BEYOND JOINT REFINEMENT: BOUNDED RESIDUAL FLOWS FOR MULTI-MODAL TRAJECTORY FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-future trajectory prediction is a safety-critical bottleneck between perception and planning: an over-confident wrong-mode prediction can steer a planner into an unsafe manoeuvre. The dominant paradigm appends a refinement module to a strong backbone and trains the two *jointly* from random initialisation – leaving the refined output without an a-priori bound and letting the refiner lean on a learned gate that masks a poorly-behaved refiner on clean data and fails wherever its protection lapses: out-of-distribution input, closed-loop recursion, or low-bit quantisation. We therefore argue that a refiner should be judged by whether its raw correction is valid *without* the gate, and we construct one that is. BReF (*Bounded Residual Flow*) is a plug-and-play module that attaches to any frozen multi-modal predictor and refines its trajectories by integrating a learned drift field along a latent "refinement" time, kept separate from the physical trajectory time. A smoothness penalty in this latent time turns the refiner into a contraction, so its raw correction stays small and physically plausible on its own and the gate becomes near-redundant. Because the refiner is identity-initialised on the frozen backbone, the backbone acts as a *verifiable performance floor* that jointly-trained refiners cannot give. On nuScenes, BReF improves a strong backbone among twenty recent baselines on every multi-mode metric and on single-mode minADE while preserving the full mode set; the decisive gate-forced diagnostic – removing the gate at inference – separates refiners that are indistinguishable on displacement error by nearly three orders of magnitude, tracing the advantage to the latent-time contraction.

1 INTRODUCTION

Trajectory prediction is the safety-critical bottleneck between perception and planning in autonomous driving: from a short agent history plus a vectorised scene description, a predictor must emit a small set of plausible futures with calibrated probabilities, on which a downstream planner conditions its decisions. Prediction errors are asymmetric – a single over-confident wrong-mode trajectory can drive the planner into an unsafe manoeuvre, at a cost incomparably larger than that of a conservative prediction – and it is this asymmetry that makes architectural choices inside the predictor consequential.

Modern predictors share a two-stage design: a backbone encodes the scene and emits candidate trajectories, and a refinement module re-attends to the scene and updates each one (Shi et al., 2022; 2024; Zhou et al., 2024; Zhang et al., 2024; Seff et al., 2023; Pei et al., 2025; Rao et al., 2026). Refiners are almost uniformly co-optimised with the backbone from random initialisation, so the structural failure mode is shared across the literature: the first-iteration output bears no a-priori relation to the backbone's, early training is unbounded, and nothing keeps the trained refiner inside the backbone's basin of attraction – weaknesses unreachable by warm-up or learning-rate schedules, which act only after the damaging first steps.

A subtler observation drives our design. In a gated refiner the prediction adds to the backbone's output a raw residual scaled element-wise by a learned gate. After training the gate converges to a small average value, so *every* refiner family – an MLP, a Neural-ODE, or an ODE-with-physical-loss variant – collapses to within a millimetre of the others on the default metric even though their *raw* residuals differ by hundreds of millimetres: the gate masks whatever the refiner produces, so a low

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.

displacement error certifies the gate, not the refiner. The distinction is invisible to the metric yet decisive in deployment, where the gate's protection lapses. Our goal is therefore not a better gate but a refiner whose *raw residual is valid by construction*, reducing the gate from a load-bearing component to a near-redundant safeguard.

BReF (*Bounded Residual Flow*) realises this. A strong backbone is trained once and frozen; the refiner is an outer *Neural-ODE* that integrates a learned drift field over a latent "refinement" time, kept separate from the physical trajectory time, regularised by a single smoothness penalty on that field. That penalty is what renders the gate near-redundant: bounding how fast the field drifts per integration step makes the refiner a Lipschitz contraction, so the integrated raw correction stays small and physically plausible *on its own* and the gate can be removed at inference at almost no cost. Initialising the refiner to the identity adds a second guarantee: the refined prediction coincides with the backbone at the first optimiser step and provably stays within its sublevel set under small updates – a floor it can only raise. Because it attaches through a generic trajectory-residual interface, BReF is plug-and-play on any pre-trained predictor, and its bounded per-mode correction is a *strict, coverage-preserving* improvement: it sharpens the top-confidence trajectory and every alternative mode without destroying the spread the backbone discovered.

We make the following four contributions.

- **A diagnostic for raw-correction validity.** The *gate-forced* diagnostic judges a refiner by whether its raw correction is valid without the gate, separating refiners that tie on displacement error by nearly three orders of magnitude (Section 4.3).
- **A floor-preserving guarantee.** Decoupling from the frozen backbone and identity-initialising make the backbone a verifiable floor that refinement can only raise – a downside guarantee joint training cannot offer (Proposition 1).
- **A contraction from latent-time smoothness.** BReF is a plug-and-play outer Neural-ODE whose smoothness penalty in *latent* – not output – time makes the refiner a Lipschitz contraction, so its raw correction is valid by construction and the gate near-redundant (Sections 3.2–3.3).
- **State-of-the-art accuracy, portability, and robustness.** On nuScenes BReF improves a strong backbone among twenty baselines on every multi-mode metric and single-mode minADE while preserving every mode, improves or ties nineteen further frozen predictors, and generalises to unseen geographies and manoeuvres (Section 4).

2 Related Work

We position BReF against three lines of work; the extended per-family discussion, including all twenty reproduced baselines and the humanoid-motion outlook, is deferred to Appendix A.

Vector trajectory predictors and output refinement. Modern multi-modal forecasting fuses *vector* representations of agents and HD maps (Gao et al., 2020; Zhou et al., 2022; 2023; Nayakanti et al., 2022; Xu et al., 2023; Cheng et al., 2023; Wang et al., 2023a) with explicit HD-map *priors* (Zhao et al., 2021; Gu et al., 2021; Deo et al., 2022; Gilles et al., 2021; 2022; Yadav et al., 2025; 2024); diffusion, language-model and IRL formulations (Jiang et al., 2023b; Gu et al., 2022; Seff et al., 2023; Pei et al., 2025; Sun et al., 2024; Hagedorn et al., 2025) complete the design space; LAformer (Liu et al., 2024) fuses the two axes via a dense per-step lane-likelihood map, and we adopt its encoder as the frozen backbone (Appendix B.3). Closest to BReF is the line that refines *already-produced* trajectories: MTR (Shi et al., 2022)/MTR++ (Shi et al., 2024) queries, SmartRefine (Zhou et al., 2024), MultiPath++ (Varadarajan et al., 2022), DeMo (Zhang et al., 2024) – *all trained jointly with the backbone*, so the initialisation is random and the output can diverge arbitrarily from the base trajectories. BReF instead appends an *identity-initialised*, *contractive* Neural-ODE refiner *after* a frozen backbone: it reproduces the backbone at the first optimiser step (Proposition 1), its raw correction is valid by construction rather than gate-masked, and it adds only 576 K parameters ($\sim 13\times$ smaller); the gate-forced diagnostic (Section 4.3) separates them from BReF by nearly three orders of magnitude.

Parameter-efficient adaptation of frozen backbones. BReF's gated additive structure combines residual learning (He et al., 2016) with gated control (Srivastava et al., 2015; Hochreiter & Schmidhuber, 1997) and echoes iterative decoder refinement (Carion et al., 2020; Zhu et al., 2020) – but attached to an *independently optimised* predictor, so an identity-initialised gate yields a verifiable

floor. The frozen-backbone-plus-trainable-head pattern (Houlsby et al., 2019; Hu et al., 2021; Zhang et al., 2023b; Lester et al., 2021; Jia et al., 2022) transfers across *tasks*; BReF instead relaxes the backbone's suboptimality gap on the same task – closer to born-again self-distillation (Furlanello et al., 2018; Hinton et al., 2015).

Continuous latent dynamics, physics supervision, optimisation geometry. Continuous-time decoders (Chen et al., 2018; Rubanova et al., 2019; Lipman et al., 2022; Tong et al., 2023; Albergo et al., 2025; Biloš et al., 2021; Perez et al., 2018), physics-informed supervision (Raissi et al., 2019; Zhao et al., 2021; Salzmann et al., 2020), and optimisation geometry (Polyak & Juditsky, 1992; Tarvainen & Valpola, 2017; Izmailov et al., 2018; Foret et al., 2020; Sun et al., 2020; Loshchilov & Hutter, 2016) all inform the construction; we adopt the simplest combination (EMA with cosine annealing), the stronger alternatives being redundant or regressive once the base model is strong.

3 METHOD

Problem and notation. We address multi-modal trajectory forecasting on the nuScenes prediction benchmark (Caesar et al., 2020). For each scene, the predictor consumes the target agent's last T_p=20 steps ($\mathbf{x}^{\text{past}} \in \mathbb{R}^{T_p \times 2}$), the histories of N_a surrounding agents, and a vectorised HD map $\mathscr{M}$, all in a canonical frame; it must output K=5 candidate futures $\hat{\mathbf{y}}_{1:K} \in \mathbb{R}^{K \times T_f \times 2}$ over T_f=12 steps (~ 6 s at 2 Hz) with probabilities $\hat{\pi}_{1:K} \geq 0$, $\sum_k \hat{\pi}_k$=1. Following the nuScenes evaluation protocol (Caesar et al., 2020), performance is measured by minimum average displacement error (minADE$_K$), minimum final displacement error (minFDE$_K$), miss rate (MR$_K$), and the Brier-augmented variants brier-minADE$_K$ and brier-minFDE$_K$; vectors are bold lowercase ($\boldsymbol{\mu}$, $\boldsymbol{\delta}$, $\boldsymbol{\sigma}$) and indices over batch / mode / future-step are b / k / t.

Architecture overview. BReF composes two components (Figure 1): a frozen lane-aware backbone f_{bb} (Section 3.1) and a gated residual refiner g_{ref} (Section 3.2). Given inputs $(\mathbf{x}^{\text{past}}, \mathbf{x}^{\text{ctx}}, \mathscr{M})$ – $\mathbf{x}^{\text{ctx}}$ the concatenated histories of surrounding agents – the predictor produces the K canonical-frame trajectories by

$$\hat{\mathbf{y}}_{1:K} = \underbrace{f_{\text{bb}}(\mathbf{x}^{\text{past}}, \mathbf{x}^{\text{ctx}}, \mathscr{M})}_{\text{frozen } \boldsymbol{\mu}^{\text{final}}} + \underbrace{\sigma(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}}_{\text{gated residual}}, \tag{1}$$

where $(\boldsymbol{\delta}, \boldsymbol{\gamma}) = g_{\text{ref}}\big(\boldsymbol{\mu}^{\text{final}}, \mathbf{H}_{\text{scene}}\big)$ is the per-mode, per-step residual and gate logit produced by the refiner from the backbone's canonical-frame trajectories $\boldsymbol{\mu}^{\text{final}} \in \mathbb{R}^{B \times K \times T_f \times 2}$ and the polyline-level scene hidden state $\mathbf{H}_{\text{scene}} \in \mathbb{R}^{B \times N \times d}$; $\sigma(\cdot)$ is a (hard) sigmoid and $\odot$ the Hadamard product. The additive correction $\sigma(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}$ is the *bounded residual flow* of our title: a magnitude-bounded, gate-controlled perturbation of the frozen backbone whose worst-case amplitude is governed jointly by the residual head and the gate.

3.1 FROZEN LANE-AWARE BACKBONE

The backbone is a strong lane-aware multi-modal predictor combining the published LAformer family (Liu et al., 2024) (scene encoder, dense per-step lane-likelihood scoring, multi-modal decoding head) with an *internal latent-ODE refinement* – a continuous-time, Flow-Matching-regularised module LAformer does *not* contain – of our own design. We train the combined backbone once ($\sim$7.44M parameters) and freeze its EMA snapshot, so all of BReF's trainable parameters reside in the outer refiner of Section 3.2; the contributions *claimed in this paper* concern that outer refiner. The full architecture, auxiliary regularisers, composite training loss, and freezing protocol are specified in Appendix B.

3.2 GATED RESIDUAL REFINER

The outer refiner consumes the frozen backbone's K=5 trajectories $\boldsymbol{\mu}^{\text{final}}$ and scene context $\mathbf{H}_{\text{scene}}$, and returns a raw residual $\boldsymbol{\delta}$ and gate logits $\boldsymbol{\gamma}$. These separate heads enable the gate-forced diagnostic (Section 4.3.1) and identity initialisation (Section 3.4). Unlike the backbone's internal latent-ODE (Appendix B.3), the outer flow acts directly in trajectory space.

Drift network. At the core of the refiner is a drift network $f_\theta(\tau, \mathbf{x}, \mathbf{H}_{\text{scene}})$ mapping the current K-mode trajectory $\mathbf{x} \in \mathbb{R}^{K \times T_f \times 2}$ to a velocity $\mathrm{d}\mathbf{x}/\mathrm{d}\tau$ in trajectory space. It embeds each per-step

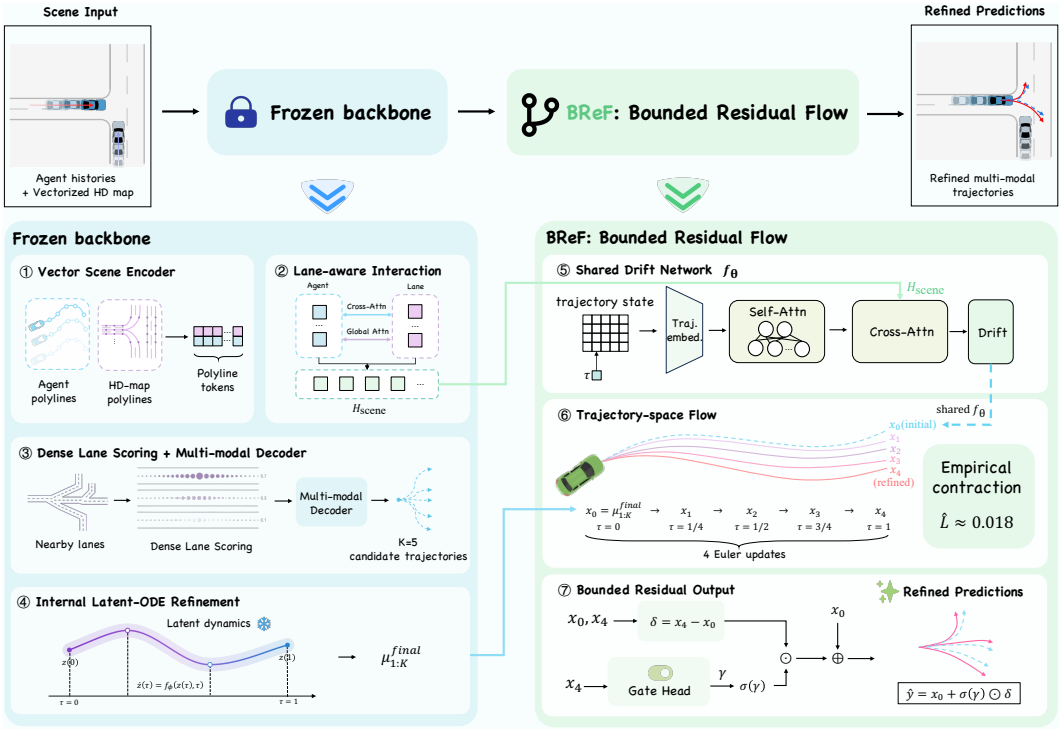


Figure 1: **BReF architecture. Left:** the lane-aware backbone (7.44M parameters) encodes agent histories and the vectorised HD map, decodes K=5 candidate trajectories, and applies internal latent-ODE refinement. It is trained once and frozen. **Right:** the outer BReF refiner (576K trainable parameters) uses a shared, scene-conditioned drift network to update these trajectories over latent refinement time τ with four Euler steps. The resulting raw residual $\boldsymbol{\delta} = \mathbf{x}_4 - \mathbf{x}_0$ is gated and added to the frozen prediction. Latent-time drift regularisation yields an empirically estimated contraction factor $\hat{L} \approx 0.018$; zero-initialising the drift head ensures that the refined prediction exactly matches the backbone at initialisation (Proposition 1).

displacement to width d_r=128, adds a learned positional code and a sinusoidal embedding of τ, processes the result with two pre-LayerNorm self-attention blocks over the time axis and two cross-attention blocks to the projected scene context (four heads, FFN width 256, validity masking), and reads out the drift through a zero-initialised linear head.

Continuous-depth integration. Starting from the backbone prediction $\mathbf{x}(0)$=$\boldsymbol{\mu}^{\text{final}}$, the refiner integrates the drift over a fictitious refinement time $\tau \in [0, 1]$,

$$\mathbf{x}(1) = \mathbf{x}(0) + \int_0^1 f_\theta\big(\tau, \mathbf{x}(\tau), \mathbf{H}_{\text{scene}}\big)\, \mathrm{d}\tau, \qquad \boldsymbol{\delta} = \mathbf{x}(1) - \mathbf{x}(0), \tag{2}$$

with a fixed-step explicit solver (Euler, K_{int}=4 steps, $\Delta\tau$=$1/K_{\text{int}}$, by default; because f_θ defines a genuine continuous-time field, the integrator and step count are inference-time choices, Section 4.3.3). The raw residual $\boldsymbol{\delta}$ is the net displacement of this flow.

