$ checks use eight sources. Convergence requires anchor and clean sup increments at most 10^{-5} for 13 consecutive iterations and clean drift at most 5×10^{-5} over nine iterations, within 64 refinements.

Table 9: Fixed-condition checks within 64 refinements. Counts include both noises. *All starts* requires all three initializations to meet the numerical criterion.

N	Sources	Conditions	Warm start	All starts
8	8	112	108	107
12	8	112	111	111
20	512	7168	7030	7015

A nonlinear Hamiltonian control includes nonzero residuals, dynamic gates, and recurrent memory. It gives measured local refinement order 1.9799 with fresh Hessians and 0.9933 with a fixed lagged

Table 10: Normalized RMS implicit-equation residuals after the second refinement. This separate panel uses eight sources for $N = 8, 12$ and 32 for $N = 20$, with two noises and seven fields.

N	Median q	Maximum q	Median p	Maximum p
8	8.55×10^{-8}	1.10×10^{-7}	8.51×10^{-8}	1.23×10^{-7}
12	8.45×10^{-8}	1.21×10^{-5}	8.54×10^{-8}	4.56×10^{-6}
20	8.41×10^{-8}	3.06×10^{-7}	8.02×10^{-8}	2.09×10^{-7}

Hessian. The fixed-point discrepancy from an independent implicit mixed rollout is at most 1.11×10^{-16} in both cases. This control tests the refinement mechanism in Theorem 3.2.

Using the norm of the signed gated correction in Eq. (27) reduces its measured budget by about 4–10% relative to the sum of component norms. This quantity enters Corollary A.1's physical-consistency bound under its smoothness and component-consistency assumptions.

C.3 Actual perturbation stability and numerical boundaries

A targeted study uses four sources, seven $N = 20$ fields, warm and terminal centers, three directions, and normalized-anchor L2 radii $10^{-2}, 10^{-3}, 10^{-4}$. Paired gain divides the distance between opposite perturbation outputs by their initial distance in the same 950-dimensional coordinates. Table 11 reports the responses after eight refinements.

Table 11: Paired anchor-distance gain after eight refinements, with 84 source–field–direction cases per row.

Center	Radius	Median gain	Maximum gain	Gain ≥ 1
Warm	10^{-2}	0.000221	0.2204	0/84
Warm	10^{-3}	0.002228	0.00518	0/84
Warm	10^{-4}	0.02429	22.0401	2/84
Terminal	10^{-2}	0.000225	0.2204	0/84
Terminal	10^{-3}	0.002186	0.00446	0/84
Terminal	10^{-4}	0.02293	22.0489	2/84

The expanding cases occur in the final field with different leaves. Replaying all expanding directions gives maximum full-trajectory RMS response 1.42×10^{-4} and positive MSE increase 2.24×10^{-7}, quantifying the prediction costs associated with these gain ratios.

In the 512-source panel, batches with an unmet 64-refinement criterion are extended to 256. There remain 137 warm-start and 150 any-start failures among 7,168 conditions; 12 conditions have cross-start anchor or clean sup distances above 10^{-4}. The largest last-nine clean span is 0.1512. Three sources with larger spans or endpoint discrepancies are additionally checked to 1,024 refinements and retain finite oscillations (last-16 spans 0.00103–0.00819). These cases fall short of the specified refinement criterion; their complete predictions are included in the flagged-source analysis below.

Actual-path tree instrumentation reproduces the deployed leaf selections without changing predictions. Two exact-threshold cells occur in one source, which is also retained in the endpoint analysis. We therefore measure boundary effects through complete sampling, including subsequent leaf selections and state updates.

Point-derivative verification. Differentiating the anchor-dependent Hessian requires third derivatives of H. The calculation expresses LayerNorm through explicit centering, variance, and reciprocal square root, and uses FP64 arithmetic. Independent finite differences check these point derivatives.

All 42 directions (14 centers, three directions each) have a resolved scale with unchanged leaves. Across five scales, the best resolved absolute discrepancy per direction has median 2.42×10^{-11} and maximum 1.91×10^{-9}; maximum relative discrepancy is 6.61×10^{-5}. Four full 950×950 Jacobians for one source give:

Point ($N = 20$)	Maximum singular value
Interval 2, left, warm	0.0155268
Interval 2, left, terminal	0.0000306129
Interval 10, left, warm	0.000535277
Interval 18, left, warm	0.000759899

Independent matrix/JVP checks agree to relative discrepancy 3.25×10^{-10}. These are FP64 point derivatives; FP64–FP32 anchor sup distances across 14 batch centers are 0.00315–0.01247, with some leaf changes. The deployed FP32 path is assessed by the complete-sampling tests below.

C.4 Single-refinement deployment and tail effects

We test whether extra within-field refinement changes complete predictions beyond the updates already carried across RF intervals. On 512 sources with $N = 20$ and two paired noises, all settings use the same accepted-left anchor schedule. Deployment uses $K = 1$; numerical references are computed with 8 and 16 within-field refinement updates. First-window MSEs for $K = 1, 8, 16$ are 0.0458448, 0.0458454, and 0.0458455. Table 12 compares $K = 1, 16$ over 192 edges, continuing each window from its own predicted endpoint without truth resets.

Table 12: 192-edge complete predictions with two paired noises. Flagged sources have at least one unmet local criterion at 64 or extended refinement, cross-start disagreement, or exact tree threshold. Positive cost is $100\,\mathbb{E}[(\ell_1 - \ell_{16})_+]/\mathbb{E}[\ell_{16}]$, taking the positive part before averaging source–noise cases.

Sources	Count	$K = 1$ MSE	$K = 16$ MSE	PhysiFormer	Positive cost (%)
All	512	0.250732	0.250635	0.340289	0.1903
Flagged	97	0.322510	0.321508	0.412714	0.4744
Other	415	0.233955	0.234069	0.323361	0.0990

The 192-edge mean $K = 1$ minus $K = 16$ MSE is 9.76×10^{-5} (source-cluster bootstrap 95% interval $[-1.55, 3.63] \times 10^{-4}$). Positive cost is 0.1903% [0.1120%, 0.2845%] overall and 0.4744% [0.2030%, 0.8123%] on flagged sources. The latter group's mean difference is 0.001002 [0.000035, 0.002167]: extra refinement has a small measurable benefit there, while its aggregate prediction advantage over PhysiFormer remains. Bootstrap resampling retains paired noises within sources; grouping by any flagged condition retains both noises.

The largest casewise $K = 1$ excess MSE is 0.07138, and some flagged cases have higher error than PhysiFormer. The source with the largest fixed-condition span has deployed MSEs 0.284685 and 0.135125 across the two noises, versus 0.285690 and 0.135129 for $K = 16$. For this source, the deployed and additional-refinement losses are closely matched. Overall, warm-started $K = 1$ approaches additional-refinement prediction accuracy beyond the updates already carried across RF intervals.

Endpoint sensitivity across RF intervals. In a separate 32-source, two-noise panel, we evaluate the 19 complete-interval hybrids in Eq. (67) for three batches covering typical behavior, warm-start failures, and the largest endpoint discrepancy. In the last case, the $K = 1$ versus $K = 8$ endpoint RMS distance is 0.039452, bounded by the propagated-increment sum 0.039702; MSE decreases from 0.025498 to 0.023068 (PhysiFormer: 0.014589). Replacing interval 5's refinement initially changes the clean output by only 5.15×10^{-6} RMS without a leaf change, but interval 17's right call changes four leaves and increases the clean difference to 0.006497. Thus, delayed amplification can arise through subsequent discrete transitions.

Anchor interventions through complete sampling. On all 512 sources and both noises, we perturb the incoming anchor at one of seven fields by both signs at normalized-anchor L2 radii $10^{-4}, 10^{-3}, 10^{-2}$, then recompute all subsequent RF fields. Across 43,008 predictions, positive MSE increases average 0.001674%, 0.001830%, and 0.002354% of the unperturbed 48-edge MSE at the three radii; 38, 42, and 42 cases change leaves. The maximum casewise increase is 0.003636.

All 152 interventions with a leaf change or 48-edge response RMS at least 10^{-4} are continued to 192 edges using paired future noise. Unperturbed controls reproduce the original predictions exactly. Positive cost is 0.3245% of this subset's baseline MSE (source-cluster 95% interval [0.0380%, 1.0226%]); the maximum individual MSE increase is 0.02411. These complete-prediction costs quantify observed boundary tails without assuming contraction across every discrete transition.

C.5 Diffusion-reset responses with full memory

We perturb all 20 physical coordinates individually by $\pm 10^{-4}$ and $\pm 10^{-3}$ at zero-based edges 0, 16, and 32 in fixed 48-edge fields. The panel contains eight sources across 12 batch fields. D predictions and affine jets are frozen, while residual and memory updates remain active. Each policy is compared to its own unperturbed rollout. Three policies are evaluated: live routing; replay of the unperturbed live gates; and $g = 1$ throughout. The 34,560 records comprise 11,520 matched perturbations per policy.

Table 13: Finite-field reset-response tails. Sup gain divides maximum coordinate response by the injected coordinate amplitude. Response energy is the mean squared subsequent normalized-state response.