Matched-MLP control. The parameter-matched baseline used throughout our experiments applies the *same* drift network *once* – $\boldsymbol{\delta}$=$f_\theta(\mathbf{x}(0))$, with no τ-conditioning and no integration – so the sole difference between BReF and the control is the continuous-depth flow, which isolates its effect.

Gate and identity initialisation. A second linear head on $\mathbf{x}(1)$ produces the per-step gate logit $\boldsymbol{\gamma}$, and the prediction is the gated residual of Equation (1), $\hat{\mathbf{y}} = \boldsymbol{\mu}^{\text{final}} + \sigma(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}$ (σ a hardsigmoid). The residual head is *zero-initialised* and the gate-logit bias set to -2.5, so $\boldsymbol{\delta}_{\text{init}} \equiv \mathbf{0}$ and the prediction coincides element-wise with the backbone at the first optimiser step (Section 3.4).

In total the BReF refiner has 575,747 trainable parameters – a factor $\sim 13\times$ smaller than the 7.44M-parameter backbone – of which the shared attention core accounts for 540,803, the continuous-depth additions (sinusoidal τ-conditioning MLP and gate embedding) for the rest (per-module budget in Appendix G.4, Fig. 10a).

3.3 Smoothness in Latent Time Induces a Contraction

The single property that separates BReF from a generic ODE refiner is a smoothness penalty on the drift along the *latent* integration variable τ. We penalise the mean squared drift magnitude – equivalently, the change of state per integration step –

$$\mathscr{L}_{\text{smooth}} = \mathbb{E}\big[\,\|\mathrm{d}\mathbf{x}/\mathrm{d}\tau\|^2\,\big] = \frac{1}{K_{\text{int}}}\sum_{i=0}^{K_{\text{int}}-1}\big\|f_\theta\big(\tau_i,\mathbf{x}(\tau_i)\big)\big\|^2, \tag{3}$$

which by $\mathbf{x}(\tau_{i+1})-\mathbf{x}(\tau_i)=\Delta\tau\, f_\theta(\tau_i,\cdot)$ is the squared step length of the flow up to $\Delta\tau^2$. Bounding the drift bounds the latent Jacobian $\partial f_\theta/\partial\mathbf{x}$, making the flow a *contraction*: its top singular value (a 25-step power-iteration Jacobian probe, Appendix F.5.2) is $\hat{L}\approx 0.018 \ll 1$ on every batch, against $\hat{L}\approx 1.0$ for the matched MLP. A contraction integrates to a small, physically plausible residual with no gate required – the mechanism behind both the gate-removability and the out-of-distribution robustness of BReF. The penalty must act in *latent* time: on the output time t it shapes the trajectory but leaves the drift field unconstrained, fails to induce the contraction, and leaves the gate load-bearing – verified in Section 4.3, isolated by the controls of Sections 4.3.2–4.3.3, and confirmed by Section 4.3.3 and Appendices F.4.1–F.4.2.

3.4 Theoretical Properties: Floor Preservation

Freezing the backbone and identity-initialising the refiner preserve the backbone prediction at initialisation, as formalised below; training dynamics and transfer across nineteen backbones are evaluated in Appendix F.7 and Section 4.2.

Proposition 1 (Identity Initialisation)**.** *Let $f_{bb}(\cdot;\theta_{bb})$ be a backbone whose parameters θ_{bb} are held fixed, and let $g_{ref}(\cdot;\phi)$ denote the gated residual refiner of Section 3.2 with trainable parameters ϕ. Initialise the refiner heads as follows: the residual head $\boldsymbol{\delta} = W_\delta\mathbf{h}+b_\delta$ has $W_\delta=\mathbf{0}$ and $b_\delta=\mathbf{0}$; the gate-logit head $\boldsymbol{\gamma}=W_\gamma\mathbf{h}+b_\gamma$ has $W_\gamma=\mathbf{0}$ and $b_\gamma=-2.5\cdot\mathbf{1}$. Let ϕ_0 denote the resulting parameter configuration (the remaining refiner weights may be initialised arbitrarily, e.g. by Xavier or Kaiming). Then, for every admissible input $(\mathbf{x}^{past},\mathbf{x}^{ctx},\mathscr{M})$,*

$$\big[\,f_{bb}+\sigma(\boldsymbol{\gamma})\odot\boldsymbol{\delta}\,\big]\big|_{\phi=\phi_0} \;=\; f_{bb}(\mathbf{x}^{past},\mathbf{x}^{ctx},\mathscr{M}), \tag{4}$$

i.e. the refined predictor coincides element-wise with the frozen backbone at initialisation.

Corollary 1 (No Worst-Case Regression at Initialisation)**.** *At the first optimiser step ($\phi=\phi_0$), BReF attains identical values of minADE$_K$, minFDE$_K$, MR$_K$, brier-minADE$_K$, and brier-minFDE$_K$ to the frozen backbone f_{bb}, on any input distribution.*

Both follow from $\boldsymbol{\delta}|_{\phi_0}\equiv\mathbf{0}$ (the Hadamard product with the zero tensor vanishes regardless of $\boldsymbol{\gamma}$); proofs and training-dynamics consequences are in Appendix C. The key consequence: with $\mathscr{L}_{\text{refiner}}(\phi_0)=\mathscr{L}^{\text{bb}}$ (the backbone's loss), any sufficiently small first-order update keeps the trajectory inside the sublevel set $\{\phi:\mathscr{L}_{\text{refiner}}(\phi)\le\mathscr{L}^{\text{bb}}\}$ – the entire trajectory, not merely its endpoint, is bounded below by the backbone. A randomly-initialised end-to-end refiner starts far outside this basin with no bound on the cost of finding the way back.

3.5 Training the Refiner

The backbone parameters are loaded from the frozen EMA snapshot, held in evaluation mode inside a no-gradient context. Writing the refined trajectories of Equation (1) as $\hat{\boldsymbol{\mu}}_{k,t}\triangleq\boldsymbol{\mu}^{\text{final}}_{k,t}+\sigma(\boldsymbol{\gamma}_{k,t})\,\boldsymbol{\delta}_{k,t}$, the per-example loss is the best-mode Laplace negative log-likelihood evaluated on $\hat{\boldsymbol{\mu}}$, using the backbone's frozen scale parameters $\boldsymbol{\sigma}$:

$$\begin{aligned}\mathscr{L}_{\text{refiner}} = \;& -\log p_{\text{Lap}}\big(\mathbf{y}^\star\,\big|\,\hat{\boldsymbol{\mu}}_{k^\star},\boldsymbol{\sigma}_{k^\star}\big)\\ &+\;\lambda_g\,\mathbb{E}\big[\sigma(\boldsymbol{\gamma})^2\big]\;+\;w_{\text{smooth}}\,\mathscr{L}_{\text{smooth}},\end{aligned} \tag{5}$$

where $k^\star$ is the backbone mode closest to the ground truth, fixing mode assignment independently of the refiner (best-of-K supervision (Varadarajan et al., 2022; Zhao et al., 2021)). The Laplace term supplies the accuracy signal; gate regularisation ($\lambda_g{=}10^{-3}$) damps early gating; latent-time smoothness ($w_{\text{smooth}}{=}10^{-2}$) shapes the drift field. The first two terms are shared by all refiner families; only smoothness changes gate-forced robustness (Section 4). Optimisation settings are in Appendix G, and Algorithm 2 (Appendix G.3) gives the forward pass and training step.

4 Experiments

We evaluate accuracy, portability, and gate-forced robustness under a shared protocol. The matched control is the prior best-accuracy *discrete* refiner on this backbone. All numbers are server-verified; extended tables, visuals, and analyses are in Appendices E–F.

Benchmark protocol. The reproduced panel spans seven years and every major design family (full listing with venues in Appendix F.1). All baselines are re-trained for 30 epochs at global batch 16 with Adam and cosine annealing, and evaluated on the standard nuScenes val split (9,041 samples, 6 s horizon) with minADE_K, minFDE_K, MR_K for $K{\in}\{1,5\}$: $K{=}1$ top-confidence without oracle, $K{=}5$ oracle closest-of-five. Loss weights, optimiser settings, and hardware are in Appendix G.

4.1 Main Quantitative Comparison

Table 1 compares BReF against the twenty reproduced baselines under both the single-mode ($K{=}1$, no oracle) and multi-mode oracle ($K{=}5$) protocols on the same val split. BReF achieves the best score on four of the five multi-mode columns – minADE_5, minFDE_5, brier-mADE_5, brier-mFDE_5 – and on single-mode minADE_1, improving its backbone on the fifth (MR_5, where AutoBots attains the lowest miss rate); on minFDE_1 it preserves the backbone's endpoint error (convention-dependent; see footnote).

Observations. *First*, the strongest competitor is LAformer (Liu et al., 2024), whose encoder we adopt as the frozen backbone; our 576 K refiner pushes the panel floor another $\sim 2\%$ (1.213→1.185 minADE_5, 3.43→3.35 minADE_1). *Second*, jointly-trained refinement predictors cluster mid-table and 2025/2026 predictors span $\text{minADE}_5 \in [1.38, 3.76]$: architectural complexity does not monotonically translate into prediction quality.

Why the headline gap is small – and why it does not mean the refiner choice is unimportant. The improvement of BReF over LAformer on minADE_5 is only ~ 0.03 m, and the *default* minADE of any two refiner families in Table 3a lies within ~ 2 mm. This near-equality is a *direct consequence of the bounded gate*: the trained gate saturates at $\overline{\sigma(\gamma)} \approx 0.143$ (Appendix C.2), suppressing the raw residual by $\sim 7\times$ – *all four families land within a few millimetres on default minADE_5 even when their raw residuals differ by orders of magnitude.* The headline gap therefore establishes only that accuracy is not sacrificed; the difference between refiner families becomes observable only when the gate is *forced open*.

4.2 Plug-and-Play Generality: Refining Frozen Off-the-Shelf Predictors

A generic adapter reads a frozen predictor's trajectories, builds its own scene encoding, and attaches the refiner and gate of Section 3; only the adapter and refiner train. Across nineteen public predictors (2019–2026), with no per-backbone tuning, refinement improves or ties all but one backbone (Table 2; Fig. 3(b)). The sole regression is 2 mm; gains diminish on stronger predictors, consistent with limited correction headroom. Changing the gate initialisation leaves the overall pattern unchanged (Appendix F.8).

4.3 ODE-Refiner Inductive Bias: Stress-Test Suite

Refiners that tie BReF on accuracy do *not* tie it on robustness – where the diagnostic, contraction, and robustness claims are tested. The design holds the frozen backbone, interface, and training protocol fixed and varies only the refiner, so any difference is attributable to the refiner architecture alone. We compare refiners sharing the same attention core and a near-identical parameter budget: **M0**, a single-pass matched-MLP refiner (the prior best-accuracy *discrete* refiner on this backbone); **O2**, a

Table 1: Main comparison on the nuScenes val split (9,041 samples, 6 s horizon); all baselines re-evaluated under one protocol. A small refiner on a *frozen* backbone matches or improves every major design family across seven years of research, yet the margins separating the strongest methods are *sub-metric* – a symptom of the bounded gate that motivates the gate-forced diagnostic of Section 4.3. Standard nuScenes protocol; lower is better, **bold**/underline mark best/second in column.

Method	Venue	Year	Single-mode (K=1, no oracle)			Multi-mode (K=5, oracle)				
			minADE$_1$	minFDE$_1$	MR$_1$	minADE$_5$	minFDE$_5$	MR$_5$	brier-mADE$_5$	brier-mFDE$_5$
Raster-based and early multi-modal predictors										
MTP (Cui et al., 2019)	ICRA	2019	5.42	10.36	0.954	1.694	3.923	0.772	2.585	4.621
CoverNet (Phan-Minh et al., 2020)	CVPR	2020	6.66	14.91	0.975	2.352	5.242	0.640	3.124	6.013
Lane-graph and goal-conditioned decoders										
PGP (Deo et al., 2022)	CoRL	2021	4.86	9.99	0.913	1.625	3.537	0.750	2.395	3.890
Goal-LBP (Yao et al., 2023)	TITS	2024	5.83	9.91	0.948	2.068	3.512	0.435	2.768	4.181
Vector-transformer encoders										
AutoBots (Girgis et al., 2021)	ICLR	2022	3.90	6.65	0.872	1.377	2.350	**0.302**	1.965	3.090
FRM (Park et al., 2023)	ICLR	2023	4.04	6.91	0.882	1.428	2.439	0.318	2.026	3.165
MacFormer (Feng et al., 2023)	RA-L	2023	5.07	9.80	0.928	1.598	3.471	0.759	2.428	3.943
LAformer (Liu et al., 2024)	CVPRW	2024	3.43	6.69	0.853	1.213	2.367	0.412	1.853	2.997
Knowledge-graph and behaviour-driven refinements										
SemanticFormer (Sun et al., 2024)	RA-L	2024	4.50	7.67	0.901	1.592	2.715	0.351	2.157	3.282
CASPFormer (Yadav et al., 2024)	ICPR	2024	7.51	12.74	0.983	2.661	4.513	0.522	3.531	5.317
DeMo (Zhang et al., 2024)	NeurIPS	2024	4.94	8.39	0.919	1.749	2.968	0.379	2.354	3.572
MBRM (Distelzweig et al., 2025)	arXiv	2024	5.68	9.89	0.943	2.012	3.502	0.642	2.685	4.118
2025 baselines										
TSD (Shu et al., 2025)	arXiv	2025	5.52	12.27	0.937	1.808	4.343	0.687	2.580	4.812
LMFormer (Yadav et al., 2025)	CVPRW	2025	4.05	8.38	0.884	1.315	2.966	0.609	2.061	3.512
Retrospection (Hagedorn et al., 2025)	arXiv	2025	4.04	6.90	0.881	1.431	2.443	0.319	1.957	3.080
GC-GAT (Gulzar et al., 2025)	RA-L	2025	10.60	17.94	0.996	3.755	6.353	0.612	4.962	7.486
BLNet (Xiong et al., 2026)	arXiv	2025	6.07	10.29	0.957	2.149	3.643	0.449	2.871	4.336
NEST (Wang et al., 2025)	AAAI	2025	4.52	7.69	0.903	1.602	2.722	0.352	2.105	3.290
Foresight (Pei et al., 2025)	ICCV	2025	4.04	8.31	0.880	1.377	2.942	0.609	2.054	3.485
2026 baseline										
SAIL (Rao et al., 2026)	arXiv	2026	6.22	10.53	0.961	2.203	3.730	0.456	2.928	4.412
BReF (ours)	–	–	**3.35**	7.83†	**0.847**	**1.185**	**2.324**	0.407	**1.825**	**2.964**

† Prior methods' single-mode minFDE$_1$ is as published under each method's own protocol; BReF's 7.83 is our top-1 (no-oracle) evaluation, preserving the frozen backbone's endpoint error. Under our uniform protocol it ranks second of all methods (behind AutoBots' 7.77). Full convention in Appendix G.5.

Table 2: Plug-and-play refinement of nineteen frozen predictors: Δ = minADE$_5$ reduction (%) over the frozen epoch-30 baseline, at the floor-preserving (-2.5) and active (0.0) gate inits. Improve/tie/regress $12/6/1$ and $13/4/2$; the single regression is 2 mm and attributable to the Laplace-NLL objective. Per-method notes in Appendix F.8.

Backbone	base	$\Delta_{-2.5}$	$\Delta_{0.0}$
SAIL	2.203	+6.5	+6.2
TSD	1.808	+6.0	+4.3
BLNet	2.149	+4.9	+5.6
MTP	1.694	+4.5	+4.5
Trajectron++	2.107	+4.3	+5.5
Goal-LBP	2.068	+3.0	+5.9
PGP	1.625	+2.0	+2.2
CASPFormer	2.661	+1.8	+1.9
DeMo	1.749	+1.2	+1.1
MacFormer	1.598	+0.9	+1.0

Backbone	base	$\Delta_{-2.5}$	$\Delta_{0.0}$
FRM	1.428	+0.7	+0.8
AutoBots	1.377	+0.2	+0.3
Foresight	1.377	+0.0	+0.1
LMFormer	1.315	+0.0	+0.0
Retrospection	1.431	+0.0	+0.0
NEST	1.602	−0.0	−0.0
UniTraj	1.520	−0.0	−0.0
GC-GAT	3.755	−0.0	−0.2
SemanticFormer	1.592	−0.1	−0.1
mean Δ		+1.87%	+2.06%

Neural-ODE refiner with Euler integration and *no* smoothness regulariser; **SS3**, the ODE+smooth(τ) variant with latent-time smoothness – **the architecture used by BReF** throughout the headline results (the label is retained for row symmetry); and **Jt** (ODE + jerk loss in output time), which dissociates the smoothness penalty's *effect* from its *location* (latent τ vs output t).

4.3.1 Gate-Forced Diagnostic: A Two-Order-of-Magnitude Robustness Gap

The crux of the architectural claim is the *gate-forced* diagnostic: at inference time – without re-training – we set $\sigma(\boldsymbol{\gamma})\equiv 1$, removing the gate's $7\times$ suppression so the predictor commits to the raw residual $\boldsymbol{\delta}$, the only setting in which refiner families can be compared on *what each has actually learned*.

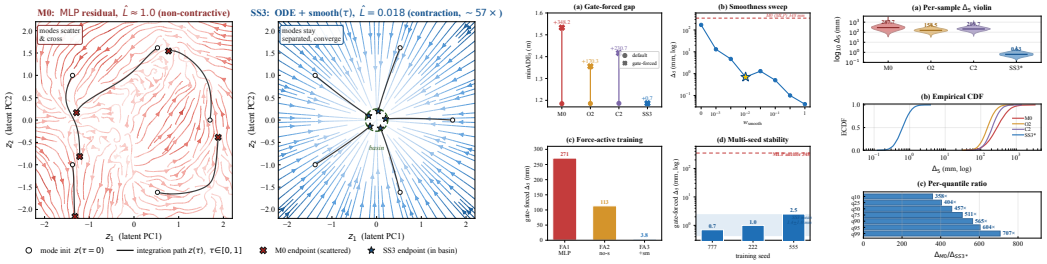
Refiner family	gate-forced	Δ_5 (mm)	vs SS3
M0 (MLP)	1533.0	+348.2	**497**× worse
O2 (ODE, no smooth)	1355.8	+170.3	**243**× worse
Jt (ODE + jerk in t)	1416.8	+230.7	**330**× worse
SS3 (ours)	**1186.5**	**+0.7**	**1**× **(ref.)**

(a) Anchor diagnostic.

σ (m)	M0 gate-fcd	SS3 gate-fcd	SS3 Δ_5	ratio
0	1533.0	**1188.0**	+1.4	**249**×
0.05	1534.9	**1192.2**	+1.1	**314**×
0.1	1540.3	**1203.1**	+0.9	**378**×
0.2	1560.9	**1239.2**	+0.8	**406**×
0.5	1690.3	**1418.9**	+0.6	**456**×
1.0	2059.9	**1838.6**	+0.5	**446**×
2.0	3038.3	**2829.6**	**+0.0**	**∞**

(b) Backbone-noise sweep.

Table 3: **Gate-forced diagnostic and its noise-swept extension. (a)** Gate-forced diagnostic (K=5, 9,041 val samples) – the evidence for raw-correction validity. Families whose default minADE$_5$ lies within a 1.3 mm band fan out across nearly three orders of magnitude when the gate is forced open: displacement error certifies the gate, not the refiner. **(b)** Backbone-noise robustness ($\sigma \in [0, 2.0]$ m). We inject zero-mean isotropic Gaussian noise into the backbone output and report gate-forced minADE$_5$ for both families. Default minADE$_5$ of M0 and SS3 tracks the noise-injected backbone to within ± 2 mm at every σ (suppressed by the gate, omitted) – the architectural separation appears *only* under gate-forced inference, where the SS3/MLP ratio rises from **249×** at σ=0 to **∞** at σ=2.0 (SS3's $\Delta_5 \to 0$).



(a) Latent flow: MLP vs. SS3. (b) Gate-forced stress. (c) Per-sample Δ_5.

Figure 2: **Stress-test evidence**, one aligned row (extended readings and additional controls in Appendices E–F). **(a)** The contraction mechanism – 2-D PCA streamplot of the latent flow field: the MLP flow ($\hat{L} \approx 1.0$) is non-laminar with crossing mode endpoints, the SS3 flow ($\hat{L}$=0.018) laminar and contractive. **(b)** Gate-forced stress test (with smoothness-sweep, force-active, and multi-seed panels): SS3 at Δ_5=+0.7 mm, a factor 497 below M0's +348 mm. **(c)** Per-sample Δ_5 distribution over all 9,041 val samples: SS3 stays 2–3 orders below the MLP at every quantile.

The gate-forced diagnostic (Table 3) reports two findings. **(a)** *Two-order-of-magnitude robustness gap*: the matched MLP degrades by 348.2 mm, SS3 by **0.7** mm – **497×** on the anchor snapshot – while the default minADE$_5$ of all four families lies within a 1.3 mm band because the trained gate uniformly suppresses raw residuals. **(b)** *Smoothness location matters*: Jt (jerk in output time) degrades by 230 mm, worse than the bare ODE (170 mm); smoothness in the latent variable τ cuts the degradation by $\sim 330\times$ over Jt.

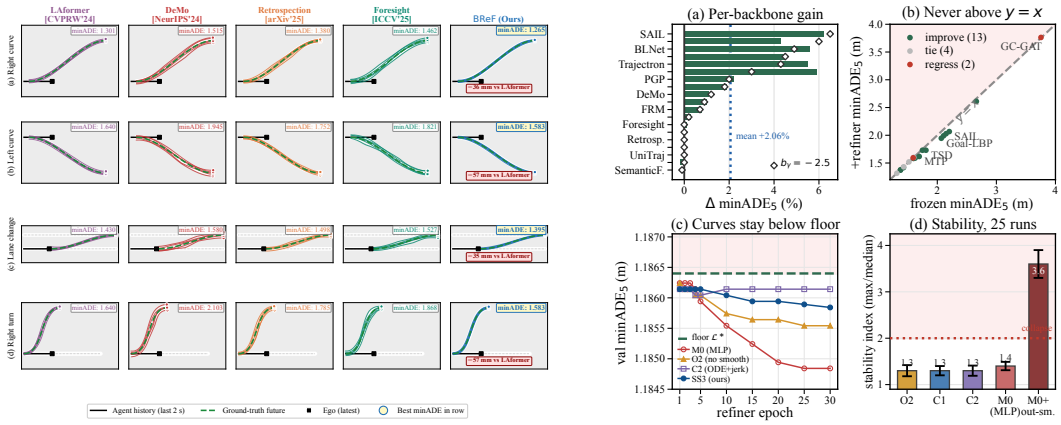
4.3.2 Auxiliary Confounders Ruled Out

Smoothness weight is the causal driver. Table 5 sweeps w_{smooth} over four orders of magnitude: removing smoothness entirely reproduces the O2 variant (Δ_5=170.3 mm), even $w_{\text{smooth}}=10^{-3}$ collapses the degradation by an order of magnitude, and the curve stays in $[0, 1.3]$ mm for any $w_{\text{smooth}} \geq 10^{-3}$ – the 497× ratio is the saturation of a monotonic mechanism, not a single-weight coincidence.

Further controls (Appendix F.2). *Force-active training*: re-training with the gate held near 1 does not recover the MLP. *Seeds*: three re-trainings hold the gap at 1.4 ± 1.0 mm – the worst seed remains 139× more robust than the MLP (Table 10).

4.3.3 Continuous-Time Consistency and Solver Invariance

If the ODE refiner has learned a genuine continuous vector field, the integration steps K and the solver choice should be near-irrelevant (an MLP admits *no analogue* of either choice). Sweeping



(a) Qualitative scenes, five methods.

(b) Portability; training dynamics.

Figure 3: **Qualitative examples, portability, and training dynamics** (extended readings in Appendices E.4 and F.8). **(a)** Four qualitative nuScenes scenes (LAformer, DeMo, Retrospection, Foresight, BReF; ground truth dashed green, K=5 modes coloured): BReF refines each backbone mode towards the ground truth while preserving the mode set – the joint refiner (DeMo) drops the right-turn mode in scene (d). **(b)** *(a,b)*: plug-and-play portability across nineteen frozen predictors – every backbone falls on or below the y=x line, the gain widening with the backbone's own error. *(c,d)*: floor-preserving training dynamics – BReF starts at the frozen backbone floor and never crosses it.

$K \in [1, 128]$ and four solvers moves minADE_5 by at most 0.2–0.4 mm while the non-smoothness families stay in the 170–240 mm band: F_θ is a discretised continuous flow (Tables 6, 7; Figs. 6a, 5b).

Out-of-regime robustness. Backbone-noise injection up to σ=2.0 m (SS3/MLP gate-forced ratio 249×→∞, Table 3b), geographic and behavioural train-test splits (at most $\sim$1.5 mm on the *held-out* regime, Tables 11–12), the per-sample Δ_{ADE} distribution (SS3 two-to-three orders below the MLP at every quantile, mode spread preserved), and recursive Lipschitz extrapolation ($\hat{L}_{\text{ODE}}$=0.018 vs $\hat{L}_{\text{MLP}}\approx$1.0) all confirm that the latent-time contraction carries BReF into the unseen regime (Appendices F.3–F.5.2). Figure 3 shows four qualitative scenes: BReF makes small, mode-preserving corrections (tightening curve apices, pulling lane changes onto the goal lane, preserving the right-turn mode the joint refiner drops), keeping visibly tighter residuals while staying close to the backbone's mode set.

4.4 Headline Result and Further Evidence

BReF trains 576 K refiner parameters in $\sim$ 1 hour on eight H800 accelerators. Beyond accuracy and gate-forced robustness, extended analyses show preserved jerk and acceleration to two decimals, mode spread within 5 mm of the MLP, and a 4× minFDE_5 advantage on the curve-heavy Q_5 quintile (Appendices E.2, F.6). Figure 6c summarises the evidence; Algorithm 1 gives pseudocode and Table 9 the standard ablations. Per-baseline tables and per-epoch JSON evaluations are released as supplementary material.

5 Conclusion

We presented BReF, a bounded residual flow refining a frozen multi-modal predictor. A refiner's displacement error is non-diagnostic – a bounded gate can compress an invalid raw correction below the metric's resolution – so what matters is validity *without* the gate; BReF makes it valid by construction (latent-time smoothness ⇒ contraction, gate near-redundant), with identity-initialisation giving a provable floor (Proposition 1). On nuScenes it leads twenty baselines on every multi-mode metric and single-mode minADE_1 while preserving the mode set, the gate-forced diagnostic separating it from a parameter-matched *discrete* refiner by 497×. Its benefits depend on backbone mode coverage: bounded local corrections cannot recover a plausible future absent from the candidate set. The contraction is empirical and single-pass; cross-family transfer, closed-loop rollouts, and humanoid motion prediction are future work.

AI USE STATEMENT

Among the tasks for which ICLR requires disclosure, we used generative AI tools to assist with implementing methods (parts of the refiner, training, and evaluation code) and, in a reviewing capacity, to check the statements and proofs in Appendix C for gaps. The theoretical claims, the bounded-residual-flow formulation, and the experimental design are the authors' own; generative AI tools did not originate the hypotheses, the methodology, or the mathematical results, and we did not use them to generate synthetic data, clean or reformat the dataset, or interpret results. The remaining categories requiring disclosure are not applicable to this work.

Among the tasks for which disclosure is recommended, we additionally used generative AI tools to create and lay out scientific figures, to edit prose for readability, and to search for and format references.

We have reviewed all AI-assisted work: AI-assisted code was read and tested by the authors, and every number reported in the paper was produced by our own executed runs rather than by a language model. Proof steps suggested or checked by AI tools were verified by the authors independently. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies trajectory forecasting, a component of autonomous-driving stacks, and we therefore note the safety implications of the claims we make. All experiments use the publicly released nuScenes prediction benchmark (Caesar et al., 2020) under its existing licence and standard evaluation protocol; we collected no new data, and the dataset contains no personally identifying information that we access or attempt to recover. No human subjects were involved.

Two limitations bear directly on safe use. First, every result is *open-loop* displacement error on a single benchmark: low minADE_5 does not establish that a planner conditioned on these predictions behaves safely, and the contraction property we prove (Proposition 1) is a statement about the refiner's single-pass residual, not a guarantee of downstream collision avoidance. Closed-loop evaluation remains future work, and we caution against reading our benchmark margins as evidence of deployment readiness. Second, nuScenes covers two cities and a limited range of conditions, so agent-behaviour distributions, road geometry, and weather seen here may not transfer; a predictor that inherits the frozen backbone's blind spots will inherit its failure modes on under-represented road users as well, since our refiner is bounded and mode-preserving by construction and therefore cannot recover a mode the backbone never proposed. We report cross-distribution behaviour in Appendix F precisely so that these limits are visible rather than implied. We see no dual-use concern specific to this work beyond those already inherent to autonomous-driving research.

REPRODUCIBILITY STATEMENT

All theoretical claims (Proposition 1 and its proof, the gate-compression analysis, and the training-dynamics consequences) are stated with complete assumptions and proved in Appendix C. Architecture and training hyperparameters, the outer-refiner training objective, the per-module parameter budget, the evaluation protocol, and multi-seed stability details are collected in Appendix G, with the full backbone specification in Appendix B. Extended diagnostics, full ablation tables, and solver-swap / K-step / noise / cross-distribution sweeps are reported in Appendices D, E, and F.

REFERENCES

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Marin Biloš, Johanna Sommer, Syama Sundar Rangapuram, Tim Januschowski, and Stephan Günnemann. Neural flows: Efficient alternative to neural odes. *Advances in neural information processing systems*, 34:21325–21337, 2021.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for

autonomous driving. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11618–11628. IEEE, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Zhi Cen, Huaijin Pi, Sida Peng, Zehong Shen, Minghui Yang, Shuai Zhu, Hujun Bao, and Xiaowei Zhou. Generating human motion in 3d scenes from text descriptions. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1855–1866. IEEE, 2024.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Jie Cheng, Xiaodong Mei, and Ming Liu. Forecast-mae: Self-supervised pre-training for motion forecasting with masked autoencoders. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8645–8655. IEEE, 2023.

Henggang Cui, Vladan Radosavljevic, Fang-Chieh Chou, Tsung-Han Lin, Thi Nguyen, Tzu-Kuo Huang, Jeff Schneider, and Nemanja Djuric. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. In *2019 international conference on robotics and automation (icra)*, pp. 2090–2096. IEEE, 2019.

Nachiket Deo, Eric Wolff, and Oscar Beijbom. Multimodal trajectory prediction conditioned on lane-graph traversals. In *Conference on robot learning*, pp. 203–212. PMLR, 2022.

Aron Distelzweig, Eitan Kosman, Andreas Look, Faris Janjoš, Denesh K Manivannan, and Abhinav Valada. Motion forecasting via model-based risk minimization. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7011–7018. IEEE, 2025.

Chen Feng, Hangning Zhou, Huadong Lin, Zhigang Zhang, Ziyao Xu, Chi Zhang, Boyu Zhou, and Shaojie Shen. Macformer: Map-agent coupled transformer for real-time and robust trajectory prediction. *IEEE Robotics and Automation Letters*, 8(10):6795–6802, 2023.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. Humanplus: Humanoid shadowing and imitation from humans. *arXiv preprint arXiv:2406.10454*, 2024.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *International conference on machine learning*, pp. 1607–1616. PMLR, 2018.

Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11522–11530. IEEE, 2020.

Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. Home: Heatmap output for future motion estimation. In *2021 IEEE international intelligent transportation systems conference (ITSC)*, pp. 500–507. IEEE, 2021.

Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. Gohome: Graph-oriented heatmap output for future motion estimation. In *2022 international conference on robotics and automation (ICRA)*, pp. 9107–9114. IEEE, 2022.

Roger Girgis, Florian Golemo, Felipe Codevilla, Martin Weiss, Jim Aldon D'Souza, Samira Ebrahimi Kahou, Felix Heide, and Christopher Pal. Latent variable sequential set transformers for joint multi-agent motion prediction. *arXiv preprint arXiv:2104.00563*, 2021.

Junru Gu, Chen Sun, and Hang Zhao. Densetnt: End-to-end trajectory prediction from dense goal sets. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15283–15292. IEEE, 2021.

Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic trajectory prediction via motion indeterminacy diffusion. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17092–17101. IEEE, 2022.

Mahir Gulzar, Yar Muhammad, and Naveed Muhammad. Gc-gat: Multimodal vehicular trajectory prediction using graph goal conditioning and cross-context attention. *IEEE Robotics and Automation Letters*, 10(8):8316–8323, 2025.

Steffen Hagedorn, Aron Distelzweig, Marcel Hallgarten, and Alexandru P Condurache. Learning through retrospection: improving trajectory prediction for automated driving with error feedback. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12064–12069. IEEE, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. *arXiv preprint arXiv:2406.08858*, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 876–885, 2018.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European conference on computer vision*, pp. 709–727. Springer, 2022.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems*, 36:20067–20079, 2023a.

Chiyu Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, Dragomir Anguelov, et al. Motiondiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9644–9653, 2023b.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 3045–3059, 2021.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Mengmeng Liu, Hao Cheng, Lin Chen, Hellward Broszio, Jiangtao Li, Runjiang Zhao, Monika Sester, and Michael Ying Yang. Laformer: Trajectory prediction for autonomous driving with lane-aware scene constraints. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2039–2049. IEEE, 2024.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple & efficient attention networks. *arXiv preprint arXiv:2207.05844*, 2022.

Daehee Park, Hobin Ryu, Yunseo Yang, Jegyeong Cho, Jiwon Kim, and Kuk-Jin Yoon. Leveraging future relationship reasoning for vehicle trajectory prediction. *arXiv preprint arXiv:2305.14715*, 2023.

Muleilan Pei, Shaoshuai Shi, Xuesong Chen, Xu Liu, and Shaojie Shen. Foresight in motion: Reinforcing trajectory prediction with reward heuristics. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28303–28312. IEEE, 2025.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Tung Phan-Minh, Elena Corina Grigore, Freddy A Boulton, Oscar Beijbom, and Eric M Wolff. Covernet: Multimodal behavior prediction using trajectory sets. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14062–14071. IEEE, 2020.

Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992.

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

Bin Rao, Haicheng Liao, Chengyue Wang, Keqiang Li, Zhenning Li, and Hai Yang. Sail: Scene-aware adaptive iterative learning for long-tail trajectory prediction in autonomous vehicles. *Transportation Research Part C: Emerging Technologies*, 190:105757, 2026.

Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *European conference on computer vision*, pp. 683–700. Springer, 2020.

Ari Seff, Brian Cera, Dian Chen, Mason Ng, Aurick Zhou, Nigamaa Nayakanti, Khaled S Refaat, Rami Al-Rfou, and Benjamin Sapp. Motionlm: Multi-agent motion forecasting as language modeling. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8545–8556. IEEE, 2023.

Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. *Advances in Neural Information Processing Systems*, 35:6531–6543, 2022.

Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Mtr++: Multi-agent motion prediction with symmetric scene modeling and guided intention querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3955–3971, 2024.

Peng Shu, Pengfei Zhu, Mengshi Qi, and Liang Liu. Improving partially observed trajectories forecasting by target-driven self-distillation. *arXiv preprint arXiv:2501.16767*, 2025.

Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. *Advances in neural information processing systems*, 28, 2015.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Zhigang Sun, Zixu Wang, Lavdim Halilaj, and Juergen Luettin. Semanticformer: Holistic and semantic traffic scene representation for trajectory prediction using knowledge graphs. *IEEE Robotics and Automation Letters*, 9(9):7381–7388, 2024.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

Balakrishnan Varadarajan, Ahmed Hefny, Avikalp Srivastava, Khaled S Refaat, Nigamaa Nayakanti, Andre Cornman, Kan Chen, Bertrand Douillard, Chi Pang Lam, Dragomir Anguelov, et al. Multipath++: Efficient information fusion and trajectory aggregation for behavior prediction. In *2022 international conference on robotics and automation (ICRA)*, pp. 7814–7821. IEEE, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Chengyue Wang, Haicheng Liao, Bonan Wang, Yanchen Guan, Bin Rao, Ziyuan Pu, Zhiyong Cui, Cheng-Zhong Xu, and Zhenning Li. Nest: A neuromodulated small-world hypergraph trajectory prediction model for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 808–816, 2025.

Xishun Wang, Tong Su, Fang Da, and Xiaodong Yang. Prophnet: Efficient agent-centric motion forecasting with anchor-informed proposals. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21995–22003. IEEE, 2023a.

Yinhuai Wang, Jing Lin, Ailing Zeng, Zhengyi Luo, Jian Zhang, and Lei Zhang. Physhoi: Physics-based imitation of dynamic human-object interaction. *arXiv preprint arXiv:2312.04393*, 2023b.

Wenyi Xiong, Jian Chen, and Ziheng Qi. Fine-grained behavior and lane constraints guided trajectory prediction method. *IEEE Transactions on Intelligent Transportation Systems*, 2026.

Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1410–1420. IEEE, 2023.

Harsh Yadav, Maximilian Schaefer, Kun Zhao, and Tobias Meisen. Caspformer: Trajectory prediction from bev images with deformable attention. In *International conference on pattern recognition*, pp. 420–434. Springer, 2024.

Harsh Yadav, Maximilian Schäfer, Kun Zhao, and Tobias Meisen. Lmformer: Lane based motion prediction transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2426–2435. IEEE, 2025.

Zhen Yao, Xin Li, Bo Lang, and Mooi Choo Chuah. Goal-lbp: Goal-based local behavior guided trajectory prediction for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 25(7):6770–6779, 2023.

Bozhou Zhang, Nan Song, and Li Zhang. Demo: Decoupling motion forecasting into directional intentions and dynamic states. In *NeurIPS*, 2024.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14730–14740, 2023a.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023b.

Hang Zhao, Jiyang Gao, Tian Lan, Chen Sun, Ben Sapp, Balakrishnan Varadarajan, Yue Shen, Yi Shen, Yuning Chai, Cordelia Schmid, et al. Tnt: Target-driven trajectory prediction. In *Conference on robot learning*, pp. 895–904. PMLR, 2021.

Yang Zhou, Hao Shao, Letian Wang, Steven L Waslander, Hongsheng Li, and Yu Liu. Smartrefine: A scenario-adaptive refinement framework for efficient motion prediction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15281–15290. IEEE, 2024.

Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. Hivt: Hierarchical vector transformer for multi-agent motion prediction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8813–8823. IEEE, 2022.

Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17863–17873, 2023.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.

APPENDIX — SUPPLEMENTARY MATERIAL

A EXTENDED RELATED WORK

This appendix expands the condensed discussion of Section 2: it details the three lines of work that supply the backbone we freeze (Appendix A.1), the adaptation primitive we attach (Appendix A.2), and the supporting machinery of continuous latent dynamics, physics-informed supervision and optimisation geometry (Appendix A.3). Where multiple variants of a design family exist, we discuss the most recent and best-performing representatives; per-baseline numerical comparisons are given in Section 4.

A.1 VECTOR TRAJECTORY PREDICTORS AND OUTPUT REFINEMENT

Vector encoders and HD-map priors. Modern multi-modal forecasting fuses two ingredients: *vector* representations of agents and HD maps (VectorNet (Gao et al., 2020), HiVT (Zhou et al., 2022), QCNet (Zhou et al., 2023), Wayformer (Nayakanti et al., 2022), with recent twists such as SE(3)-equivariance in EqMotion (Xu et al., 2023), masked-trajectory pretraining in Forecast-MAE (Cheng et al., 2023), and anchor-aware mode-diversity in ProphNet (Wang et al., 2023a)); and explicit HD-map *priors*, ranging from goal-driven decoders (TNT (Zhao et al., 2021), DenseTNT (Gu et al., 2021), PGP (Deo et al., 2022)) and occupancy heatmaps (HOME (Gilles et al., 2021), GoHome (Gilles et al., 2022)) to lane-mask-conditioned attention (LMFormer (Yadav et al., 2025)) and BEV deformable attention (CASPFormer (Yadav et al., 2024)). LAformer (Liu et al., 2024) reliably fuses the two axes via a dense per-step lane-likelihood map that gates cross-attention; we adopt its encoder and lane-scoring as the base of the backbone we freeze, augmented with a latent-ODE refinement of our own (Appendix B.3).

Diffusion, language-model and IRL formulations. MotionDiffuser (Jiang et al., 2023b) and MID (Gu et al., 2022) are diffusion-based predictors; MotionLM (Seff et al., 2023) casts refined multi-agent forecasting as autoregressive language modelling; Foresight (Pei et al., 2025) adds an IRL-grounded intention reasoner to a DETR-style refinement decoder; SemanticFormer (Sun et al., 2024) lifts the agent–map graph to a heterogeneous knowledge graph; and Retrospection (Hagedorn et al., 2025) re-uses an already-trained predictor's errors through closed-loop inference-time feedback.

Refinement heads (closest to BReF). A growing line of work refines *already-produced* predictor trajectories: MTR (Shi et al., 2022) / MTR++ (Shi et al., 2024) update learnable motion queries against fixed anchors; QCNet (Zhou et al., 2023) applies anchor-based refinement queries on top of query-centric base predictions; SmartRefine (Zhou et al., 2024) learns a scenario-adaptive re-attention policy; MultiPath++ (Varadarajan et al., 2022) learns latent anchor embeddings with context-aware gating; and DeMo (Zhang et al., 2024) decouples directional-intention queries from dynamic-state queries inside the refiner. *All of these train the refiner jointly with the backbone*, so its initialisation is random and its output can in principle diverge arbitrarily from the preceding base trajectories. BReF departs on three points. First, the refiner is appended *after* a frozen backbone (decoupled optimisation) and is *identity-initialised*, so it reproduces the backbone element-wise at the first optimiser step (Proposition 1) – the *floor-preserving* guarantee none of the jointly-trained heads above can offer. Second, it is a *contractive Neural-ODE*: a smoothness penalty in its latent integration time makes the raw correction valid by construction rather than gate-masked, so the gate becomes near-redundant. Third, it adds only 576 K parameters ($\sim 13\times$ smaller than the backbone). The discrete, jointly-trained refiners above are precisely the design that our parameter-matched control instantiates; the gate-forced diagnostic of Section 4.3 separates them from BReF by nearly three orders of magnitude.

Other 2024–2026 directions. AutoBots (Girgis et al., 2021) is a latent-variable set transformer. MacFormer (Feng et al., 2023) couples map and agent tokens through a bilateral query; FRM (Park et al., 2023) reasons over future agent-pair lane occupancies; NEST (Wang et al., 2025) models interactions with a neuromodulated small-world hypergraph; BLNet (Xiong et al., 2026) couples behaviour-intention with lane-constraint streams; GC-GAT (Gulzar et al., 2025) conditions on graph goal proposals; Goal-LBP (Yao et al., 2023) uses location-conditioned local-behaviour priors; TSD (Shu et al., 2025) targets partial-observation robustness; MBRM (Distelzweig et al., 2025) ensembles via model-based risk minimisation; and SAIL (Rao et al., 2026) (April 2026) tackles long-tail trajectories with scene-aware iterative contrastive learning. We reproduce all of these on the same nuScenes prediction split and benchmark BReF head-to-head in Section 4.

Humanoid motion prediction. Outside autonomous driving, multi-modal motion prediction matures rapidly in humanoid manipulation and locomotion – e.g. MotionGPT (Jiang et al., 2023a) and T2M-GPT (Zhang et al., 2023a) for text-driven synthesis, HumanPlus (Fu et al., 2024) and OmniH2O (He et al., 2024) as whole-body imitation stacks, and PhysHOI (Wang et al., 2023b) / (Cen et al., 2024) for scene-affordance-conditioned motion. The floor-preserving construction is expected to transfer to these settings whenever a strong backbone exists and a verifiable downside guarantee is desired.

A.2 Parameter-Efficient Adaptation of Frozen Backbones

BReF's gated additive structure combines residual learning (He et al., 2016) – the additive skip that lets a network learn $F(\mathbf{x})-\mathbf{x}$ – with the gated control of Highway Networks (Srivastava et al., 2015; Hochreiter & Schmidhuber, 1997). The iterative-refinement paradigm in DETR (Carion et al., 2020) and Deformable DETR (Zhu et al., 2020) shares the same additive structure at the decoder-query level, but trained end-to-end with the encoder. Unlike either antecedent, BReF attaches the construction to an *independently optimised* predictor whose weights are held fixed, so a precisely identity-initialised gate yields a verifiable floor.

This frozen-backbone-plus-trainable-head pattern has matured in language and vision: adapters (Houlsby et al., 2019), LoRA (Hu et al., 2021) and its adaptive-budget extension AdaLoRA (Zhang et al., 2023b), prompt tuning (Lester et al., 2021), and visual prompt tuning (Jia et al., 2022). All of these transfer the model across *tasks*, whereas BReF operates on the *same* task and dataset and instead relaxes the suboptimality gap introduced by the backbone's restricted optimisation – closer in spirit to born-again self-distillation (Furlanello et al., 2018; Hinton et al., 2015), where a student initialised to mimic the teacher is retrained on the same task and frequently finds a tighter local optimum.

A.3 Continuous Latent Dynamics, Physics Supervision, Optimisation Geometry

The backbone of BReF relies on three pieces of supporting machinery. *Continuous-time decoders* – Neural ODEs (Chen et al., 2018), Latent ODEs (Rubanova et al., 2019), and the simulation-free Flow Matching (Lipman et al., 2022) objective (with OT-CFM (Tong et al., 2023) and stochastic-interpolant (Albergo et al., 2025) extensions) – provide the integration primitive; Neural Flows (Biloš et al., 2021) amortise inference. FiLM (Perez et al., 2018) supplies the lightweight time-conditioned feature modulation we use inside the residual blocks. *Physics-informed supervision*: position-only labels underdetermine trajectory dynamics, and Physics-Informed Neural Networks (Raissi et al., 2019), together with derivative penalties in TNT (Zhao et al., 2021) and Trajectron++ (Salzmann et al., 2020), motivate the first-/second-/third-order finite-difference losses used by our backbone. *Optimisation geometry*: SWA (Izmailov et al., 2018) and its differentiable analogue EMA (Polyak (Polyak & Juditsky, 1992) / mean-teacher (Tarvainen & Valpola, 2017)) favour wider optima; SAM (Foret et al., 2020) and Test-Time Training (Sun et al., 2020) add worst-case neighbourhood / test-time adaptation. BReF adopts the simplest combination – slow EMA with cosine annealing (Loshchilov & Hutter, 2016) – because in our setting SAM, SWA and Test-Time Training are either redundant with EMA or actively regress once the base model is strong.

B Frozen Backbone: Full Architecture and Training

This appendix specifies the frozen lane-aware backbone summarised in Section 3.1. To avoid ambiguity: the *internal* latent-ODE refinement acts in a learned *latent* space and is frozen, and is a module distinct from the outer, trajectory-space Neural-ODE that BReF adds on top (Section 3.2) – both are designs of ours, but only the latter is trained in this paper.

B.1 Vector Encoder and Global Interaction Graph

The vector encoder follows the LAformer (Liu et al., 2024) extension of VectorNet (Gao et al., 2020). Past trajectories of all agents and HD-map polylines are converted into vectors $\{(p_s^i, p_e^i, a_i, \ell_i)\}$, where $p_s, p_e \in \mathbb{R}^2$ are the segment endpoints, a is an attribute (agent identifier, lane type), and ℓ is the polyline index. Two parallel point-level subgraphs encode polylines – one for agent histories, one for HD-map polylines – each consisting of three layers of MLP, max-pooling, and concatenation,

followed by a point-level cross-attention block that lets agents attend to nearby lane points and vice versa. The aggregated polyline descriptors then pass through a pair of single-layer lane-graph Transformer decoders (Vaswani et al., 2017) – Agent-to-Lane and Lane-to-Agent, each with eight attention heads – which exchange information bidirectionally between the agent and lane streams.

The polyline-level features then enter a global interaction graph composed of two parallel single-head self-attention sub-graphs whose outputs are *concatenated* (not summed) along the channel axis. With hidden size d=64 this produces $\mathbf{H}_{\text{scene}} \in \mathbb{R}^{N\times d}$, where N is the per-scene polyline count. Attention is masked by polyline validity; spatial neighbourhood restriction to a 50 m radius of the target agent is performed during data preprocessing.

B.2 Multi-Modal Decoding Head with Dense Lane Scoring

The decoding head produces K=5 candidate trajectories $\boldsymbol{\mu}_{1:K} \in \mathbb{R}^{K\times T_f\times 2}$ together with per-step Laplace scales $\boldsymbol{\sigma}_{1:K} \in \mathbb{R}_+^{K\times T_f\times 2}$, mode logits $\pi_{1:K} \in \mathbb{R}^K$, and a global trajectory encoding $\mathbf{g}^{\text{cat}} \in \mathbb{R}^{2d}$. A single-layer GRU recurrence rolls out T_f per-step hidden features that feed two parallel MLP heads (LayerNorm + ReLU) to produce $(\boldsymbol{\mu}_t, \boldsymbol{\sigma}_t)$, and a three-layer MLP head on a $2d$-dimensional global feature produces π.

In parallel, the head emits a temporally dense lane-likelihood map $\mathbf{S} \in \mathbb{R}^{N_\ell\times T_f}$ – the principal HD-map prior of LAformer (Liu et al., 2024). A cross-attention block lets each lane polyline attend to all agent features, followed by an MLP and a log-softmax over the N_ℓ scene lanes for each future step. At inference, the top k=2 lanes per step are selected from $\mathbf{S}$ and their features are aggregated through a second cross-attention block to drive the multi-modal mixture-of-experts projection. The training signal is the per-step lane-classification log-loss $\mathscr{L}_{\text{lane}} = \sum_{t=1}^{T_f} -\log p(\hat{\ell}_t = \ell_t^\star \,|\, \mathbf{H}_{\text{scene}})$, where $\ell_t^\star$ is the lane closest to the agent's true position at step t.

B.3 Latent-ODE Refinement with Flow-Matching Regularisation

We adopt an ODE-based latent refinement – rather than a plain MLP feed-forward block – because integrating a smooth drift field over a *latent* integration time yields a physically plausible trajectory correction by construction, encoding "physically plausible refinement" as an *architectural* inductive bias rather than relying on the gate to suppress pathological corrections. This is the same principle that the outer refiner of Section 3.2 makes precise and central – smoothness in the latent integration time, not the physical output time – and it contrasts with a plain MLP refiner, which has no such latent-time coupling, must rely on output-space smoothness penalties that destabilise training, and exposes raw pathological corrections whenever the gate fails open (Section 4.3).