Routing policy	Sup gain > 10 (%)	Maximum sup gain	Response energy
Live	0.590	1460	1.55×10^{-6}
Fixed live gates	0.286	2604	2.61×10^{-7}
All $H + r$	2.352	8448	3.26×10^{-5}

Live routing gives a smaller subsequent RMS response than all-$H + r$ in 95.23% of the matched perturbations (source-cluster 95% interval [92.11%, 98.00%]); the terminal fraction is 94.07%. The ratio of total response energies is 0.04751, with source-cluster interval [0.00444, 0.11168]. Live cases with sup gain above 10 account for 99.29% of its perturbation-response energy, whose absolute mean remains 1.55×10^{-6}. The response is therefore concentrated in a small tail while aggregate energy is substantially attenuated relative to all-$H + r$.

These fixed-field perturbations preserve the complete memory path and normalize RMS gain by the injected coordinate amplitude. They quantify the finite-window response mechanism in Proposition 3.3. Complete-sampling comparisons in Appendix C.4 and the internal-memory interventions below measure aggregate end-to-end effects.

C.6 Internal-memory responses through complete RF sampling

We additionally intervene on eight internal memory blocks: the recurrent hidden state, previous gate, previous H candidate, and five accumulated context blocks. The same eight-source panel covers RF intervals 2, 10, and 18, with both left and right calls. At physical edge 16, we perturb one block and recompute the remaining RF process, including subsequent D predictions. Each block uses both signs at amplitude 10^{-3}, giving 768 source–field–block–sign records with one sampling-noise realization. Hidden-state perturbations use alternating coordinate signs; previous-gate perturbations are clipped to $[0, 1]$. Context perturbations scale each entry by $1 + |\mathrm{entry}|$, with energy and mass entries kept nonnegative. Zero-intervention replays agree with the ordinary complete prediction within 10^{-5} maximum coordinate error.

Across paired memory interventions on eight sources, the largest aggregate MSE increase is approximately 0.060%, and the maximum individual response distance is 0.02391. These measurements quantify the aggregate prediction cost and response tails, providing empirical support for the finite-window memory-response mechanism described in Proposition 3.3.

Reproducibility. Aggregate prediction losses use FP64 subtraction and accumulation. Model weights remain fixed throughout these experiments. The full-refresh/sparse-refresh comparison is reported in Appendix E.3.

Table 14: Internal-memory perturbations followed by complete RF sampling. Each block has 96 paired cases. Response RMS compares the resulting 48-edge predictions with the unperturbed predictions. MSE changes use a common unperturbed aggregate MSE of 0.0371873; maxima retain individual tails.

Perturbed block	Maximum RMS	Aggregate MSE change (%)	Maximum MSE increase
Recurrent hidden state	0.000142	+0.000012	1.64×10^{-7}
Previous gate	0.000143	+0.000017	3.30×10^{-7}
Previous H candidate	0.000143	+0.000082	2.67×10^{-6}
Contact-provenance context	0.000142	+0.000024	1.72×10^{-7}
Cumulative reset	0.023837	+0.003641	1.26×10^{-4}
Momentum reset energy	0.023837	+0.003445	1.26×10^{-4}
Position reset energy	0.023906	+0.059833	9.23×10^{-4}
Reset mass	0.000142	+0.000011	4.15×10^{-7}

D Architecture and Training Details

D.1 Architecture and Forward Computation

D.1.1 State conventions and architecture

For spatial dimension d, each active object has $2d$ canonical state coordinates and each position or momentum readout has d outputs; HamiBalls-1 uses $d = 2$ and HamiBalls-2 uses $d = 3$.

With n active objects, $\mathbf{q}, \mathbf{p} \in \mathbb{R}^{nd}$ and the full-system canonical state has dimension $2nd$. All Hamiltonian derivatives in Eq. (5) are taken in physical canonical coordinates; fixed scaling converts candidates and corrections to and from the normalized trajectory representation used by the diffusion model and losses. Under a general normalization, the symplectic form becomes the pulled-back matrix $\mathbf{S}^\top \mathbf{J}\mathbf{S}$, as derived in Appendix A.1. The fixed normalization statistics define the conversion between physical and normalized coordinates.

Table 15: HamiBalls-1 architecture and learned-size accounting. Counts for the router include its leaf-affine terms. Linear coefficients and other learned state are reported separately from neural parameters.

Component	Architecture or state	Count
Diffusion expert	Width 128; 4 blocks; 4 heads; 2 registers; feed-forward width 256	1,159,172
Hamiltonian model	Width 16; 2 object-interaction blocks; 4 heads; scalar readout	4,464
Shared residual	SiLU MLP, $154 \to 32 \to 32 \to 4$	6,148
Router	12-dimensional projection and physical-time GRU; width-16 q/p offset heads; leaf-affine corrections	7,588
Neural subtotal	All four components	1,177,372
Ridge readouts	Fitted linear readouts	1,056
Combined scalars	Neural parameters plus fitted linear coefficients	1,178,428
Other learned state	CART tree (maximum depth four; capacity eight leaves); thresholds; feature statistics	Separate
PhysiFormer	Diffusion-only backbone; feed-forward width 265	1,173,068

The diffusion backbone alternates spatial, temporal, object, and temporal attention blocks. It uses RMS normalization, query/key normalization, SwiGLU feed-forward layers, RF-time modulation, and rotary positional encoding of physical time. There is one state token per object per frame; within-object vertex attention is therefore a singleton operation, whereas across-object attention is nontrivial. The PhysiFormer baseline is trained separately in the same model family with a larger feed-forward width. Our phase-space implementation adapts the trajectory-diffusion architecture of the original PhysiFormer study (Chen et al., 2026).

Table 16: HamiBalls-2 architecture and learned-size accounting. Totals include neural parameters and fitted ridge coefficients; tree structure, thresholds, and normalization statistics are additional state.

Component	Architecture or state	Count
Diffusion expert	Width 168; 12 blocks; 8 heads; 2 registers; feed-forward width 336	5,615,598
Hamiltonian model	Width 32; 2 object-interaction blocks; 4 heads; pair-energy readout	22,252
Shared residual and router	Token-conditioned SiLU residual; physical-time GRU with q/p offset heads and leaf-affine corrections	32,729
Neural subtotal	All HamiFormer neural components	5,670,579
Ridge readouts	Fitted linear readouts	6,960
Combined scalars	Neural parameters plus fitted linear coefficients	5,677,539
Other learned state	CART tree (maximum depth four; capacity eight leaves); thresholds; feature statistics	Separate
Transformer-AR, context 1	Comparison model	5,574,318
Transformer-AR, context 4	Comparison model	5,581,798
DHN	Two width-336, two-layer, four-head Transformers	5,607,170
DiT	Comparison model	5,654,014
PhysiFormer	Comparison model	5,615,598

On HamiBalls-2, DiT uses full spatiotemporal attention and adaLN-Zero under the matched RF protocol. Table 16 reports the component architecture and exact model sizes. HamiFormer's count includes its Hamiltonian model, shared residual, and router.

HG-DPF on HamiBalls-1 uses a parameter-matched DPF/HNN (785,300/386,386 parameters), trained for 50,000/40,000 AdamW updates with batch size 64. The Perceiver has 64 width-64 latents, two heads, and eight encoder/four decoder self-attention layers; the HNN has four width-248 CELU layers. DPF training predicts noise at 1,000 cosine levels, sampling query length $N_q \in \{5, 10, 14, 19, 24, 29, 34, 38, 43, 48\}$ and context length uniformly in $[1, N_q - 1]$. Context comprises independently noised query-prefix copies and the clean initial state. Inference uses 20 DDIM steps, random half-query context, and four-candidate SNIS ($\lambda = 1$, stochasticity 1) with standard stochastic-DDIM variance allocation and categorical selection using recomputed clean-estimate energies. Under the 512-source, two-noise, 192-edge protocol, deterministic unguided DPF/SNIS MSEs are 1.307886/1.179656: the larger error is already present without guidance.

DHN retains the reference Hamiltonian update and denoising routines (Deng et al., 2026), with block size and stride one. Observable scene attributes replace the codebook conditioning; padded coordinates are masked, and position/momentum scales are fitted on training data. Training uses the equations-of-motion loss plus 0.1 times the denoising loss, with the reference coordinatewise corruption. Adam uses learning rate 10^{-4} and weight decay 10^{-4} for 50,000 updates, each averaging valid-object losses over all 48 transitions in 64 training windows. At inference, each physical edge uses 11 alternating right/left Hamiltonian denoising rounds at levels $0, 0.1, \ldots, 1$, conditioned on the preceding predicted state. Evaluation uses the same 512 sources, two noise seeds, and 192-edge horizon.

D.1.2 Observable features and model-tree semantics

The shared residual reads a diffusion token together with the current noisy state, initial state, mixed predecessor, candidates, and object attributes. A separate feature construction summarizes candidate increments and disagreements, geometry and recovery descriptors, RF level, attributes, and available history. Its inference-time routing history ends at the previous physical edge, and all remaining inputs are current causal observables.