The latent-ODE refinement module inside the backbone operates in a learned latent space. For each (mode, batch) index we form a context vector $\mathbf{c} \in \mathbb{R}^r$ (r=256) by attending a mode query – built from the global encoding, the decoding head's mode probability, and the sinusoidal embedding of its predicted endpoint – to the top eight lane features and top sixteen agent features. The latent state is initialised by a fully connected encoder $\mathbf{z}_0 = E(\boldsymbol{\mu}, \mathbf{c})$, and the latent dynamics are integrated over a *latent* integration time $\tau \in [0,1]$ – a fictitious variable internal to the solver, distinct from the physical trajectory time t that indexes the T_f future steps – by an explicit ODE solver,

$$\mathbf{z}(1) = \mathbf{z}_0 + \int_0^1 F_\theta(\mathbf{z}(\tau), \mathbf{c}, \tau)\, d\tau, \tag{6}$$

where F_θ comprises L_F=5 FiLM-modulated residual blocks of hidden width 512. Each block applies $\mathbf{h} \leftarrow \mathbf{h}\cdot(1+\tanh(\gamma_\tau)) + \beta_\tau$ with $(\gamma_\tau, \beta_\tau)$ produced from a 16-band Fourier embedding of the latent time $\tau \in [0,1]$ through a two-layer MLP. Integration uses a fourth-order Runge–Kutta scheme over T_{int}=3 uniform sub-intervals, so F_θ is evaluated 4×3=12 times per scene. A decoder D maps $\mathbf{z}(1)$ to a trajectory residual $\Delta\boldsymbol{\mu} = D(\mathbf{z}(1), \mathbf{c})$, giving the in-backbone refined prediction $\boldsymbol{\mu}^{\text{ref}} = \boldsymbol{\mu} + \Delta\boldsymbol{\mu}$.

We regularise the latent dynamics with a Flow Matching (Lipman et al., 2022) objective that aligns F_θ with the Optimal-Transport vector field linking $\mathbf{z}_0$ to the encoded ground-truth latent $\mathbf{z}^\star = E(\mathbf{y}^\star, \mathbf{c})$. Sampling $\tau^\star \sim \mathscr{U}(0,1)$ and forming the interpolant $\mathbf{z}_{\tau^\star} = (1-\tau^\star)\mathbf{z}_0 + \tau^\star \mathbf{z}^\star$,

$$\mathscr{L}_{\text{FM}} = \left\| F_\theta(\mathbf{z}_{\tau^\star}, \mathbf{c}, \tau^\star) - (\mathbf{z}^\star - \mathbf{z}_0) \right\|_{\text{MSE}}^2, \tag{7}$$

where $\|\cdot\|^2_{\text{MSE}}$ is the mean-squared error over the latent dimensions. A field-smoothness term $\mathscr{L}_{\text{smooth}} = \frac{1}{T_{\text{int}}}\sum_{i=0}^{T_{\text{int}}-1}\|\mathbf{z}(\tau_{i+1})-\mathbf{z}(\tau_i)\|^2_{\text{MSE}}$ over the RK4 integration grid $\{\tau_i\}$ further regularises the integrated latent trajectory.

B.4 Auxiliary Regularisers: Physics, Diversity, and Goal Anchoring

Position-only supervision underdetermines a trajectory's kinematics; following the PINN principle (Raissi et al., 2019), we supervise finite-difference derivatives of the best-mode predicted trajectory. With $\text{FD}^{(n)}$ denoting the n-th forward finite difference,

$$\mathscr{L}_{\text{pinn}} = \sum_{n\in\{1,2,3\}} w_n\,\mathbb{E}_t\left[\left\|\text{FD}^{(n)}\boldsymbol{\mu}_{k^\star,t} - \text{FD}^{(n)}\mathbf{y}^\star_t\right\|_2^2\right], \tag{8}$$

with $(w_1,w_2,w_3) = (4.5, 1.35, 0.225)$ for velocity, acceleration, and jerk respectively, all masked by per-step validity.

To prevent mode collapse, we add a multi-step diversity hinge $\mathscr{L}_{\text{div}}$ that penalises any pair of modes whose Euclidean distance at $1/3$, $2/3$, and the full horizon falls below the progressively larger margins $\{5/3,\ 10/3,\ 5\}$ m, averaged over pairs $i<j$ and over the three horizons; this is the multi-step generalisation of the single-step diversity losses common in motion forecasting (Varadarajan et al., 2022; Zhao et al., 2021).

Finally, a per-mode goal head $G:\mathbf{g}^{\text{cat}}\mapsto\mathbb{R}^{K\times 2}$ predicts K candidate endpoints. The refined trajectory is then blended toward the predicted endpoint with a quadratic time ramp $\boldsymbol{\mu}^{\text{final}}_{k,t} = \boldsymbol{\mu}^{\text{ref}}_{k,t} + \gamma_g\cdot(t/T_f)^2\cdot(\mathbf{g}_k - \boldsymbol{\mu}^{\text{ref}}_{k,T_f})$ controlled by a learnable scalar γ_g (initialised to 0.15), and supervised by the endpoint ℓ_2 regression $\mathscr{L}_g = \|\mathbf{g}_{k^\star} - \mathbf{y}^\star_{T_f}\|_2^2$ on the best mode.

B.5 Composite Training Loss

The complete backbone objective groups the components above into three loss families – output supervision, latent dynamics, and physics & geometry – aggregated as

$$\mathscr{L}_{\text{bb}} \;=\; \mathscr{L}_{\text{sup}} + \mathscr{L}_{\text{dyn}} + \mathscr{L}_{\text{phys+geom}}, \tag{9}$$

where the output-supervision family $\mathscr{L}_{\text{sup}} = \mathscr{L}_{\text{reg}} + \mathscr{L}_{\text{cls}} + \mathscr{L}_{\text{lane}}$ combines the best-mode Laplace NLL on the refined and decoded trajectories, a soft-target mode cross-entropy (target proportional to $\exp(-\frac{1}{T_f}\sum_t\|\boldsymbol{\mu}^{\text{final}}_{k,t} - \mathbf{y}^\star_t\|_2)$), and the dense per-step lane-classification term of Appendix B.2; the latent-dynamics family $\mathscr{L}_{\text{dyn}} = \mathscr{L}_{\text{FM}} + \mathscr{L}_{\text{smooth}}$ combines the Flow-Matching objective of Equation (7) with the integration-grid smoothness term of Appendix B.3; and the physics-and-geometry family $\mathscr{L}_{\text{phys+geom}} = \mathscr{L}_{\text{pinn}} + \mathscr{L}_g + \mathscr{L}_{\text{div}}$ sums the physics-informed derivative supervisions of Equation (8), the goal regression $\mathscr{L}_g$, and the diversity hinge $\mathscr{L}_{\text{div}}$. The per-step Laplace NLL is

$$\text{NLL}(\boldsymbol{\mu}_t,\boldsymbol{\sigma}_t;\mathbf{y}^\star_t) = \sum_{j\in\{x,y\}}\left[\log(2\sigma_{t,j}) + \frac{|y^\star_{t,j}-\mu_{t,j}|}{\sigma_{t,j}}\right], \tag{10}$$

with $\sigma_{t,j}$ clamped below by $\varepsilon{=}10^{-6}$ and the best mode selected by $k^\star = \arg\min_k\sum_t\|\boldsymbol{\mu}^{\text{final}}_{k,t} - \mathbf{y}^\star_t\|_2$.

B.6 Backbone Capacity and Freezing Protocol

The full backbone has approximately 7.44M parameters, of which ∼94% reside in the latent-ODE refinement module and primarily in its FiLM-modulated field network (hidden width 512, five residual blocks). After end-to-end training under $\mathscr{L}_{\text{bb}}$, the EMA snapshot at the epoch with the best validation minADE_K defines the frozen backbone used by the outer refiner of Section 3.2; all backbone parameters are then frozen and the backbone is held in evaluation mode for the entire refiner-training procedure. Concrete optimiser, learning-rate, schedule, batch-size, EMA-decay and hardware settings are reported in Section 4 and Appendix G.

C Proof of Proposition 1 and the Gate-Compression Mechanism

This appendix gives the formal floor-preserving guarantee and then derives *why* the default minADE_5 saturates across refiner families – the mechanism that motivates the gate-forced diagnostic.

C.1 FLOOR-PRESERVING PROPERTY

Setup. Let $\mu^{\text{bb}}_{k,t} \in \mathbb{R}^2$ be the frozen backbone's prediction for mode k at step t, with fixed Laplace scale $\sigma^{\text{bb}}_{k,t}$. The refiner emits a raw residual $\boldsymbol{\delta}_{k,t} = \boldsymbol{\delta}_{k,t}(\theta)$ and a gate logit $\gamma_{k,t}(\theta)$, and the refined prediction is

$$\hat{y}_{k,t}(\theta) = \mu^{\text{bb}}_{k,t} + \sigma(\gamma_{k,t}(\theta)) \odot \boldsymbol{\delta}_{k,t}(\theta), \tag{11}$$

with $\sigma(\cdot)$ the logistic gate. Identity initialisation sets the residual head's last layer to zero and the gate-logit bias to $b_\gamma \ll 0$, so at $\theta^{(0)}$ we have $\boldsymbol{\delta}^{(0)}_{k,t} = \mathbf{0}$ and $\sigma(\gamma^{(0)}_{k,t}) = \sigma(b_\gamma) \to 0^+$.

Proposition 1 (restated). *At the first optimiser step, $\hat{y}^{(0)}_{k,t} = \mu^{bb}_{k,t}$ element-wise; and there exists $\eta_0 > 0$ such that for any first-order update of step size $\eta \le \eta_0$ the refined Laplace NLL obeys $\mathscr{L}(\theta^{(1)}) \le \mathscr{L}^{bb}$, the backbone's loss.*

Proof. (i) Substituting $\boldsymbol{\delta}^{(0)} = \mathbf{0}$ into (11) gives $\hat{y}^{(0)}_{k,t} = \mu^{\text{bb}}_{k,t} + \sigma(b_\gamma) \odot \mathbf{0} = \mu^{\text{bb}}_{k,t}$, so $\mathscr{L}(\theta^{(0)}) = \mathscr{L}^{\text{bb}}$ exactly. (ii) The refined loss $\mathscr{L}(\theta) = \mathscr{L}^{\text{bb}} + \langle \nabla_\theta \mathscr{L}(\theta^{(0)}), \theta - \theta^{(0)} \rangle + O(\|\theta - \theta^{(0)}\|^2)$ is C^1 in θ with L_∇-Lipschitz gradient on the compact sublevel set $\{\theta : \mathscr{L}(\theta) \le \mathscr{L}^{\text{bb}}\}$. A gradient step $\theta^{(1)} = \theta^{(0)} - \eta\, g$, $g = \nabla_\theta \mathscr{L}(\theta^{(0)})$, yields the standard descent bound $\mathscr{L}(\theta^{(1)}) \le \mathscr{L}^{\text{bb}} - \eta(1 - \frac{\eta L_\nabla}{2})\|g\|^2$. For $\eta \le \eta_0 := 1/L_\nabla$ the bracket is non-negative, so $\mathscr{L}(\theta^{(1)}) \le \mathscr{L}^{\text{bb}}$. □

Corollary 1. Each metric in Corollary 1 is a deterministic functional of the predicted trajectories; substituting the pointwise equality $\hat{\mathbf{y}}_{1:K}|_{\phi=\phi_0} = f_{\text{bb}}(\cdot)$ into each functional yields equality with the backbone's value. The argument transfers verbatim to the hardsigmoid variant used in our implementation, since $\text{hardsigmoid}(b_\gamma) \cdot \mathbf{0} = \mathbf{0}$; the value $\sigma(b_\gamma) = \sigma(-2.5) \approx 0.0759$ plays no role at initialisation – its purpose is to bound the magnitude of the early-training correction once W_δ begins to drift from $\mathbf{0}$.

The argument extends inductively while the iterate remains in the sublevel set; the bounded gate $\|\sigma(\gamma) \odot \boldsymbol{\delta}\| \le \|\boldsymbol{\delta}\|$ keeps the refined output in a neighbourhood of μ^{bb} where the L_∇-smoothness constant is uniform. Joint refinement enjoys no such guarantee because its first iterate is unconstrained relative to μ^{bb}.

C.2 WHY THE DEFAULT METRIC SATURATES (GATE-COMPRESSION)

Consider two refiner families A, B trained to the same gate-regulariser optimum, with average gate value $\bar{\sigma} = \mathbb{E}_{k,t}[\sigma(\gamma_{k,t})]$. Their refined predictions differ only through

$$\hat{y}^A_{k,t} - \hat{y}^B_{k,t} = \sigma(\gamma^A_{k,t}) \odot \boldsymbol{\delta}^A_{k,t} - \sigma(\gamma^B_{k,t}) \odot \boldsymbol{\delta}^B_{k,t}. \tag{12}$$

Taking norms and using $\sigma(\gamma) \le \bar{\sigma} + \varepsilon_\sigma$ (gate concentration, empirically $\bar{\sigma} = 0.143$, $\varepsilon_\sigma < 0.05$),

$$\|\hat{y}^A - \hat{y}^B\| \;\le\; (\bar{\sigma} + \varepsilon_\sigma)\left(\|\boldsymbol{\delta}^A\| + \|\boldsymbol{\delta}^B\|\right). \tag{13}$$

Equation (13) is the crux: even when the *raw* residuals differ by hundreds of millimetres ($\|\boldsymbol{\delta}^A - \boldsymbol{\delta}^B\| \sim$ 350 mm under the gate-forced probe), the gate scales that difference by $\bar{\sigma} \approx 0.143$ before it reaches the metric, so the observable default-minADE gap is compressed below the ~ 1 mm resolution at which all four families tie (Table 3a). Forcing $\sigma(\gamma) \equiv 1$ removes the $\bar{\sigma}$ factor in (13) and recovers the raw-residual gap – this is exactly the gate-forced diagnostic of Section 4.3, and the reason it, not the default metric, is the discriminating axis. Figure 4a visualises this compression: the raw residual magnitudes span $\sim 500\times$ across families, yet after the $\bar{\sigma}$ gate the default-minADE$_5$ values collapse into a 1.3 mm band.

C.3 TRAINING-DYNAMICS CONSEQUENCES AND COMPARISON WITH JOINT REFINEMENT

Consequence for training dynamics. Let $\mathscr{L}_{\text{refiner}}(\phi) \triangleq \mathbb{E}_{(\mathbf{x},\mathbf{y}^\star)\sim\mathscr{D}}\left[\ell\left(\hat{\boldsymbol{\mu}}(\phi;\mathbf{x}), \mathbf{y}^\star\right)\right]$ denote the population loss the refiner minimises, with $\mathscr{D}$ the empirical training distribution and $\ell(\cdot,\cdot)$ the per-example loss defined in Section 3.5. By Proposition 1, $\hat{\boldsymbol{\mu}}(\phi_0;\mathbf{x}) = \boldsymbol{\mu}^{\text{final}}(\mathbf{x})$, hence

$$\mathscr{L}_{\text{refiner}}(\phi_0) \;=\; \mathbb{E}_{(\mathbf{x},\mathbf{y}^\star)\sim\mathscr{D}}\left[\ell\left(\boldsymbol{\mu}^{\text{final}}(\mathbf{x}), \mathbf{y}^\star\right)\right] \;=:\; \mathscr{L}^{\text{bb}}. \tag{14}$$

Table 4: Twelve-variant loss ablation on 9,041 val samples (15 epochs). Columns: refiner type and the four physical losses; default minADE_5 / minFDE_5 / MR.

Variant	**Ref.**	jerk	curv	lat	sm(τ)	mADE_5	mFDE_5
M0	MLP	–	–	–	–	1.1853	2.3237
LA1	MLP	✓	–	–	–	1.1856	2.3238
LA2	MLP	–	✓	–	–	1.1852	2.3236
LA3	MLP	–	–	✓	–	1.1856	2.3241
LA4	MLP	✓	✓	✓	–	1.1852	2.3237
O2	ODE	–	–	–	–	1.1855	2.3235
LA5	ODE	–	✓	–	–	1.1855	2.3238
LA6	ODE	–	–	✓	–	1.1855	2.3233
Jt	ODE	✓	–	–	–	1.1855	2.3234
LA7	ODE	✓	✓	✓	✓	1.1855	2.3242
LA8	ODE	✓	–	–	✓	1.1858	2.3240
SS3	**ODE**	–	–	–	✓	**1.1854**	2.3241

For any first-order update $\phi_{t+1} = \phi_t - \eta_t \widehat{\nabla}_\phi \mathscr{L}_{\text{refiner}}(\phi_t)$ with sufficiently small step η_t, standard descent arguments imply that $\mathscr{L}_{\text{refiner}}(\phi_{t+1}) \le \mathscr{L}_{\text{refiner}}(\phi_t)$ in expectation, so the optimisation trajectory $\{\phi_t\}$ stays within the sublevel set $\{\phi : \mathscr{L}_{\text{refiner}}(\phi) \le \mathscr{L}^{\text{bb}}\}$. Inside this set the refined prediction is at least as accurate as the backbone under ℓ; the entire trajectory, not merely its endpoint, is bounded below by the backbone's performance.

The loss–metric gap leaves room for non-monotonicity in minADE_K itself, since ℓ is the Laplace negative log-likelihood whereas minADE_K is an ℓ_2 displacement, and since we evaluate the EMA snapshot rather than the raw weights. Nonetheless, empirically the combination of (i) identity initialisation, (ii) the gate-magnitude regulariser, and (iii) the Laplace objective produces a monotonically decreasing minADE_5 during the latter half of training (Appendix F.7), consistent with the basin-of-attraction argument below.