The tree is learned once after the second scalar-router stage, with maximum depth four and capacity for eight leaves. CART splits on observable features to reduce normalized squared-error objectives for candidate mixing and Hamiltonian correction in both position and momentum. For feature vector $\boldsymbol{\phi} \in \mathbb{R}^m$, the design vector is

$$\boldsymbol{\psi}(\boldsymbol{\phi}) = [\mathrm{clip}\left((\boldsymbol{\phi} - \boldsymbol{\mu}_{\boldsymbol{\phi}}) \oslash \mathbf{s}_{\boldsymbol{\phi}}, -8, 8\right); 1] \in \mathbb{R}^{m+1}, \tag{45}$$

Algorithm 1 Mixed clean estimate with one PLAS refinement ($K = 1$)

Require: Noisy future block $\mathbf{X}_\tau$, RF time τ, initial state $\mathbf{z}_0$, attributes $\mathbf{c}$, frozen mixed-coordinate anchor $\mathbf{a}$, diffusion-only indicator b_D

Evaluate $D(\mathbf{X}_\tau, \tau; \mathbf{z}_0, \mathbf{c})$; retain its clean trajectory and tokens.
if b_D **then**
  Form $\mathbf{a}^D$ from D predecessor positions and D target momenta, keeping $\mathbf{q}_0$ fixed.
  return the normalized diffusion clean trajectory and $\mathbf{a}^D$.
end if
Construct all $(\mathbf{A}_k, \mathbf{b}_k)_{k=1}^L$ from the fixed anchors, using Eqs. (5) and (6).
Set $\widehat{\mathbf{z}}_0 = \mathbf{z}_0$ and reset physical-time router history.
for $k = 1, \ldots, L$ **do**
  Form $\widehat{\mathbf{z}}_k^H = \mathbf{A}_k \widehat{\mathbf{z}}_{k-1} + \mathbf{b}_k$ and obtain $\widehat{\mathbf{z}}_k^D$.
  Evaluate the shared residual and observable feature construction.
  Route to $\ell = \mathcal{T}(\boldsymbol{\phi})$ and evaluate $\Delta\mathbf{r}_\ell = \mathbf{W}_\ell \boldsymbol{\psi}$.
  if $k = 1$ **then**
    Set $\Delta\mathbf{r}_\ell = \mathbf{0}$.
  end if
  Form $\mathbf{r}_k$ using Eq. (8), then form corrected $\widehat{\mathbf{z}}_k^{H+r}$ as in Eq. (3).
  Evaluate the recurrent router on the corrected candidate and compute $\mathbf{g}_k$.
  Mix with Eq. (3); update physical-time history using this edge.
end for
Form $\mathbf{a}^+ \leftarrow ((\widehat{\mathbf{q}}_{k-1})_{k=2}^L, (\widehat{\mathbf{p}}_k^H)_{k=1}^L)$ in physical coordinates.
return the full normalized mixed trajectory and $\mathbf{a}^+$.

where division is componentwise and the positive normalization scales are fixed during deployment. Each leaf has separate position and momentum readouts with output dimension d. After fitting the leaf readouts, $\alpha_q, \alpha_p \in [0, 1]$ are fitted separately by closed-form constrained least squares against the required Hamiltonian corrections, using held-out sources within the training split, and fixed during inference. On the first physical edge, setting $\Delta\mathbf{r}_\ell = \mathbf{0}$ yields the shared contribution $(\mathbf{1} - \boldsymbol{\alpha}) \odot \mathbf{r}_0$. The leaf correction to the router is affine in standardized current observables and an intercept. For numerical evaluation of a base probability's logit, probabilities are clamped to $[10^{-6}, 1 - 10^{-6}]$; this clamp serves numerical stability.

Tree structure, feature standardization, and component coefficients are fitted from the training split and fixed during inference.

D.1.3 One complete mixed clean estimate

Algorithm 1 specifies the causal ordering within a field evaluation. Anchors are mixed-coordinate pairs $(\bar{\mathbf{q}}_{k-1}, \bar{\mathbf{P}}_k)$, with $\bar{\mathbf{q}}$ from the committed mixed predecessor and $\bar{\mathbf{P}}$ from its uncorrected H candidate. The first position is the observed $\mathbf{q}_0$. The derivative construction is batched over physical edges. The subsequent scan is sequential because both the residual/router and the next Hamiltonian candidate depend on the newly mixed state.

D.1.4 Stateful RF sampler

The sampler carries the noisy RF trajectory and a separate mixed-coordinate anchor. Both evaluations of a Heun interval receive the same incoming anchor, which remains frozen throughout the interval. After numerical acceptance, the anchor returned by the left field evaluation is committed. Algorithm 2 specifies this ordering.

The first two *complete intervals* use the diffusion expert for both Heun evaluations. The hybrid part consequently begins at $\tau = 2/N$, namely 0.25, $1/6$, or 0.10 for $N = 8, 12, 20$. The corresponding numbers of field evaluations are $15, 23, 39$. Each hybrid field evaluation produces an entire L-edge physical rollout; N counts RF sampling intervals.

The deployed sampler checks $1/N \geq 0.05$. Therefore the last Euler update has $d_{\tau_{N-1}} = \Delta$, and $\mathbf{X}_N = \mathbf{Y}^L$ on that interval. Appendix E.2 gives finite-schedule perturbation accounting with this denominator floor.

Algorithm 2 RF sampling with accepted-left mixed-coordinate anchor commits

Require: Initial state $\mathbf{z}_0$, attributes $\mathbf{c}$, intervals N, denominator floor $\varepsilon_\tau = 0.05$

Draw $\mathbf{X}_0 \sim \mathcal{N}(\mathbf{0}, \mathrm{I})$; set $\Delta = 1/N$ and initialize the anchor from repeated $\mathbf{z}_0$.
for $s = 0, \ldots, N-1$ **do**
  $\tau_s = s/N$; $b_D \leftarrow (s < 2)$.
  $(\mathbf{Y}^L, \mathbf{a}^L) \leftarrow \mathrm{Clean}(\mathbf{X}_s, \tau_s, \mathbf{z}_0, \mathbf{c}, \mathbf{a}, b_D)$ using Algorithm 1.
  $\mathbf{v}^L \leftarrow (\mathbf{Y}^L - \mathbf{X}_s)/\max(1 - \tau_s, \varepsilon_\tau)$.
  if $s = N - 1$ **then**
    $\mathbf{X}_{s+1} \leftarrow \mathbf{X}_s + \Delta \mathbf{v}^L$.
  else
    $\mathbf{P}_s \leftarrow \mathbf{X}_s + \Delta \mathbf{v}^L$.
    $(\mathbf{Y}^R, \mathbf{a}^R) \leftarrow \mathrm{Clean}(\mathbf{P}_s, \tau_{s+1}, \mathbf{z}_0, \mathbf{c}, \mathbf{a}, b_D)$.
    $\mathbf{v}^R \leftarrow (\mathbf{Y}^R - \mathbf{P}_s)/\max(1 - \tau_{s+1}, \varepsilon_\tau)$.
    $\mathbf{X}_{s+1} \leftarrow \mathbf{X}_s + \frac{\Delta}{2}(\mathbf{v}^L + \mathbf{v}^R)$.
  end if
  $\mathbf{a} \leftarrow \mathbf{a}^L$ after accepting the interval; keep $\mathbf{q}_0$ fixed.
end for
return $\mathbf{X}_N$.

D.2 Training Objectives and Schedule

The staged objectives and carrier schedule below apply to both HamiBalls datasets.

D.2.1 Joint base-model training

The joint base stage minimizes $\mathcal{L}_{\mathrm{base}} = \mathcal{L}_D + \mathcal{L}_H$ with the settings in Table 17. RF time follows a logit-normal distribution with $\mu_\tau = -0.8$, $\sigma_\tau = 0.8$. Both base networks remain frozen throughout downstream training.

Table 17: Training settings for HamiBalls-1 (H1) and HamiBalls-2 (H2).

Setting	H1	H2
Base updates; batch size	50,000; 64	50,000; 64
Optimizer; weight decay; grad clip	AdamW; 10^{-4}; 1.0	AdamW; 10^{-4}; 1.0
Peak learning rate: D / H	3×10^{-4} / 10^{-4}	3×10^{-4} / 10^{-4}
Peak LR: residual / scalar / final q/p	3×10^{-4} / 3×10^{-4} / 3×10^{-3}	10^{-3} / 3×10^{-3} / 3×10^{-3}
RF denominator floor	0.05	0.05
Downstream stages	4×200 updates	4×200 updates
Local replay length; cache refresh	12 edges; 8 updates	12 edges; 8 updates
Tree capacity	depth 4; 8 leaves	depth 4; 8 leaves
Residual weights (local; recovery/hull)	$1/4$; $1/8$	$1/4$; $1/8$
No-harm weight; relative ridge	0.25; 0.01	0.25; 0.01

Learning-rate schedules use linear warmup and cosine decay where applicable.