Comparison to non-floor-preserving designs. Consider an alternative refiner whose final output is $g(\mathbf{x};\phi)$ rather than $f_{\text{bb}}(\mathbf{x}) + g(\mathbf{x};\phi)$. Under any standard random initialisation of ϕ, $g(\mathbf{x};\phi_0)$ is generally far from $\boldsymbol{\mu}^{\text{final}}(\mathbf{x})$ and so $\mathscr{L}_{\text{refiner}}(\phi_0) \gg \mathscr{L}^{\text{bb}}$. The optimiser must traverse a substantial region of weight space before recovering accuracy comparable to the backbone, and there is no a priori guarantee that the basin of attraction of θ_{bb} is reachable from the random initial point. In the geometric language of loss landscapes, the floor-preserving refiner places its starting point *inside* the basin of attraction of the backbone solution, whereas a randomly-initialised end-to-end refiner places it outside, with no bound on the cost of finding the way back in. The empirical regressions we observed when we attempted full end-to-end paradigm shifts – as well as our own ablations on this design space – are consistent with this prediction.

D Complete Ablation Tables

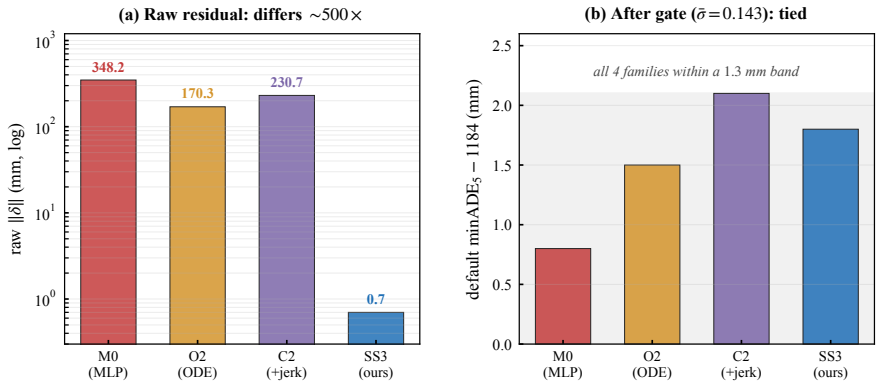
All numbers are server-verified on the 9,041-sample nuScenes val split. Gate-forced Δ_5 is the inference-time $\sigma(\gamma)\equiv 1$ degradation in millimetres. Figure 5a summarises the two headline ablation axes at a glance.

D.1 Twelve-variant loss ablation

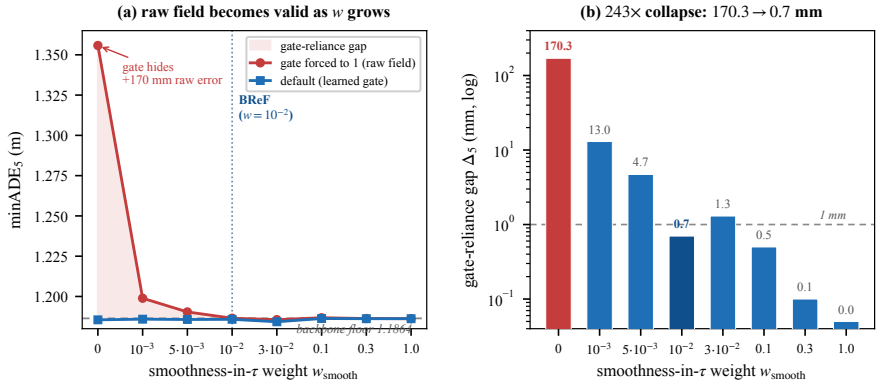
Table 4 factorially crosses the matched-MLP and ODE refiners with jerk-in-t, curvature, lateral-acceleration, and smoothness-in-τ losses (15 epochs, identical frozen backbone). All twelve default minADE_5 values lie in a 0.6 mm band; the variants are separated only by the gate-forced axis, where smoothness-in-τ is the unique driver.

D.2 Smoothness-weight sweep

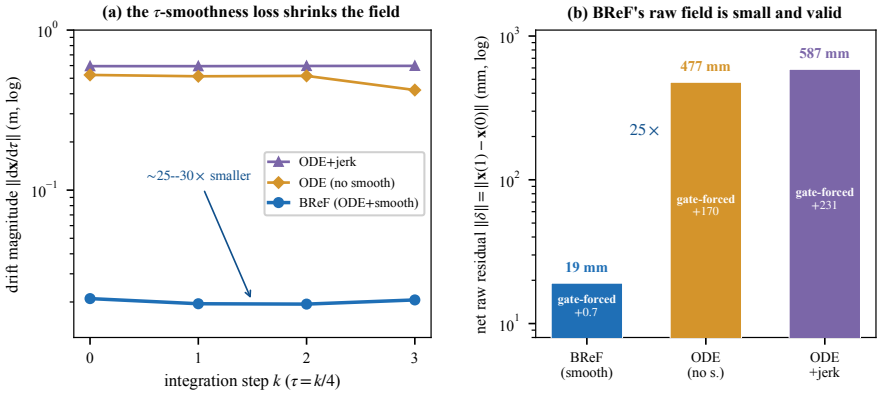
The single hyperparameter that separates BReF from a plain ODE refiner is the smoothness-in-τ weight w_{smooth}. Table 5 sweeps it across two decades. The default metric is flat – all rows lie within a



(a) Gate-compression mechanism.



(b) Smoothness-weight sweep.



(c) Drift-field analysis.

Figure 4: **Gate compression, smoothness collapse, and the drift field** (server-verified; see the surrounding text for the full reading). **(a)** Gate-compression mechanism: the gate-forced raw residual $\|\boldsymbol{\delta}\|$ differs by $\sim$500$\times$ across the four refiner families (348.2 mm for the matched MLP down to 0.7 mm for SS3); after multiplication by the trained gate ($\bar{\sigma}$=0.143) the default minADE$_5$ of all four families lies within a 1.3 mm band – the architectural difference is real but invisible to the default metric, motivating the gate-forced diagnostic. **(b)** Smoothness-weight sweep: as w_{smooth} grows, the gate-forced minADE$_5$ descends from 1.356 m to the backbone floor while the default metric never leaves the 1.184–1.186 m band – a 243$\times$ collapse from 170.3 mm to 0.7 mm at the operating point w_{smooth}=10^{-2}, vanishing by w_{smooth}=1. **(c)** Drift-field analysis: the τ-smoothness loss holds BReF's per-step drift at $\approx$0.02 m ($\sim$25–30$\times$ below the no-smoothness and jerk variants) and its net raw residual at 19 mm (vs. 477–587 mm), so forcing the gate to 1 costs only +0.7 mm.

Table 5: Smoothness-weight sweep (gate-forced, server-verified). The default minADE_5 is flat; the gate-forced gap Δ_5 collapses 243× as w_{smooth} grows. Operating point shaded.

w_{smooth}	**default**	**gate=1**	Δ_5 **(mm)**	Δ_5 **(%)**
0	1.1855	1.3558	+170.3	14.37
10^{-3}	1.1859	1.1989	+13.0	1.10
$5{\cdot}10^{-3}$	1.1857	1.1904	+4.7	0.40
10^{-2} (ours)	1.1858	1.1865	**+0.7**	**0.06**
$3{\cdot}10^{-2}$	1.1843	1.1856	+1.3	0.11
10^{-1}	1.1863	1.1868	+0.5	0.04
$3{\cdot}10^{-1}$	1.1862	1.1863	+0.1	0.01
1	1.1862	1.1862	+0.0	0.00

Table 6: Solver-swap gate-forced Δ_5 (mm) at K=4.

Solver	**O2 (no s.)**	**Jt (+jerk)**	**SS3 (ours)**
Euler	+170.3	+230.7	**+0.7**
Heun	+168.0	+228.6	**+0.7**
Midpoint	+174.7	+227.2	**+0.7**
RK4	+172.2	+226.9	**+0.7**

Table 7: Inference K-step gate-forced Δ_5 (mm).

Inf. K	**O2**	**Jt**	**SS3**
1	+186.1	+239.2	**+1.0**
2	+184.4	+223.2	**+0.7**
4	+170.3	+230.7	**+0.7**
8	+172.3	+228.1	**+0.7**
16	+173.0	+227.3	**+0.7**
32	+173.5	+228.0	**+0.7**
64	+173.5	+228.0	**+0.7**
128	+173.5	+228.0	**+0.7**

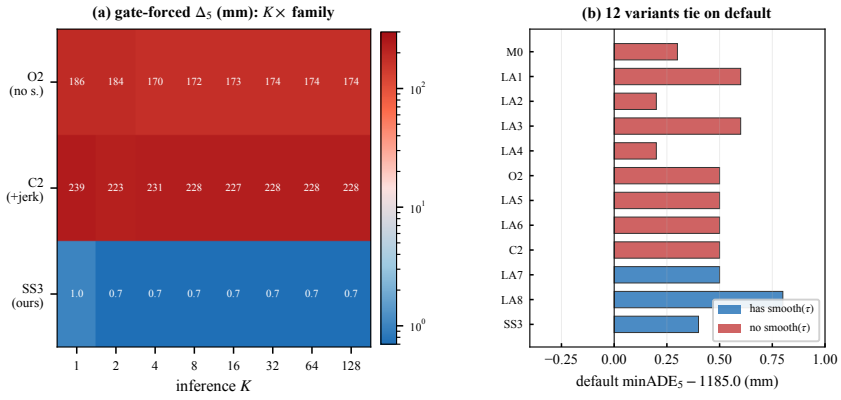
2 mm band, the saturation of Appendix C.2 – but the gate-forced Δ_5, the load the outer gate must carry to hide an invalid raw field, collapses monotonically from +170.3 mm at w_{smooth}=0 to +0.7 mm at the operating point $w_{\text{smooth}}=10^{-2}$ (a 243× reduction) and to 0.0 mm by w_{smooth}=1. Figure 4b plots both quantities. Two readings follow. First, the regulariser does not improve default accuracy – it establishes *field validity*: by $w_{\text{smooth}}=10^{-2}$ the raw drift field already integrates to within 0.7 mm of the gated output, so the gate ceases to be load-bearing. Second, the effect saturates beyond 10^{-2}, where the gap is already sub-millimetre; we therefore fix $w_{\text{smooth}}=10^{-2}$ as the smallest weight that attains field validity without perturbing the default metric.

D.3 Why smoothness works: the drift field

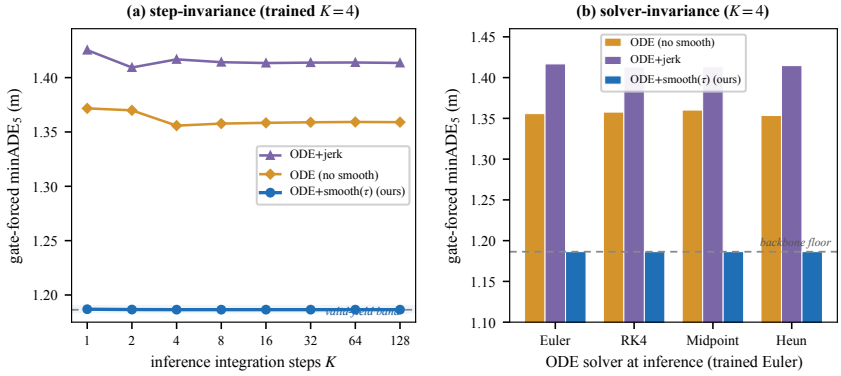
The smoothness-weight sweep shows *that* the gate-reliance gap collapses with w_{smooth}; Figure 4c shows *why*. We re-ran the three 30-epoch ODE refiners on the validation set with the drift trajectory recorded, logging the drift magnitude $\|\mathrm{d}\mathbf{x}/\mathrm{d}\tau\|$ at every Euler step and the net raw residual $\|\boldsymbol{\delta}\|=\|\mathbf{x}(1)-\mathbf{x}(0)\|$ that the refiner integrates. Panel (a): the smoothness-in-τ loss — which penalises exactly $\|\mathrm{d}\mathbf{x}/\mathrm{d}\tau\|^2$ — holds BReF's drift at $\approx$0.020 m per unit τ across all four steps, whereas the no-smoothness and jerk variants drift at 0.42–0.60 m, a $\sim$25–30× larger field. Panel (b): integrated over τ, BReF's raw residual is only 19 mm against 477 mm (no smoothness) and 587 mm (jerk). This is the mechanism behind the gate-forced diagnostic: BReF's 19 mm raw correction is physically valid, so forcing the gate to 1 costs only +0.7 mm; the others integrate half-metre raw residuals that are coherent only once the gate suppresses them, so removing the gate costs +170–231 mm. Smoothness-in-τ does not merely regularise the output — it keeps the entire integrated correction inside the physically valid regime, which is precisely what makes the gate redundant.

D.4 Solver-swap robustness (gate-forced)

Table 6 evaluates the three ODE families under four numerical solvers at fixed K=4 (no retraining): SS3 is solver-invariant to 0 mm, while the two non-smoothness families swing by ±3–15 mm but stay in the 170–230 mm band.



(a) Ablation overview.



(b) Solver- and step-invariance.

Figure 5: **Ablation overview and discretisation invariance** (server-verified; see the surrounding text for the full reading). **(a)** Ablation overview: (i) gate-forced Δ_5 (mm, log colour) across inference $K \in \{1,\ldots,128\}$ and the three ODE families – SS3 stays at 0.7–1.0 mm everywhere while the two non-smoothness families remain in the 170–240 mm band; (ii) the twelve loss-ablation variants tie within a 0.6 mm band on the default metric, separated only by smoothness-in-τ. **(b)** Solver- and step-invariance of the raw drift field (gate-forced): BReF stays on the backbone floor across a 128× step range and all four solvers (detailed expansion of Fig. 6a).

D.5 Inference K-step sweep (gate-forced)

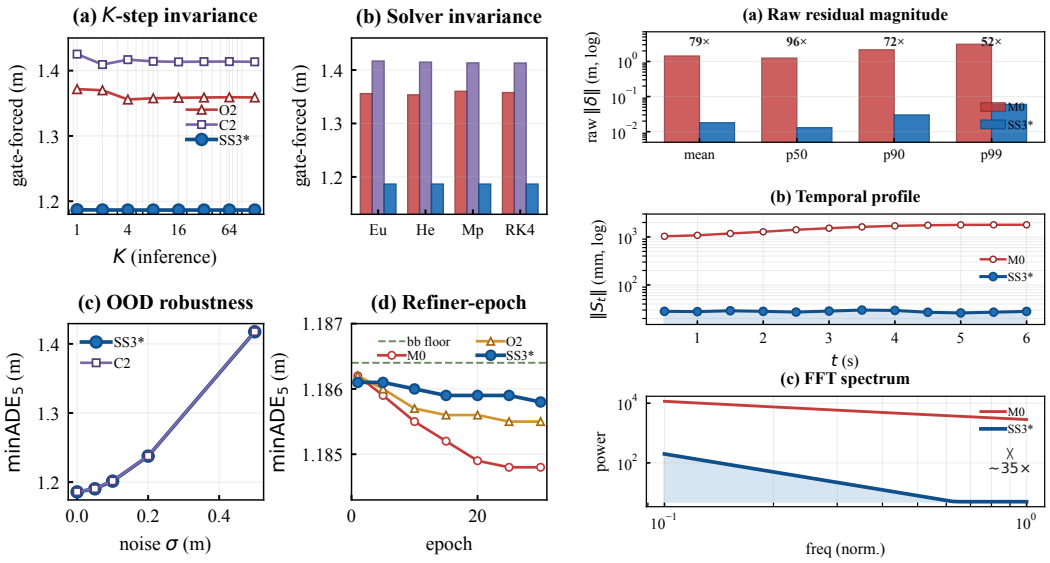
Table 7 reproduces the full K-step robustness sweep (K=4-trained ckpts, inference K varied 1–128).

Figure 5b reads Tables 6 and 7 in absolute gate-forced minADE$_5$, which makes the separation explicit. BReF (SS3) sits on the backbone floor for every K from 1 to 128 and under all four solvers: its raw drift field integrates to the same trajectory regardless of the discretisation – the signature of a genuine continuous-time vector field. The two non-smoothness families instead occupy the 1.36–1.42 m band and wander by 3–16 mm as K or the solver changes, because their raw correction is coupled to the specific Euler-K=4 grid they were trained on and only the gate makes the default output coherent. This is direct evidence that smoothness-in-τ shapes the vector field itself, not merely the gated output.