The Hamiltonian contribution $\mathcal{L}_H$ uses local adjacent-state relations. For an observed adjacent pair $(\mathbf{z}_k, \mathbf{z}_{k+1})$, let $\mathbf{m}_k = (\mathbf{z}_k + \mathbf{z}_{k+1})/2$, let $\mathbf{s}_z$ denote the fixed state scale, and define the normalized relation residual

$$\mathbf{e}_k = (h\mathbf{J}\nabla H_\theta(\mathbf{m}_k) - (\mathbf{z}_{k+1} - \mathbf{z}_k)) \oslash \mathbf{s}_z. \tag{46}$$

For each canonical coordinate, a fixed anisotropic scale a_j is the median absolute normalized adjacent-state increment over the training sources, edges, and objects. Degenerate scales are rejected. With $\langle\cdot\rangle$ denoting an average over coordinates, the robust weight for object i at edge k and loss are

$$u_{k,i} = \left\langle (\mathbf{e}_{k,i} \oslash \mathbf{a})^2 \right\rangle, \quad w_{k,i} = \frac{\nu}{\nu + u_{k,i}}, \quad \nu = 4, \quad \mathcal{L}_H = \frac{\nu+1}{\nu} \mathbb{E}_{k,i} \left[\mathrm{sg}(w_{k,i}) \frac{1}{2} \langle \mathbf{e}_{k,i}^2 \rangle \right]. \tag{47}$$

Here sg denotes stop-gradient. The anisotropic scale determines retention weights, while the quadratic term uses the original normalized residual. This iteratively reweighted objective is evaluated on sampled observable adjacent-state pairs.

D.2.2 SHARED-RESIDUAL LEARNING

The residual is freshly initialized and trained for 200 updates with fixed H/D, using 12-edge local replay within cached 48-edge random-gate trajectories refreshed every eight updates. For $c \in \{q, p\}$, let E_c, I_c be the coordinate-mean normalized squared correction error and ideal-correction energy at each source–edge–object entry on the local teacher-forced branch. Its local term is

$$\mathcal{L}_{\mathrm{local},c} = \mathbb{E}_{b,k,i}\left[\frac{\log(1+E_c)}{1+I_c} + \left[\sqrt{E_c} - \sqrt{I_c}\right]_+\right], \tag{48}$$

where $[a]_+ = \max(a, 0)$ and $\mathbb{E}_{b,k,i}$ averages valid source–edge–object entries. For closed-loop terms, use corrected candidate $\mathbf{C}_c$, diffusion candidate $\mathbf{D}_c$, target $\mathbf{z}_c^*$, and fixed scales $\mathbf{s}_c > \mathbf{0}$:

$$\mathbf{v}_c = (\mathbf{C}_c - \mathbf{D}_c) \oslash \mathbf{s}_c, \quad \mathbf{u}_c = (\mathbf{z}_c^* - \mathbf{D}_c) \oslash \mathbf{s}_c, \quad \ell_{\mathrm{PH}}(\mathbf{u}) = \sqrt{1 + \langle \mathbf{u}^2 \rangle} - 1. \tag{49}$$

$$\theta_c = \mathrm{sg}\left[\mathrm{clip}\left(\frac{\mathbf{u}_c^\top \mathbf{v}_c}{\max(\|\mathbf{v}_c\|_2^2, \epsilon_{\mathrm{hull}})}, 0, 1\right)\right]. \tag{50}$$

$$\mathcal{L}_{\mathrm{recovery},c} = \mathbb{E}_{b,k,i}[\ell_{\mathrm{PH}}(\mathbf{v}_c - \mathbf{u}_c)], \qquad \mathcal{L}_{\mathrm{hull},c} = \mathbb{E}_{b,k,i}[\ell_{\mathrm{PH}}(\theta_c \mathbf{v}_c - \mathbf{u}_c)]. \tag{51}$$

Here $\epsilon_{\mathrm{hull}} > 0$ is a numerical floor. The target, diffusion candidate, and projection coefficient are detached; residual learning differentiates through $\mathbf{C}_c$. Recovery penalizes corrected-candidate error, while hull penalizes the target's distance to the candidate segment. The total objective is

$$\mathcal{L}_r = \frac{1}{4}\left(\mathcal{L}_{\mathrm{local},q} + \mathcal{L}_{\mathrm{local},p}\right) + \frac{1}{8}\sum_{c\in\{q,p\}}\left(\mathcal{L}_{\mathrm{recovery},c} + \mathcal{L}_{\mathrm{hull},c}\right). \tag{52}$$

D.2.3 ROUTER LOSSES: ENDPOINT PREFERENCE, RECURRENT REGRET, AND RISK

A training cell is indexed by source, physical edge, and object; $c \in \{q, p\}$ denotes its d-coordinate component. Let $\mathbf{D}_c, \mathbf{C}_c$ denote its diffusion and residual-corrected Hamiltonian candidates, and let $\mathbf{z}_c^*$ be the target. For a gate g, write $\mathbf{M}_c(g) = \mathbf{D}_c + g(\mathbf{C}_c - \mathbf{D}_c)$. All targets, labels, normalizing headroom, and oracle coefficients described below are detached when used as supervision.

Endpoint preference. Let $e_D = \mathrm{MSE}(\mathbf{D}_c, \mathbf{z}_c^*)$, $e_C = \mathrm{MSE}(\mathbf{C}_c, \mathbf{z}_c^*)$ in normalized state coordinates. Define the binary preferred endpoint $y = \mathbf{1}[e_C < e_D]$ and the endpoint regret $a = |e_D - e_C|$. Exact ties receive zero weight. When both endpoint classes have positive regret mass, each class receives total weight one half, distributed proportionally to a within that class. A single active class receives total weight one. For normalized weights ω_j and router logits λ_j, the loss is

$$\mathcal{L}_{\mathrm{end},c} = \sum_j \omega_j \left[\log(1 + \exp(\lambda_j)) - y_j \lambda_j\right]. \tag{53}$$

An all-tie batch contributes zero. The label encodes which candidate endpoint has lower MSE.

Global recurrent projection regret. For the final dual-channel recurrent term, position error is evaluated in physical position units and momentum error in one-step position-equivalent units, $h\,\Delta\mathbf{p}/m_i$. In either component, let $\mathbf{a}_j$ be the diffusion error and $\mathbf{b}_j$ the candidate disagreement in these units. Define

$$E_j(g) = \langle(\mathbf{a}_j + g\mathbf{b}_j)^2\rangle, \qquad \theta_j = \mathrm{clip}\left(-\frac{\langle \mathbf{a}_j \mathbf{b}_j\rangle}{\langle \mathbf{b}_j^2\rangle}, 0, 1\right), \tag{54}$$

with any minimizing value, chosen as zero, when the disagreement vanishes. The pooled recurrent regret is

$$\mathcal{L}_{\mathrm{rec},c} = \frac{\sum_j [E_j(g_j) - E_j(\theta_j)]_+}{\sum_j [E_j(1/2) - E_j(\theta_j)]_+ + \epsilon_{\mathrm{rec}}}. \tag{55}$$

Normalization uses pooled midpoint headroom with a stabilizer $\epsilon_{\mathrm{rec}} > 0$. Physical-time router recurrence is retained when computing g_j. The first scalar-router stage uses the analogous global sequence projection regret for its single shared gate; the second adds the componentwise penalty below. Cached carrier trajectories are treated as fixed data within each optimization block.

Componentwise non-degradation penalty. For the componentwise penalty, errors $e_{bki}(g)$ are MSEs in normalized state coordinates, for source b, edge k, and object i. With a bar denoting the mean over edges,

$$\mathcal{L}_{\mathrm{nh},c} = \mathrm{mean}_{b,i} \frac{\overline{[e_{bki}(g) - e_{bki}(0)]_+}}{\mathrm{sg}\left(\overline{[e_{bki}(1) - e_{bki}(0)]_+}\right)}. \tag{56}$$

A ratio is defined to be zero when its denominator is at most machine epsilon; active denominators are numerically clamped. This empirical penalty averages positive degradation over physical edges. The second scalar-router stage and the final stage use weight 0.25 for the corresponding component-averaged penalty.

Momentum tail term. Using the position-equivalent errors E from the recurrent term, first average the error *difference* over physical edges and then take its positive part:

$$\begin{aligned} h_{bi} &= \left[\overline{E_{bki}(g) - E_{bki}(0)}\right]_+, \qquad h^H_{bi} = \left[\overline{E_{bki}(1) - E_{bki}(0)}\right]_+, \\ \mathcal{L}_{\mathrm{tail},p} &= \frac{\mathrm{RMS}_{b,i}(h)}{\mathrm{sg}(\mathrm{RMS}_{b,i}(h^H))}. \end{aligned} \tag{57}$$

The same zero-denominator convention applies, and the zero vector uses the canonical zero subgradient. Equation (57) averages over time before taking the positive part, while Eq. (56) takes the positive part first. The tail term preserves the absolute severity of source–object harm relative to a batch-level scale.

Final objective. The two component objectives and their joint loss are

$$\mathcal{L}_q = \tfrac{1}{2}(\mathcal{L}_{\mathrm{end},q} + \mathcal{L}_{\mathrm{rec},q}) + \tfrac{1}{4}\mathcal{L}_{\mathrm{nh},q}, \tag{58}$$

$$\mathcal{L}_p = \tfrac{1}{3}(\mathcal{L}_{\mathrm{end},p} + \mathcal{L}_{\mathrm{rec},p} + \mathcal{L}_{\mathrm{tail},p}) + \tfrac{1}{4}\mathcal{L}_{\mathrm{nh},p}, \tag{59}$$

$$\mathcal{L}_g = \tfrac{1}{2}(\mathcal{L}_q + \mathcal{L}_p). \tag{60}$$

The tail term is applied to momentum. These soft penalties target candidate preference and non-degradation on the training carriers. Proposition 3.3 describes the finite-window diffusion and propagated error budgets.