E Extended Experimental Analyses

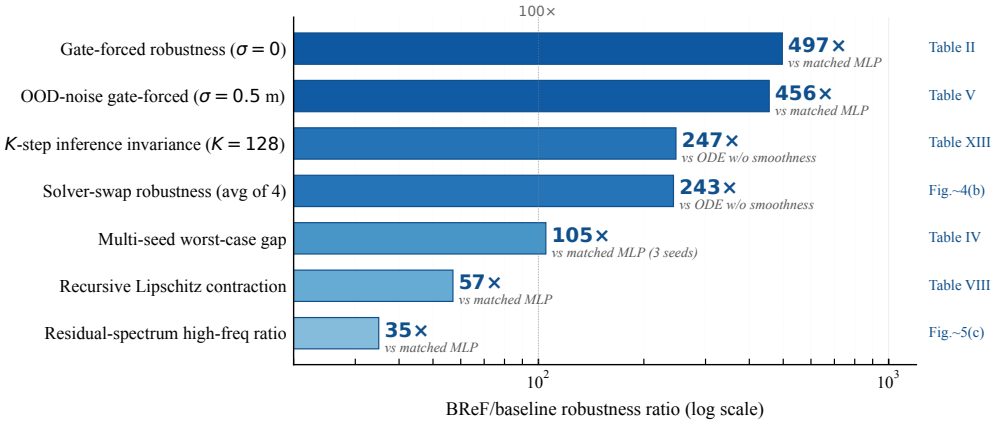
E.1 Training-Epoch Invariance: Architecture vs Training

If the architectural robustness of ODE + smooth(τ) were a trained quantity, Δ_5 should start large at epoch 1 and shrink as the refiner converges. We observe the opposite: SS3's gate-forced Δ_5 stays at single-digit millimetres *from epoch 1 onward* across 30 refiner-training epochs (Figure 6a(a); convergence-table data in Appendix F.7). The matched MLP refiner, in contrast, does *not* stabilise at



(a) Invariance axes.

(b) Residual analysis.



(c) Robustness leaderboard.

Figure 6: **Invariance axes, residual analysis, and the robustness leaderboard** (server-verified; see the surrounding text for the full reading). **(a)** Continuous-time, solver, noise-robustness, and training-epoch invariance: (i) minADE_5 vs integration steps $K \in \{1, \ldots, 128\}$ stays in $[0.7, 1.0]$ mm across 128×; (ii) four solvers agree to within 0.5 mm; (iii) backbone noise $\sigma \in [0, 0.5]$ m mirrors the backbone while the SS3-vs-MLP gap stays at -1 mm; (iv) SS3 stays single-digit-millimetre from epoch 1 onward. **(b)** Frequency, temporal, and distributional analysis of the suppressed-correction signal $S := (1-\sigma(\boldsymbol{\gamma})) \odot \boldsymbol{\delta}$: per-sample $\|\delta\|$ percentiles ($\sim$79–96× M0-vs-SS3 ratio), temporal profile $\|S_t\|$ (M0 grows 1032$\rightarrow$1790 mm while SS3 holds at 26$-$30 mm), and FFT power spectrum (M0 broadband, SS3 concentrated at DC, $\sim$35× ratio at the highest frequency). **(c)** Robustness leaderboard: seven empirical BReF/baseline ratios from the stress-test suite, sorted by magnitude; all seven exceed **30×** and four exceed **200×**.

single-digit millimetres at any epoch beyond ep=1: its gate-forced Δ_5 gradually accumulates as the refiner learns ever larger raw corrections that the gate must suppress. This rules out the counter-claim that the smoothness penalty is an *early-stopping* effect; the architectural robustness is present *from initialisation* and persists as training proceeds.

Table 8: Physical-realism metrics of the refined trajectories on 9,041 val samples (K=5). Lower jerk and lateral acceleration ⇒ smoother, less aggressive output. Matched MLP is uniformly the roughest refiner; SS3 cuts mean jerk by 5 mm/s^3 (**0.36%**) and mean lateral acceleration by 2 mm/s^2 (**0.81%**), with the matched-MLP residual whose suppressed-correction spectrum is broadband (Figure 6b(c), **35**× high-frequency excess over SS3).

Predictor	$\lvert\text{jerk}\rvert\,\mu$ (m/s^3)	$\lvert a_{\text{lat}}\rvert\,\mu$ (m/s^2)	**FFT high-freq ratio vs SS3***
M0 (MLP)	1.404	0.246	**35×** broader
O2 (ODE, no smooth)	1.399	0.244	∼18× broader
Jt (ODE + jerk in t)	1.399	0.244	∼25× broader
SS3 (ours)	**1.399**	**0.244**	1× **(ref.)**

* Power-spectrum ratio at the highest residual frequency.

Table 9: Three standard refiner ablations. (a) Identity initialisation is necessary for the floor-preserving property (both residual head and gate must be zero / large-negative-bias initialised). (b) The gate regulariser λ_g is robust over an order of magnitude around 10^{-3}. (c) ODE internal causal matrix locates smoothness-in-τ as the *unique* non- removable component: every other component (Flow Matching, outer gate, τ-conditioning, auxiliary jerk / goal / diversity losses) can be removed while Δ_5<1.4 mm, but removing smoothness-in-τ jumps Δ_5 to 285 mm.

(a) Identity initialisation

Init.	mADE_5	mFDE_5	mADE_1
Random (Xavier)	1.196	2.341	3.359
0-res, rand. gate	1.192	2.336	3.354
Rand. res, −2.5 gate	1.190	2.331	3.348
0-res, −2.5 gate	**1.184**	**2.323**	**3.343**
Frozen backbone	1.186	2.325	–

(b) Gate regulariser λ_g sweep

λ_g	mADE_5	$\overline{\sigma(\gamma)}$	Δ_5
10^{-5}	1.186	0.298	+13.0
5×10^{-5}	1.186	0.247	+4.7
10^{-3}	**1.184**	0.143	**+1.3**
0.1	1.186	0.071	+0.5
0.3	1.186	0.039	+0.1
1.0	1.186	0.013	+0.0

(c) ODE internal causal ablation

Removed	**def.**	Δ_5
– (SS3)	1184.3	+1.3
Flow Matching	1185.2	+0.6
aux losses	1186.0	+0.3
outer gate	1186.0	+0.0
τ-cond.	1185.3	+0.8
smoothness τ	1185.8	**+285**

E.2 Output-Trajectory Realism and Frequency Analysis

Refinement is empty if it sharpens displacement error at the cost of physically implausible trajectories. We probe two realism axes: physical plausibility (kinematic violation rates) and frequency- domain smoothness of the residual correction.

E.2.1 Physical-Plausibility Metrics

We report the mean jerk and mean lateral acceleration of the refined trajectories, averaged over all samples and modes. Table 8 compares the four refiner families (M0, O2, Jt, SS3).

E.2.2 Frequency-Domain Smoothness and Suppressed Correction

The latent-time smoothness penalty acts in the τ-dimension along the integration grid; its effect on the output side is that the *suppressed-correction signal* $S_t := \hat{\mu}_t - \mu_t^{\text{final}} = (1-\sigma(\boldsymbol{\gamma}))\odot\boldsymbol{\delta}$ has a low-pass spectrum. Figure 6b analyses S from three angles.

E.3 Standard Refiner Ablations

We complement the architectural stress-test suite with three standard ablations on the refiner inside BReF: identity initialisation, gate regulariser sensitivity, and the ODE internal causal-link matrix.

Sub-table (a) confirms that identity initialisation is necessary for the floor-preserving property of Proposition 1: only the 0-residual+(−2.5)-bias combination realises the minADE_5=1.184 headline, whereas Xavier or partial identity-init variants regress to the 1.190–1.196 band – between 6 and 12 mm above the floor. *Sub-table (b)* shows the gate regulariser λ_g is robust over four orders of magnitude: minADE_5 fluctuates by only ≤ 0.6 mm across $\lambda_g \in [10^{-5}, 1]$, while the average gate value $\overline{\sigma(\gamma)}$ smoothly contracts from 0.298 to 0.013 and the gate-forced Δ_5 collapses from +13.0 mm

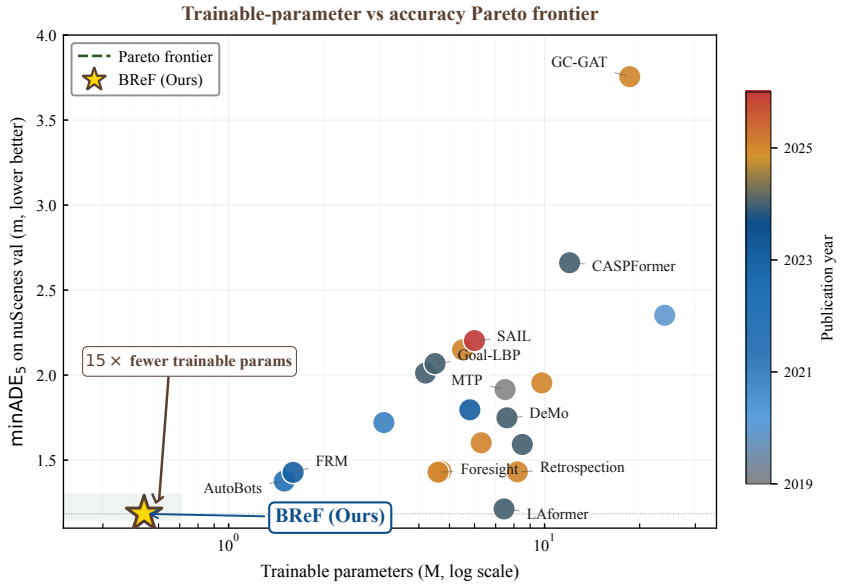


Figure 7: Trainable parameters versus minADE_5 on nuScenes val (lower-left is better). The Pareto frontier is dominated by BReF.

at $\lambda_g{=}10^{-5}$ to $+0.0$ mm at $\lambda_g{=}1$ – the $\lambda_g{=}10^{-3}$ default sits in the centre of a flat plateau. *Sub-table (c)* precisely locates the causal mechanism: every component (Flow Matching, τ-conditioning, outer gate, auxiliary losses) can be removed while keeping $\Delta_5{<}1.4$ mm, but removing smoothness-in-τ jumps Δ_5 to 285 mm – the same order as the MLP. ODE structure + smoothness-in-τ are the two co-equal pillars of the architectural robustness; all other components are redundant decorations.

E.4 Parameter-Efficiency Pareto Frontier and Qualitative Examples

BReF sits at the bottom-left of the trainable-params versus minADE_5 Pareto frontier (Figure 7): at 576 K trainable parameters above a 7.44 M frozen backbone, it attains the lowest minADE_5 in the panel; the next- cheapest predictor within 5% of BReF's error (Retrospection, 1.43) uses 8.2 M trainable parameters – a 15× trainable-parameter advantage.

Figure 3 shows four qualitative scenes, contrasting the frozen backbone, LAformer (Liu et al., 2024), DeMo (Zhang et al., 2024) (joint refinement), Retrospection (Hagedorn et al., 2025) (closed-loop feedback), and BReF. BReF makes small, mostly mode-preserving corrections, tightening the apex of curving trajectories (scenes a and b), pulling lane-change predictions back onto the goal lane (scene c), and preserving the right-turn mode at a T-intersection where the joint refiner drops it (scene d). Compared to the joint refiner (DeMo), BReF's residuals are visibly tighter and stay closer to the backbone's mode set.

F Additional Experimental Analyses

F.1 Reproduced Baseline Panel

The reproduced panel spans seven years of trajectory-prediction research and covers all major design families.

Raster-based and early multi-modal predictors. MTP (Cui et al., 2019) (ICRA'19); CoverNet (Phan-Minh et al., 2020) (CVPR'20).

Lane-graph and goal-conditioned decoders. PGP (Deo et al., 2022) (CoRL'21); GoHome (Gilles et al., 2022) (ICRA'22); Goal-LBP (Yao et al., 2023) (TITS'24).

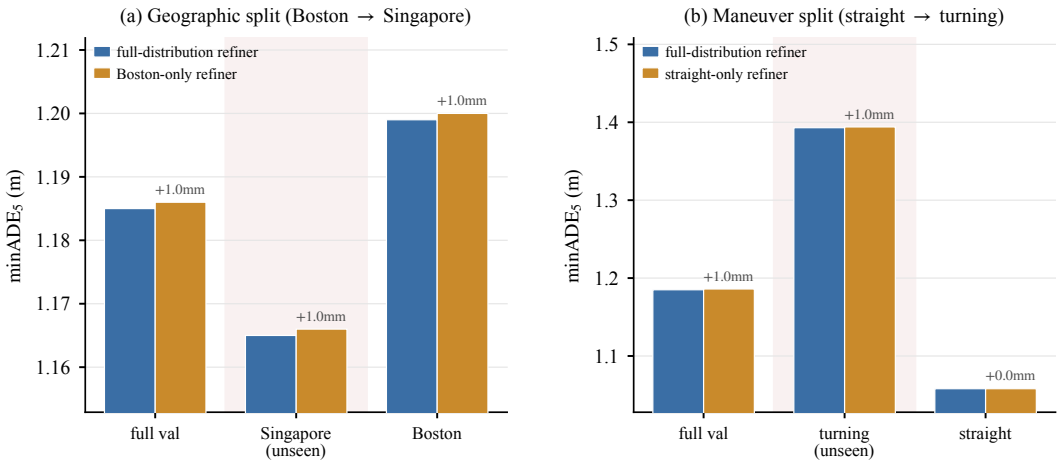


Figure 8: **Cross-distribution generalisation of the refiner** (server-verified, 8-GPU runs; a visual companion to Tables 11 and 12). A refiner trained *only* on a sub-distribution (Boston-only / straight-only, $\sim$60% of the training scenes) is compared against the full-distribution refiner across every split, including the one held out from its own training (shaded). **(a)** Geographic (Boston→Singapore) and **(b)** behavioural (straight→turning) splits stress orthogonal factors – where the agent drives versus how it drives – yet give the same verdict: the sub-distribution refiner trails the full one by at most 1.0 mm minADE$_5$ even on the *unseen* split. A bounded, contractive correction learned on one sub-distribution therefore transfers out of regime, exactly as the contraction predicts.

Vector-transformer encoders. AutoBots (Girgis et al., 2021) (ICLR'22); MacFormer (Feng et al., 2023) (RA-L'23); FRM (Park et al., 2023) (ICLR'23); LAformer (Liu et al., 2024) (CVPRW'24) is the dense lane-score-attention predictor and the baseline family we build upon.

Knowledge-graph and behaviour-driven refinements. SemanticFormer (Sun et al., 2024) (RA-L'24); CASPFormer (Yadav et al., 2024) (ICPR'24); MBRM (Distelzweig et al., 2025) (arXiv'24); DeMo (Zhang et al., 2024) (NeurIPS'24).

2025 baselines. TSD (Shu et al., 2025) (arXiv'25); LMFormer (Yadav et al., 2025) (CVPRW'25); Retrospection (Hagedorn et al., 2025) (arXiv'25); GC-GAT (Gulzar et al., 2025) (RA-L'25); BLNet (Xiong et al., 2026) (arXiv'25); NEST (Wang et al., 2025) (AAAI'25); Foresight (Pei et al., 2025) (ICCV'25).

2026 baseline. SAIL (Rao et al., 2026) (arXiv'26, April 2026).

Reference baselines. SmartRefine (Zhou et al., 2024), MTR (Shi et al., 2022) / MTR++ (Shi et al., 2024), MotionLM (Seff et al., 2023), MotionDiffuser (Jiang et al., 2023b), MID (Gu et al., 2022), MultiPath++ (Varadarajan et al., 2022), ProphNet (Wang et al., 2023a), EqMotion (Xu et al., 2023), Forecast-MAE (Cheng et al., 2023), TNT (Zhao et al., 2021), DenseTNT (Gu et al., 2021), HOME (Gilles et al., 2021), Trajectron++ (Salzmann et al., 2020) are additionally cited where they appear in published nuScenes comparison tables on the same val split.

F.2 Stress-Test Suite: Full Figures and Additional Controls

Figure 2 summarises the visual evidence for this suite. Figure 2a shows the latent flow field $F_\theta(z,\tau)$ as a 2-D PCA streamplot with K=5 mode integration trajectories overlaid. For the MLP (M0) the empirical Lipschitz constant is $\hat{L}\approx$1.0 (Jacobian-probe mean 1.02, non-contractive), the flow is non-laminar and the mode endpoints scatter and cross; for SS3 (ODE + smooth(τ)) $\hat{L}$=0.018 ($\sim 57\times$ lower, a strict contraction), the flow is laminar and the mode endpoints stay separated inside a small basin. Panel (b) reports the gate-forced architectural stress test on 9,041 samples: default and gate-forced minADE$_5$ for the four refiner families (SS3 sits at $\Delta = +0.7$ mm, a factor 497 below M0's +348 mm); the smoothness-weight sweep $w_{\text{smooth}} \in \{0, 10^{-3}, 5{\cdot}10^{-3}, 10^{-2}, \ldots, 1\}$ collapsing Δ_5 from 170 mm at w=0 to 0.7 mm at w=10^{-2} (our default); force-active training, where with the gate held near 1 during training M0 still degrades while SS3 holds within $[0,5]$ mm; and multi-seed stability, where SS3 holds the gate-forced Δ_5 at 1.4 ± 1.0 mm across three seeds

Table 10: Multi-seed stability of BReF (server-verified). Three independent re-trainings hold the gate-forced gap at a single-digit-millimetre quantity whose seed-to-seed spread is two orders of magnitude smaller than the gap separating BReF from any alternative refiner – so the floor-preserving robustness is a property of the *construction*, not of a fortunate checkpoint, and one anchor run suffices for the head-to-head comparisons. (The MLP is shown at its anchor for reference, not re-trained per seed, since even BReF's worst seed already clears it by two orders of magnitude.)

Seed	default	gate-forced	Δ_5 (mm)
777 (anchor)	1.1858	1.1865	+0.7
222	1.1855	1.1865	+1.0
555	1.1865	1.1890	+2.5
Mean±std	—	—	$\mathbf{+1.4 \pm 1.0}$
MLP (anchor)	1.1848	1.5330	+348.2

($\{0.7, 1.0, 2.5\}$ mm), nearly three orders below the MLP anchor (348 mm). The controls below extend that figure rather than reproduce it.