D.2.4 Conditional ridge fitting and training partitions

For a fixed carrier and tree, let $\mathbf{y}_j$ be the normalized target correction from the pure Hamiltonian candidate and $\boldsymbol{\psi}_j$ the standardized feature vector with intercept. Write the hierarchical fit as a common matrix $\mathbf{W}_0$ plus leaf deviation $\mathbf{V}_\ell$:

$$\min_{\mathbf{W}_0,\{\mathbf{V}_\ell\}} \sum_j \omega_j \left\|\mathbf{y}_j - (\mathbf{W}_0 + \mathbf{V}_{\ell(j)})\boldsymbol{\psi}_j\right\|^2 + \lambda \left(\left\|\mathbf{W}_0\mathbf{P}^{1/2}\right\|_F^2 + \sum_\ell \left\|\mathbf{V}_\ell\mathbf{P}^{1/2}\right\|_F^2 \right), \tag{61}$$

Here ω_j equalizes total weight across physical sources and is constant for equal row counts; $\mathbf{P}$ is diagonal with unit feature penalties and intercept penalty 0.01. The relative ridge strength is 0.01 times the mean diagonal of the common-block design Gram matrix. Fitting is componentwise on collected carriers; deployment uses the merged leaf matrices $\mathbf{W}_\ell = \mathbf{W}_0 + \mathbf{V}_\ell$ and the calibrated blend in Eq. (8). The final stage accumulates design sufficient statistics across refreshes.

The final router stage trains the added q/p heads and leaf-affine routing terms, with the scalar trunk frozen. Each final refresh contains 64 physical sources, split into two *disjoint* 32-source groups with independent noise streams. Each group contains 16 sources sampled with $N = 8$, 10 with $N = 12$, and 6 with $N = 20$. The two scalar stages train on fixed pools generated by their preceding policies, with source, noise, and random-routing assignments fixed within each pool.

Tree fitting, calibration, and ridge solves add computation outside the optimizer-update count in Table 18.

Table 18: Training sequence. The diffusion and Hamiltonian losses form one 50,000-update base stage. A refresh regenerates rollout carriers. Fixed previous-stage policies generate the scalar-stage pools.

Trainable component	Updates	Objective and carrier policy
Diffusion + Hamiltonian base	50,000	$\mathcal{L}_D + \mathcal{L}_H$; whole-window RF (batch 64) and local relation mini-batches
Shared residual	200	Eq. (52); random-gate carriers; refresh every 8 updates
First scalar router	200	Global sequence projection regret; fixed mixed-N pool
Second scalar router	200	Projection regret plus non-degradation penalty; fixed pool from the preceding scalar router
Model tree	—	Fit once after scalar stages outside gradient-based optimization
Dual-channel/leaf router	200	Eq. (60); 25 refreshes of 8 updates; cumulative ridge fitting

E Numerical Methods and Computational Efficiency

This appendix records numerical consequences for the implemented sampler and the optional low-frequency Hessian refresh. These analyses use the same distinction between anchor transition $\mathcal{R}_{\mathbf{y}}$ and clean readout $\mathcal{C}_{\mathbf{y}}$ as Eq. (20).

E.1 Integrator Variants and Evaluation Protocol

The eight sampled zero-based physical edges are $\{0, 7, 13, 20, 27, 34, 40, 47\}$. Each solver is evaluated at its own realized mixed predecessors on the same 512 sources and two noise realizations. The reference is an independently converged FP64 Newton solve, requiring normalized residual and increment below 10^{-12} and a change below 10^{-12} after two additional iterations. All reference checks pass. SymEuler2 performs two momentum fixed-point updates followed by a final position-gradient evaluation. For each component, Table 3 reports the square root of the mean squared error over the sampled entries and accepted RF fields. The left and center panels of Figure 5 report the arithmetic cumulative mean of the separately aggregated per-field RMSEs. Their final ordinates therefore have a different aggregation from the table.

Timings are medians of 11 warmed, synchronized B64×48 sampling calls on an A800 80GB PCIe GPU with $N = 20$. Model loading, data I/O, compilation, and graph capture are excluded. Each solver computes its candidate before the same compiled residual/router stage; applicable derivative compilation, kernel autotuning, and attention contractions are shared across methods. The comparison uses FP32 model evaluation. PLAS-DC and PLAS-DC-S accumulate the defect correction in FP64; PLAS-DC-F uses FP32 correction arithmetic. End-to-end Ours timings using the deployed scan are reported separately in Appendix E.4.

Unified defect correction. Work in physical coordinates and suppress the physical-edge index. Let $\widetilde{\mathbf{A}}\mathbf{z} + \widetilde{\mathbf{b}}$ be the selected affine jet from Eq. (6), with blocks $\widetilde{\mathbf{B}}, \widetilde{\mathbf{V}}$. These blocks may be current or cached as specified below; assume $\widetilde{\mathbf{B}}$ is nonsingular. For the actual mixed predecessor $\mathbf{z} = [\mathbf{q}^\top, \mathbf{p}^\top]^\top$, first obtain the affine momentum seed and its implicit-equation defect:

$$\widehat{\mathbf{P}} = [\mathbf{0} \quad \mathrm{I}]\,(\widetilde{\mathbf{A}}\mathbf{z} + \widetilde{\mathbf{b}}), \qquad \mathbf{d} = \mathbf{p} - \widehat{\mathbf{P}} - hH_{\mathbf{q}}(\mathbf{q}, \widehat{\mathbf{P}}). \tag{62}$$

The full tangent correction is

$$\widetilde{\mathbf{B}}\,\boldsymbol{\delta} = \mathbf{d}, \qquad \begin{bmatrix}\mathbf{Q}_{\mathrm{DC}} \\ \mathbf{P}_{\mathrm{DC}}\end{bmatrix} = \begin{bmatrix}\mathbf{q} + hH_{\mathbf{p}}(\mathbf{q}, \widehat{\mathbf{P}}) \\ \widehat{\mathbf{P}}\end{bmatrix} + \begin{bmatrix}\widetilde{\mathbf{V}} \\ \mathrm{I}\end{bmatrix}\boldsymbol{\delta}. \tag{63}$$

Both Hamiltonian gradients are freshly evaluated at $(\mathbf{q}, \widehat{\mathbf{P}})$. The momentum update is a Newton-type correction using the selected approximation to $\mathrm{I} + hH_{\mathbf{qp}}(\mathbf{q}, \widehat{\mathbf{P}})$; the position update transports the same increment through the selected approximation to $hH_{\mathbf{pp}}$. Thus both components are corrected by one coupled update. In the implementation, the momentum block column of $\widetilde{\mathbf{A}}$ has upper and lower blocks $\widetilde{\mathbf{V}}\widetilde{\mathbf{B}}^{-1}$ and $\widetilde{\mathbf{B}}^{-1}$, respectively, so its multiplication by $\mathbf{d}$ supplies both increments without another Hessian evaluation. The resulting candidate is normalized and passed once to the residual/router stage before advancing the mixed scan.

Refresh policies. In the method names, DC denotes *defect correction*, F denotes a *frozen affine jet*, and S denotes *sparse Hessian refresh*. The frozen-jet variant holds both the affine matrix and offset fixed within a sampling chunk; the sparse-Hessian variants recompute anchor gradients and offsets while refreshing the Hessian intermittently.

Each physical edge has its own affine jet. PLAS constructs these jets using current anchor gradients and complete Hessians and applies the affine map directly. PLAS-DC uses the same current jets and applies Eqs. (62)–(63). PLAS-DC-F constructs the window's jets at the first Hamiltonian-bearing anchor and holds their matrices and offsets fixed throughout that sampling chunk; the gradients in Eqs. (62)–(63) remain fresh at every physical step.

PLAS-S and PLAS-DC-S use the lagged-Hessian construction in Appendix E.3: they recompute anchor gradients and affine offsets at every distinct anchor, while refreshing the complete Hessian at anchor ordinals $0, 16, 32, \ldots$. Exact reuse of an identical anchor does not advance this counter. The Hessian cache and counter reset at the start of each sampling chunk. PLAS-S applies the resulting affine map directly; PLAS-DC-S additionally applies Eqs. (62)–(63), including fresh physical-step gradients. All variants retain the same RF schedule, weights, residuals, and routing rules.

The underlying affine maps retain the PLAS symplectic identity. Defect correction augments these maps through Eqs. (62)–(63). The RMSE values and Pareto boundary report measured point estimates under the arithmetic settings above.

End-to-end prediction under integrator substitutions. Table 19 evaluates 512 HamiBalls-1 sources with two paired noises over four 48-edge windows, initialized at episode start and continued from predicted endpoints. PLAS achieves prediction MSE close to SymEuler2 at lower runtime than either sequential solver. Against Explicit Euler, its 192-edge MSE is 1.29% lower, with 10.5% lower sampling time; its sampling time is 37.7% lower than SymEuler2's.

Table 19: HamiBalls-1 prediction accuracy and sampling time (↓). MSE uses 512 sources and two noises; wall-clock is per B64×48 sampling call under the shared timing protocol of Table 3.

Integrator	48-edge z-MSE	192-edge z-MSE	Wall-clock (s)
Explicit Euler	0.045910	0.253974	0.8875
SymEuler2	0.045846	0.250647	1.2752
PLAS	0.045850	0.250707	**0.7940**

E.2 Finite-refinement deployment error

Anchor and field errors. For a field call that performs K refinements, its output is $\mathcal{C}_{\mathbf{y}}^{[K]}(\mathbf{a}) = \mathcal{C}_{\mathbf{y}}(\mathcal{R}_{\mathbf{y}}^{K-1}(\mathbf{a}))$. A converged reference uses $\mathcal{C}_{\mathbf{y}}(\mathbf{a}^*(\mathbf{y}))$. Let $e_0 = \|\mathbf{a} - \mathbf{a}^*\| \le r$ in an invariant regular neighborhood with quadratic constant $C > 0$ and $Cr < 1$. Induction on Theorem 3.2 gives the finite-K bound

$$\left\|\mathcal{R}_{\mathbf{y}}^{K}(\mathbf{a}) - \mathbf{a}^*\right\| + \left\|\mathcal{C}_{\mathbf{y}}^{[K]}(\mathbf{a}) - \mathcal{C}_{\mathbf{y}}(\mathbf{a}^*)\right\| \le C^{-1}(Ce_0)^{2^K}.$$

The induction uses $e_{j+1} \le Ce_j^2$ and the last-update readout bound. For $K = 1$, $e_0 \le e_{s-1} + d_s$ (Appendix A.4). Apply the fixed normalization and divide by $d_\tau \ge \varepsilon_\tau$ to bound field error, comparing each call at its own RF state.

For a more general local tracking estimate $e_s \le \gamma_s e_{s-1} + \gamma_s d_s + b_s$, where d_s bounds target motion and b_s includes implementation error or a bounded branch jump, repeated substitution gives

$$e_s \le \left(\prod_{i=1}^{s}\gamma_i\right) e_0 + \sum_{j=1}^{s}\left(\prod_{i=j+1}^{s}\gamma_i\right)(\gamma_j d_j + b_j). \tag{64}$$

The gain and jump bounds apply along the compared states; suffix products quantify downstream amplification. Within regular regions, b_s can account for finite precision; at hard-tree crossings, b_s additionally accounts for the bounded discrepancy between branches.

Conditional Heun perturbation bound. Let $\mathbf{v}^*$ be the converged-refinement field, L_v-Lipschitz in state on a region containing both sampling trajectories and their predictors. Let ξ_s^L, ξ_s^R bound approximate-field discrepancies from $\mathbf{v}^*$ at the actual left state and actual right predictor. Comparing the actual predictor to the exact-field predictor from the same left state gives a distance at most $\Delta_s \xi_s^L$. Therefore the one-interval update discrepancy is at most

$$\Delta_s w_s, \qquad w_s = \tfrac{1}{2}\big[(1 + L_v \Delta_s)\xi_s^L + \xi_s^R\big] \tag{65}$$

for Heun; on the final Euler interval $w_s = \xi_s^L$. The exact-field Heun update has state Lipschitz bound $1 + L_v\Delta_s + \frac{1}{2}L_v^2\Delta_s^2$. With equal initial states and $\sum_s \Delta_s = 1$, unrolling the interval recurrence yields

$$\left\|\mathbf{X}_N - \widetilde{\mathbf{X}}_N\right\| \le \exp\big(L_v + \tfrac{1}{2}L_v^2\Delta_{\max}\big) \sum_s \Delta_s w_s. \tag{66}$$

This finite-schedule estimate applies on regions with the stated field regularity. Discontinuities at hard-tree crossings enter through explicit jump terms; the full-sampling comparisons measure their downstream effects.

Exact paired-sampling decomposition. Let F_j, G_j denote the $S = N - 2$ hybrid RF-interval transitions for $K = 1$ and a finite-K comparator on extended state (RF trajectory, committed anchor, and schedule metadata), including both Heun calls and the final readout. From the common state $\mathbf{x}_0$ after two diffusion-only intervals, let $\mathbf{P}$ extract the final trajectory and define

$$\mathbf{h}_j = \mathbf{P}\, G_S \cdots G_{j+1} F_j \cdots F_1(\mathbf{x}_0), \qquad \Delta\mathbf{h}_j = \mathbf{h}_j - \mathbf{h}_{j-1}, \qquad B_{\text{end}} = \sum_{j=1}^{S} \|\Delta\mathbf{h}_j\|_{\text{RMS}}. \tag{67}$$

Cancellation gives $\mathbf{h}_S - \mathbf{h}_0 = \sum_j \Delta\mathbf{h}_j$ and hence $\|\mathbf{h}_S - \mathbf{h}_0\|_{\text{RMS}} \le B_{\text{end}}$. With $\mathcal{L}(\mathbf{h}) = \big\|\mathbf{h} - \mathbf{Z}^\dagger\big\|_{\text{RMS}}^2$,

$$\mathcal{L}(\mathbf{h}_S) - \mathcal{L}(\mathbf{h}_0) = \sum_j \big[\mathcal{L}(\mathbf{h}_j) - \mathcal{L}(\mathbf{h}_{j-1})\big], \qquad |\mathcal{L}(\mathbf{h}_S) - \mathcal{L}(\mathbf{h}_0)| \le 2\sqrt{\mathcal{L}(\mathbf{h}_0)}B_{\text{end}} + B_{\text{end}}^2. \tag{68}$$

These identities apply to finite-refinement sampling paths, including discrete tree transitions: every downstream gate, leaf, memory update, and diffusion evaluation is recomputed. They measure endpoint discrepancies, separately from the local convergence guarantee. The comparator uses $K = 8$; the 57 complete hybrid evaluations in Appendix C.4 use this procedure.

E.3 Low-frequency Hessian refresh

This optional implementation retains current Hamiltonian gradients while reusing a symmetric Hessian from an earlier RF field. The approximation is temporal: all Hessian blocks are retained at a refresh.

Proposition E.1 (Structure and defect with a lagged Hessian)**.** *At the current anchor $\bar{\mathbf{v}} = (\bar{\mathbf{q}}, \bar{\mathbf{P}})$, replace $\nabla^2 H(\bar{\mathbf{v}})$ by a symmetric matrix $\widetilde{\mathbf{H}}$ while retaining $\nabla H(\bar{\mathbf{v}})$. Form the affine map from these blocks and assume $\widetilde{\mathbf{B}} = \mathrm{I} + h\widetilde{\mathbf{H}}_{\mathbf{qP}}$ is invertible. The resulting fixed-anchor affine map is symplectic. If $\nabla^2 H$ is M_H-Lipschitz on the segment between $\bar{\mathbf{v}}$ and the realized $\mathbf{v} = (\mathbf{q}, \mathbf{P})$, its implicit-equation defect obeys*

$$\|\mathbf{d}\| \le h\left\|\nabla^2 H(\bar{\mathbf{v}}) - \widetilde{\mathbf{H}}\right\| \|\mathbf{v} - \bar{\mathbf{v}}\| + \tfrac{1}{2}hM_H \|\mathbf{v} - \bar{\mathbf{v}}\|^2 + \|\mathbf{e}_{\text{lin}}\|, \tag{69}$$

where $\mathbf{e}_{\text{lin}}$ is the equation residual of the affine linear solve. In exact arithmetic, a consistent anchor has zero defect.

Proof. Symmetry of the diagonal Hessian blocks and transposed cross blocks gives the factorization in Eq. (14), with the replaced blocks. Translation does not affect that identity. For the defect, subtract the lagged linearized equations from the exact implicit equations, add and subtract $\nabla^2 H(\bar{\mathbf{v}})(\mathbf{v} - \bar{\mathbf{v}})$, and bound the Taylor remainder as in Eq. (23). □

If $\widetilde{\mathbf{H}} = \nabla^2 H(\mathbf{v}_{\text{old}})$, the first matrix norm is at most $M_H \|\bar{\mathbf{v}} - \mathbf{v}_{\text{old}}\|$. This first-order term quantifies the effect of lagged curvature on the local refinement budget.

Paired HamiBalls-1 validation. We compare full refresh with the period-16 PLAS-S policy in Appendix E.1 on matched 48-edge inputs: 512 sources, two noise realizations, $N = 20$, and unchanged weights and other settings. Table 20 uses FP64 subtraction and accumulation to compute prediction errors.

Table 20: Prediction accuracy with full and low-frequency Hessian refresh on HamiBalls-1 over 48 edges; paired timing is described in Appendix E.4. Win rates use noise-mean pointwise errors against PhysiFormer.

Implementation	z MSE	q MSE	p MSE	Win (%)
Ours	0.04584945	0.01450559	0.07719332	64.4263
Ours (PLAS-S)	0.04584186	0.01450082	0.07718289	64.4173

PLAS-S reduces median sampling latency by about 6% in paired timing (Appendix E.4), with essentially unchanged aggregate prediction accuracy in this 48-edge HamiBalls-1 comparison. Relative to full refresh, total MSE changes by -0.01657% (source-cluster bootstrap 95% interval $[-0.04862\%, +0.01055\%]$) and win rate by -0.00895 percentage points. One of 512 sources exceeds a 1% MSE increase, with a worst increase of 2.55%.

E.4 Inference wall-clock

We compare prediction accuracy at similar measured inference cost on HamiBalls-1 using PhysiFormer with $N = 20$ and $N = 40$ RF steps and Ours with $N = 20$. PhysiFormer at $N = 40$ retains the original model weights and denominator floor of 0.05. Prediction MSE is evaluated on 512 sources and two noise realizations over 192 physical edges without resets.

We time complete 48-edge sampling calls on one NVIDIA A800 80GB PCIe GPU with PyTorch 2.5.1, CUDA 12.4, FP32, and batch size 64. CUDA is synchronized before and after each call. Measurements use warmed execution, excluding model loading, data I/O, compilation, and graph capture.

Table 21: Inference cost and prediction accuracy on HamiBalls-1, evaluated on 512 sources and two noise realizations. Latency is the median seconds per batch of 64 48-edge windows; normalized phase-space MSE covers all 192 predicted edges.

Method	RF steps N	Seconds / batch	192-edge MSE
PhysiFormer	20	0.402	0.34029
PhysiFormer	40	0.809	0.32767
Ours (PLAS)	20	0.786	0.25071

At comparable measured latency, Ours achieves 23.5% lower MSE than PhysiFormer with $N = 40$ (Table 21). Reducing PhysiFormer's denominator floor to $0.025 = 1/N$, so the final Euler step returns the clean prediction, gives a 192-edge MSE of 0.343529 under the same evaluation protocol. The accuracy advantage therefore persists when PhysiFormer receives a slightly larger wall-clock budget.

The sampler employs compiled derivative and mixed-scan kernels, batched Hamiltonian derivatives, and cached invariant computations. Ours in the main text uses full Hessian refresh; PLAS-S is an optional variant (Appendix E.3).

F Towards Scaling Up

F.1 Diagonal–low-rank affine propagation

PLAS-DLR uses diagonal–low-rank curvature to construct affine symplectic maps. Let $m = dn$ be the number of position coordinates and $2m$ the phase-space dimension. At anchor $\mathbf{a}$, it retains the

exact gradient and approximates the Hessian by

$$\widetilde{\mathbf{S}} = \mathrm{diag}(\mathbf{d}) + \mathbf{R}\mathbf{C}\mathbf{R}^\top, \qquad \mathbf{R} \in \mathbb{R}^{2m\times r}, \quad \mathbf{C} = \mathbf{C}^\top \in \mathbb{R}^{r\times r}. \tag{70}$$

The diagonal is estimated with s Rademacher Hessian–vector products (HVPs), averaging $\mathbf{v} \odot (\nabla^2 H(\mathbf{a})\mathbf{v})$. A randomized range approximation of the remaining curvature supplies $\mathbf{R}$; projection and symmetrization give $\mathbf{C}$. Probe directions are held fixed during inference. We use $r = 2$ and $s = 4$.

Write $\mathbf{R} = [\mathbf{R}_q^\top, \mathbf{R}_p^\top]^\top$. The implicit block becomes $\widetilde{\mathbf{B}} = \mathrm{I}_m + h\mathbf{R}_q\mathbf{C}\mathbf{R}_p^\top$. The Woodbury identity gives

$$\widetilde{\mathbf{B}}^{-1}\mathbf{v} = \mathbf{v} - h\mathbf{R}_q\mathbf{C}(\mathrm{I}_r + h\mathbf{R}_p^\top\mathbf{R}_q\mathbf{C})^{-1}\mathbf{R}_p^\top\mathbf{v}. \tag{71}$$

Thus propagation uses an $r \times r$ solve and tall-factor products. The symmetric quadratic Hamiltonian defined by $\widetilde{\mathbf{S}}$ has symmetric pure Hessian blocks, so Theorem 3.1's factorization preserves the symplecticity of the resulting affine maps whenever $\widetilde{\mathbf{B}}$ is invertible.

For a Hessian with Lipschitz constant L_H and curvature error $\left\|\widetilde{\mathbf{S}} - \nabla^2 H(\mathbf{a})\right\| \le \epsilon$, Taylor's theorem yields

$$\left\|\nabla H(\mathbf{z}) - \nabla H(\mathbf{a}) - \widetilde{\mathbf{S}}(\mathbf{z}-\mathbf{a})\right\| \le \epsilon\,\|\mathbf{z}-\mathbf{a}\| + \tfrac{1}{2}L_H\,\|\mathbf{z}-\mathbf{a}\|^2. \tag{72}$$

The generating-relation defect is multiplied by h; a uniformly stable implicit solve transfers this estimate to state error. Bounded curvature error and $O(h)$ anchor mismatch therefore give an $O(h^2)$ local contribution.

F.2 Computational and memory complexity

Let $C_{\mathrm{HVP}}(m)$ denote the cost of one HVP. Dense Hessian construction requires $O(mC_{\mathrm{HVP}}(m))$ work, followed by $O(m^3)$ dense factorization and $O(m^2)$ affine application. PLAS-DLR construction costs

$$O\big((s+2r)C_{\mathrm{HVP}}(m) + mr^2 + r^3\big), \qquad \text{with application cost } O(mr + r^2). \tag{73}$$

For batch size b and window length L, stored jets require $O(bLm^2)$ space for PLAS and $O(bL(mr + r^2))$ for PLAS-DLR. The packed DLR representation stores $3mr + 6m + r^2$ scalars per anchor, compared with $4m^2 + 2m$ for a dense affine map. At $d = 2, n = 100, r = 2$, these are 2,404 and 160,400 scalars. With fixed width, rank, and probe count, full-attention HVPs have quadratic object-count work, giving a subcubic construction route. Total inference memory also includes attention, automatic-differentiation intermediates, and model buffers; its measured scaling is shown below.

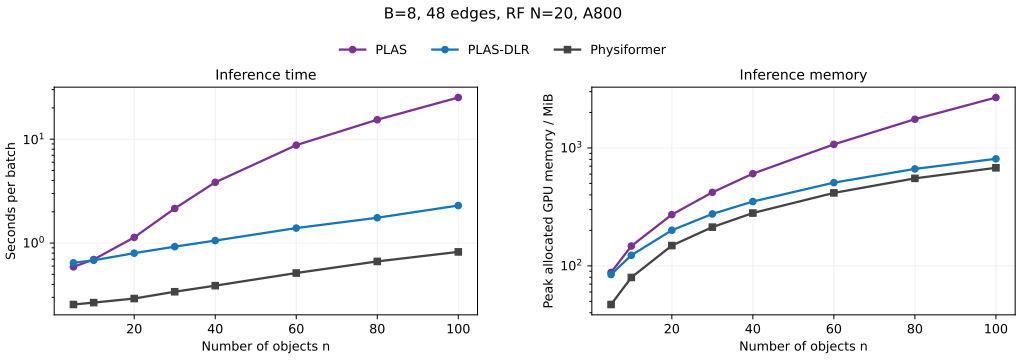


Figure 6: Object-count scaling on an A800. Left: median inference time after three warm-up calls, using five timed calls. Right: peak allocated GPU memory after warm-up, measured in separate processes for each method and size. Both vertical axes are logarithmic. Batch size is 8, window length is 48, and RF uses 20 steps.

We extend the two-dimensional token interface to $n \in \{5, 10, 20, 30, 40, 60, 80, 100\}$ using replicated physical attributes and Gaussian noisy trajectories. Figure 6 measures computational scaling with the

same tensor modules. At $n = 100$, PLAS-DLR takes 2.297 seconds per batch, compared with 0.821 seconds for PhysiFormer. Their inference allocation peaks are 808.73 and 678.39 MiB, respectively, compared with 2,678.30 MiB for PLAS. Allocated memory includes model tensors, inputs, outputs, and live computation-graph buffers; allocator reservations and CUDA context are separate. Hessian construction is batched to fit the working-memory budget.

F.3 HamiBalls-1 accuracy

We evaluate 64 physical sources with two noise realizations over 48 edges. PLAS-DLR obtains normalized q-MSE 0.0144733, p-MSE 0.0772124, and z-MSE 0.0458429. The corresponding PLAS values are 0.0144720, 0.0772396, and 0.0458558. Together with the runtime and memory curves, these results support diagonal–low-rank curvature as a practical route to larger systems while retaining trajectory accuracy on HamiBalls-1.

G Experimental Setup and Additional Results

G.1 Dataset generation

Both datasets contain independently simulated mechanical episodes with fixed object identities and observable physical attributes. Each episode spans 192 edges (193 saved states), with a saved-frame interval of $h = 1/30$ and total duration 6.4. Table 22 summarizes the physical settings and the train/validation portions used in this work. HamiBalls-2 balances object counts across $n = 5, \ldots, 10$, with equal training counts at each n.

Table 22: Dataset generation settings. $\mathcal{U}[a, b]$ denotes uniform sampling; speed ranges apply before the HamiBalls-1 center-of-mass adjustment. Restitution entries are per-body material coefficients.

Setting	HamiBalls-1	HamiBalls-2
Domain	$[-1, 1]^2$	$[-1, 1]^2 \times [0, 2]$
Objects	5 disks	5–10 spheres
Mass; radius	$\mathcal{U}[0.5, 1.5]$; $\mathcal{U}[0.06, 0.10]$	Same ranges
Initial speed	$\mathcal{U}[0.35, 0.90]$	$\mathcal{U}[0.35, 0.90]$
Body restitution	$\mathcal{U}[0.4, 0.9]$	$\mathcal{U}[0.90, 0.98]$
Boundary restitution	0.8	1.0
Gravity magnitude	0	1 (downward)
Springs	Central harmonic, $k = 0.5$	Sparse pairwise Hookean
Simulator	Pymunk 7.3.0 / Chipmunk2D	PyBullet 3.2.7
Substeps per saved edge	8	32
Contact solver iterations	20	50
Train / validation episodes	14,336 / 512	30,720 / 512

Initial states and spring connections. Mass, radius, and restitution are sampled independently. Centers are sampled uniformly within the admissible box, rejecting initial overlaps with a clearance of 0.015. Initial velocity directions are isotropic; HamiBalls-1 additionally subtracts the mass-weighted center-of-mass velocity. Its central force is $\mathbf{F}_i = -0.5\mathbf{q}_i$. In HamiBalls-2, each unordered pair is connected with probability $2/(n-1)$, conditioned on at least one spring; connectivity is otherwise unrestricted. For each spring, sample a period T_{ij} log-uniformly from $[\sqrt{0.5}, \sqrt{8}]$ and set $k_{ij} = \mu_{ij}(2\pi/T_{ij})^2$, where $\mu_{ij} = m_i m_j/(m_i + m_j)$. Its rest length is the initial separation multiplied by $\exp(u)$, with $u \sim \mathcal{U}[-0.15, 0.15]$.

Simulation and state representation. HamiBalls-1 applies a symplectic-Euler force kick before each Pymunk drift/contact step. HamiBalls-2 recomputes spring forces at every PyBullet substep and advances under gravity and contact impulses. Both use frictionless contacts and no velocity damping; the engines combine colliding bodies' restitution coefficients multiplicatively. PyBullet uses a restitution velocity threshold of 0.2. Episodes are simulated continuously across the full horizon. Saved states contain position and canonical momentum $\mathbf{p}_i = m_i\mathbf{v}_i$, together with mass,

radius, restitution, and the HamiBalls-2 spring graph, stiffnesses, and rest lengths. Variable-size scenes use masked padding. Fixed normalization scales are computed from the training split. Training uses 48-edge windows, while the reported 192-edge evaluation begins at episode initialization, as detailed below.

G.2 Aggregation and evidence scope

Under the protocol in Section 4, all component MSEs use fixed normalized phase-space coordinates. For an object–edge stratum $\mathcal{S}$, component dimension d_c ($d_z = 4$, $d_p = d_q = 2$ for HamiBalls-1; $d_z = 6$, $d_p = d_q = 3$ for HamiBalls-2), and two noise realizations, the absolute metric is

$$\mathrm{MSE}_{c,\mathcal{S}} = \frac{1}{2|\mathcal{S}|d_c} \sum_{(b,k,i)\in\mathcal{S}} \sum_{\omega=1}^{2} \left\| \widehat{\mathbf{x}}^{(\omega)}_{bki,c} - \mathbf{x}^{*}_{bki,c} \right\|_2^2 . \tag{74}$$

Tables 1 and 2 use this reduction over their respective intervals. HG-DPF uses matched evaluation with SNIS guidance; deterministic Transformer-AR repeats each source's prediction across both noise labels for aggregation.

For fixed trained models, 95% percentile CIs for the 192-edge phase-space MSE reductions are $[23.8\%, 28.8\%]$ over PhysiFormer on HamiBalls-1 and $[20.7\%, 22.1\%]$ over DiT on HamiBalls-2. We use 10,000 paired bootstrap resamples of 512 sources, retaining both noises and all objects and edges per source and recomputing pooled MSE from summed squared errors and valid coordinate counts.

Contact marks object–edge cells with positive normal contact impulse in any substep, including sustained contact; all others are continuous, including objects away from simultaneous contacts. Both datasets use force threshold 10^{-12}, dividing impulse by substep duration on HamiBalls-1. Its $512 \cdot 192 \cdot 5 = 491{,}520$ cells comprise 7,829 contact and 483,691 continuous cells. Contact-mask reconstruction from original scene seeds and accepted sampling attempts reproduced all stored ground-truth trajectories bitwise. Both noises are aggregated before cell/source/track win statistics.

G.3 Autoregressive baselines on HamiBalls-2

Deterministic Transformer-AR inputs are normalized positions/momenta, observable attributes, spring graph/parameters, and physical time increment. Object tokens are linearly embedded into four pre-layer-normalized Transformer blocks (Vaswani et al., 2017): width 340, four heads, width-1360 GELU feed-forward layers, zero dropout. Graph summaries augment object features; spring features provide learned pairwise attention bias. Final layer normalization and a shared linear head predict six-component state increments; padding masks exclude absent objects.

Context-one uses the current state; context-four concatenates the latest four chronologically, zero-padding missing states with history-validity indicators (5,574,318/5,581,798 parameters). Training uses all 48 teacher-forced transitions per window, including shorter initial histories, averaging normalized squared errors over valid objects, edges, and coordinates.

AdamW (Loshchilov & Hutter, 2019) runs for 50,000 updates with 64-window batches, 1,000-update warmup to 3×10^{-4}, cosine decay to 3×10^{-7}, weight decay 10^{-4}, and norm-1 gradient clipping. BF16 autocast retains float32 parameters/loss; evaluation uses final weights.

Inference rolls out 192 edges from the observed initial frame, retaining up to four states for context-four.

Using available ground-truth predecessors at every edge over all 512 sources and 192 edges gives single-step MSEs of 0.004242/0.003419 for context-four/one. Supplying three ground-truth transitions before free rollout leaves a 192-edge MSE gap (1.478274/0.883231), so cold start alone does not explain the difference.

G.4 Per-seed errors and continuation chunks

Table 24 separates the four windows; Table 2 pools the last two. Equation (74) uses each interval's valid coordinate and contact/continuous cell counts.

Table 23: HamiBalls-1 absolute 48-edge normalized MSE by physical data-generation seed.

	PhysiFormer			Ours		
Physical seed	z	q	p	z	q	p
40	0.06718	0.02263	0.11173	0.04557	0.01520	0.07594
41	0.06582	0.02046	0.11118	0.05146	0.01686	0.08606
42	0.06718	0.02420	0.11016	0.04431	0.01550	0.07311
43	0.05907	0.01574	0.10240	0.04206	0.01045	0.07366

Table 24: HamiBalls-1 absolute normalized MSE for disjoint 48-edge continuation chunks.

	PhysiFormer			Ours		
Physical edges	z	q	p	z	q	p
1–48	0.06481	0.02076	0.10887	0.04585	0.01451	0.07719
49–96	0.26704	0.23468	0.29940	0.18787	0.16078	0.21496
97–144	0.46841	0.49981	0.43702	0.33972	0.36346	0.31598
145–192	0.56089	0.63230	0.48947	0.42938	0.48507	0.37370

G.5 Comparison with the internal diffusion expert

In Section 4.5's 512-source, paired-noise comparison, internal D-only/Ours MSEs are 0.066289/0.045850 (48 edges) and 0.338632/0.250707 (192 edges): reductions of 30.83%/25.96%. With noises averaged per source, Ours improves 82.62%/78.91% of trajectories, respectively.

G.6 Model-tree and objective ablation: controls and diagnostics

Table 5's five variants (D: Ours) share frozen base models, sources, normalization, RF evaluation ($N = 20$), batch size 64, and 800 downstream neural updates. Each uses its own training carriers; tree construction, ridge fitting, and calibration add computation.

A/B supervise residuals with mean normalized q/p squared errors on teacher-forced local and mixed autoregressive states, and routers with mixed-prediction squared errors on unrolled histories. The full objective adds robust/recovery and hull residual terms and preference, regret-reweighting, and no-harm router terms. Ridge regularization, features, cumulative fitting, and training-side calibration follow the full configuration. B/D share tree-construction rules, with five/seven reachable leaves fixed during final training; A/C retain shared final q/p heads. GL uses global residual/router readouts with the same features and calibration. Combined neural-parameter/ridge-coefficient counts are 1,176,220 (A/C), 1,178,428 (B/D), and 1,176,496 (GL), with 1,056 (B/D) and 132 (GL) ridge coefficients.

Difficulty stratification. Quartiles rank source–edge–object cells by two-noise-mean phase-space error of Ours' uncorrected H candidate on its mixed history. From easiest to hardest, Ours reduces total prediction error relative to PhysiFormer by 93.13%, 96.21%, 87.16%, and 20.66%. Momentum correction reduces direct candidate SSE by 11.61% in the hardest quartile and increases it in easier quartiles with small uncorrected errors.

Training-side regional readouts. Ours' seven reachable leaves occupy 3.87%, 1.86%, 2.20%, 2.42%, 8.75%, 8.57%, and 72.34% of readout-fitting rows. Regional momentum readouts reduce direct candidate SSE by 12.95%, versus GL's 6.10%, complementing the evaluation in Table 5. Tree inputs exclude event labels and source identifiers.