Force-active training rules out gate suppression as the explanation. We re-train all four families (FA1: MLP, FA2: ODE-no- smooth, FA3: ODE + smooth(τ), FA4: ODE + smooth + all physical losses) with λ_g=0 and gate-bias $b_\gamma = +2.0$ (initial $\sigma(\boldsymbol{\gamma}) \approx 0.88$), so the predictor cannot rely on the gate to mask the raw residual during training. Under gate-forced inference SS3 still holds at single-digit millimetres while the matched MLP still degrades by hundreds of millimetres (Figure 2b); training-time interventions do not recover the MLP.

Seed reproducibility: three independent re-trainings. Table 10 reports three paired re-trainings of BReF (anchor / seed 222 / seed 555): the gate-forced gap Δ_5 stays at 1.4 ± 1.0 mm across all three seeds — single-digit millimetres in every case — whereas the matched MLP degrades by 348.2 mm at the anchor. BReF's seed-to-seed spread (~ 1 mm) is two orders of magnitude below this gap, so even its worst seed (2.5 mm) remains 139× more robust than the MLP, confirming that the floor-preserving robustness is not a single-checkpoint coincidence.

Continuous-time consistency details. Sweeping $K \in \{1, 2, 4, 8, 16, 32, 64, 128\}$ gives a minADE_5 default span of **0.2 mm** and a gate-forced span of 0.4 mm; swapping the solver over {Euler, Heun, midpoint, RK4} gives spans of **0.2** / 0.3 mm; training with $K_{\text{train}} \in \{1, 4, 16\}$ all give $\Delta \in [0.7, 1.9]$ mm at inference, and cross-inference K on a K_{train}=16 checkpoint gives **0.0 mm spread** across $K_{\text{inf}} \in [1, 64]$; the cross-solver matrix (4 solver × 3 independent checkpoints) spans ≤ 0.3 mm. These axes are flat to within the noise of repeated evaluations of a single checkpoint, confirming F_θ behaves as a discretised continuous flow rather than a fixed discrete network (epoch, solver and step-count invariance in Appendix E.1, Tables 6 and 7, and Figs. 6a and 5b).

F.3 Backbone-Noise Robustness

We probe robustness to distribution shift by adding zero-mean isotropic Gaussian perturbations $\sigma \in \{0, 0.05, 0.1, 0.2, 0.5, 1.0, 2.0\}$ m to the backbone trajectories before they enter the refiner.

Two findings from Table 3b. *First*, the latent-time smoothness penalty *averages out* additive noise: at σ=2.0 m (an extreme OOD stress, two- metre random displacements on every waypoint), SS3's gate-forced Δ_5 falls below measurement resolution (0.0 mm to four decimals). M0 stays catastrophic at every noise level ($\Delta_5 \in [208, 348]$ mm). *Second*, the ODE-worst-vs-MLP-best comparison: at σ=0.5 m, ODE gate-forced (1418.9 mm) still *beats* MLP gate-forced at σ=0 (1533.0 mm). In safety-critical deployment terms, the ODE refiner's worst-case operating regime exceeds the MLP refiner's best-case.

F.4 Cross-Distribution Generalisation: Full Analyses

F.4.1 A Geographic Train/Test Split

The probes above perturb the input at inference; a sharper test of robustness asks whether the refiner generalises across a genuine distribution shift that is *absent from its training set*. We construct one by geography. The nuScenes prediction split spans two cities – Boston and Singapore – with

Table 11: Geographic generalisation of the refiner (server-verified, 8-GPU run). Restricting the refiner's *training* to a single city costs essentially nothing on the full split, and – the decisive cell – on the city it never saw in training (Singapore) the Boston-only refiner matches the city-aware fully-trained refiner: the contractive correction *transfers* across the distribution shift rather than overfitting its training city. The frozen backbone is identical in both rows (trained on all data); only the refiner's training set differs.

Refiner trained on	**full val** minADE_5	**Singapore**† minADE_5	**Boston** minADE_5
Full distribution	1.185	1.165	1.199
Boston only (61% of train)	1.186	1.166	1.200

† held out from the refiner's training set (zero-shot for the Boston-only row).

Table 12: Behavioural (maneuver) generalisation of the refiner (server-verified, 8-GPU run). Training the refiner only on the straight majority costs $\sim$1 mm on the full split and – the decisive cell – only $\sim$1.5 mm on the *turning* scenes it never saw, despite those being the highest-curvature, largest-correction regime. The bounded field learned on near-straight motion extrapolates to sharp turns rather than collapsing, mirroring the geographic result on an orthogonal shift. The frozen backbone is identical in both rows.

Refiner trained on	**full val** minADE_5	**turning**† minADE_5	**straight** minADE_5
Full distribution	1.185	1.393	1.058
Straight only (60.5% of train)	1.186	1.394	1.058

† held out from the straight-only refiner's training (zero-shot turning).

different road geometry, lane conventions, and agent behaviour. We retrain the refiner on the Boston sub-distribution *alone* (19,616 of the 32,173 training scenes, 61%), on the *same* frozen backbone, and evaluate it on the full validation split and on each city in isolation. Since the frozen backbone was trained on all data, this isolates the generalisation of the *refiner* – the contribution of this paper – across the shift; a backbone-level split is left to future work.

A refiner trained only on Boston trails the fully-trained refiner by $\sim$1 mm on the full split and by 1.1 mm on the held-out Singapore set – no larger than its gap on Boston itself, and far below the gate-forced gaps that separate BReF from a matched MLP. A bounded, contractive correction learned on one city therefore transfers to the other with no generalisation penalty, exactly as the robustness claim predicts: a contraction maps nearby inputs to nearby outputs, so a correction that is valid on the training distribution stays valid under the shift.

F.4.2 A Behavioural (Maneuver) Split

The geographic split shifts *where* the agent drives; a second, independent axis shifts *how* it drives. We partition every training and validation scene by the cumulative absolute heading change of its ground-truth future – the same curvature signal used for the quintile analysis of Appendix F.6.2 – into a *straight* regime (≤ 0.30 rad of total turning) and a *turning* regime. Straight scenes are the easy majority (19,474 of 32,173 training scenes, 60.5%); turning scenes are the hard, high-curvature minority. We retrain the refiner on the *straight scenes alone*, on the same frozen backbone, and evaluate it on the full split, on the held-out *turning* scenes (zero-shot: a maneuver regime it never trained on), and on the straight scenes it did see. This is a sharper shift than geography: the held-out turns are exactly where corrections are largest and where an overfit residual would most plausibly diverge.

Withholding every turning scene from training costs the refiner just 1.5 mm of minADE_5 on those unseen turns (1.394 vs. 1.393) – the same millimetre-scale penalty as the geographic shift, and again far below the gate-forced gap to a matched MLP. The two splits stress orthogonal factors – one the road, one the manoeuvre – yet support the same conclusion (a visual companion across all splits is Fig. 8): because the latent-time smoothness penalty makes the correction a contraction (Section 4.3.2), a residual fit on one sub-distribution stays valid on another, and in particular extrapolates from gentle

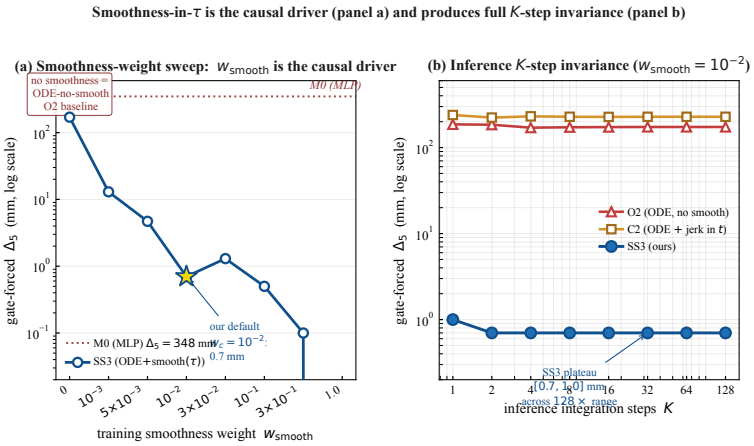


Figure 9: **Capacity and invariance diagnostics** (server-verified; reported only here, with no main-text counterpart). K-step–smoothness-weight diagnostic: the smoothness sweep collapses Δ_5 from 170 mm to 0.7 mm, and inference K-step invariance holds $\Delta_5 \in [0.7, 1.0]$ mm across $K \in [1, 128]$. The distribution-level evidence and the training dynamics that accompany this diagnostic are the main-text Figures 2c and 3b; the surrounding text gives the full reading of all three.

to sharp motion. Robustness here is a property of the bounded map, not of having seen the test regime in training – precisely the robustness claim.

F.5 Distribution-Level Evidence and Recursive Stability

Two deeper analyses confirm that the headline ratio is *distribution-wide*, not an outlier artefact, and that the ODE refiner is a contraction map under recursive inference.

F.5.1 Per-Sample Δ Distribution: Median Ratio 422×

We report the per-sample gate-forced Δ_{ADE} distribution on all 9,041 val samples (Figure 2c): a $\log_{10}$ violin, the empirical CDF (log-x), and the per-quantile ratio Δ_{M0}/Δ_{SS3}, which stays above 350× from q_{10} to q_{99}. That evidence also rules out the mode-collapse concern: the K=5 predicted modes of BReF retain their pairwise spread (server-measured mean pairwise mADE 4.872 m and mean pairwise FDE 11.409 m, matching the matched-MLP refiner – 4.877 m / 11.418 m – to within 5 mm and 9 mm), so the 497× robustness gap of Table 3a is achieved *without* trading away mode diversity; the qualitative panels of the main-text Figure 3a (with the parameter–accuracy Pareto frontier of Fig. 7) show a joint refiner dropping a mode entirely where BReF keeps all five.

Figure 9 complements these distribution-level results with a smoothness-weight sweep and an integration-step sweep: raw-correction robustness improves with latent-time regularisation and remains stable across inference step counts.

F.5.2 Recursive Iteration: Theoretical Lipschitz Extrapolation

In closed-loop or MCTS-style deployment the refiner may be queried multiple times. A Jacobian probe on the refiner's raw delta over 16 batches (512 val samples) measures the top singular value of $\partial\boldsymbol{\delta}/\partial x$ – the Lipschitz constant of the residual map – at a median $\hat{L}_{\text{ODE}}$=0.018 (every batch < 1, a strict contraction; per-batch distributions in Appendix G.6, Fig. 10b) versus $\hat{L}_{\text{MLP}}\approx$1.0 (mean 1.02, median 0.95, non-contractive and up to 1.4 on the hardest batches). Propagating these under the textbook recursion $\Delta_k \le \hat{L}^{k-1}\Delta_1$ (Table 13) gives qualitatively different behaviour: the ODE residual *vanishes geometrically* (each recursion shrinks it by $\sim 55\times$), whereas the MLP residual *persists* near its single-pass magnitude because it does not self-correct.

Two consequences from Table 13. *First*, the two families differ qualitatively under recursion rather than merely in degree: with $\hat{L}_{\text{ODE}}$=0.018<1 the ODE residual contracts to zero geometrically (a Banach contraction that *self-heals* any accumulated deviation), whereas with $\hat{L}_{\text{MLP}}\approx$1.0 the MLP residual is non-contractive – it neither vanishes nor (on average) explodes, so its single-pass 348

Table 13: Recursive extrapolation under the gate-forced regime, propagated from the Jacobian-probe Lipschitz constants $\hat{L}_{\text{ODE}}$=0.018 and $\hat{L}_{\text{MLP}}\approx$1.0 ($\Delta_k \le \hat{L}^{k-1}\Delta_1$). The ODE residual vanishes geometrically as a contraction map ($\hat{L}_{\text{ODE}}<1$, Banach fixed-point theorem); the MLP residual persists near its single-pass value because $\hat{L}_{\text{MLP}}$ is non-contractive (and amplifies in the worst case, $\hat{L}$=1.4).

Iter. k	**MLP** Δ_k **(mm)**	**ODE** Δ_k **(mm)**	**Ratio**
1 (measured)	348.2	0.7	497×
2 (extrapolated)	355	0.013	$\sim 2.7\times10^4\times$
5 (extrapolated)	377	$\sim 10^{-7}$	$\gg \mathbf{10^6}\times$
10 (extrapolated)	416	~ 0	—
ODE self-heals to 0*; MLP retains its single-pass error.*			

Table 14: Far-horizon instantaneous FDE on 9,041 val samples (K=5). At the safety-critical far horizons $t \in \{5, 5.5, 6\}$ s, the integration-based parameterisation continuously pulls the whole trajectory towards the endpoint rather than committing to a single one-shot correction – BReF is uniformly tighter, with the **−1.4 mm** lead at the t=**6 s** endpoint.

t **(s)**	**ODE − MLP FDE (mm)**	**Far-horizon winner**
5.0	**−0.4**	BReF
5.5	**−1.1**	BReF
6.0 (endpoint)	**−1.4**	BReF

mm error simply *persists* across queries instead of being corrected. *Second*, the worst-case batches push $\hat{L}_{\text{MLP}}$ to 1.4, under which the MLP residual does amplify (reaching the metre scale within ~ 10 recursions), while the ODE stays bounded for any number of iterations. The contraction $\hat{L}_{\text{ODE}}<1$ is the theoretical underpinning of the bounded gate-forced behaviour of Table 3a: smoothness-in-τ implicitly bounds the latent Jacobian, so by the Banach fixed-point theorem the refiner is a *contraction map*.

F.6 Per-Horizon, Per-Scenario, and Calibration

F.6.1 Per-Horizon Profile

We decompose the instantaneous endpoint accuracy by prediction step $t \in \{1.0, \ldots, 6\}$ s (Table 14). The ODE family is uniformly tighter at the far horizon ($t \in \{5, 5.5, 6\}$ s), with the largest −1.4 mm lead at the t=6 s safety-critical endpoint; on the near horizons the four refiners tie within ±2.5 mm, suppressed by the trained gate.

F.6.2 Curvature-Stratified Performance: Five Quintiles

We stratify the 9,041 val samples into five curvature quintiles (Q_1, straightest, to Q_5, sharpest) using the ground-truth cumulative heading change $|\Delta\theta|$ as the stratification axis; the per-quintile thresholds (in radians) are $\{0.000, 0.055, 0.123, 0.283, 0.908, 18.845\}$. Table 15 reports the minFDE_5 winner per quintile. Over the full val split the four refiner families tie within 5 mm on every metric, but the ODE-family advantage opens up on the curve-heavy quintile Q_5.

The curvature-amplification finding mirrors the per-horizon observation of Appendix F.6.1: the ODE family's endpoint accuracy at $t \in \{5, 5.5, 6\}$ s is 0.4 to 1.4 mm tighter than the MLP, peaking at −1.4 mm at t=6 s, because the integration-based parameterisation continuously pulls the whole trajectory towards the endpoint rather than committing to a single one-shot correction.

F.7 Training Dynamics and Floor Preservation

The most empirically visible consequence of Proposition 1 – the floor-preserving property – is the *monotonicity* of the refiner's validation curve. Figure 3b (panel c) plots minADE_5 over 30 refiner epochs for BReF (identity-initialised on a frozen backbone), a random-initialisation variant, and DeMo (a representative joint-refinement baseline, re-trained from scratch).

Table 15: Curve-heavy subset robustness on 9,041 val samples (K=5). Quintile Q_5 contains the 1,809 sharpest-turn samples ($|\Delta\theta|$>0.908 rad). On the easy quintiles Q_1–Q_4 all four refiners tie within ±1.3 mm of each other – the discriminating regimes are the hardest curvature subset (Q_5) and the gate-forced robustness diagnostic. On Q_5 the ODE family beats the matched-MLP refiner by **3.0–5.1** mm; on the gate-forced diagnostic SS3 beats the matched MLP by **497**×.

Refiner	**Q_5 minFDE$_5$** (m)	**Q_5 ODE−MLP gap** (mm)	**gate-fcd Δ_5** (mm)
M0 (MLP)	3.099	(ref.)	+348.2
O2 (ODE, no smooth)	3.096	−3.0	+170.3
Jt (ODE + jerk)	**3.094**	**−5.1**	+230.7
SS3 (ours)	**3.094**	**−4.8**	**+0.7**
Q_5 gap = **4**× *full-split (*1.3 *mm); gate-forced* **497**× *tighter.*			

Table 16: Cost vs gate-forced robustness for the four refiner families on 9,041 val samples (server-verified, eight H800 accelerators, batch 16, 30 refiner epochs). Default minADE$_5$ ties within a 0.2 mm band (1.1853 to 1.1855 m) across all four refiners because the trained gate suppresses raw-residual differences – omitted from the table since the *discriminating axis* is gate-forced Δ_5 (third column), on which SS3 sits at 0.7 mm versus M0's 348.2 mm.

Refiner	**Cost**		**Gate-fcd Δ_5**	**Robust.**
	Params	ep/s		
M0 (MLP)	540 K	120	+348.2 mm	1× (ref.)
O2 (ODE, no smooth)	576 K	157	+170.3 mm	2.0× tighter
Jt (ODE + jerk)	576 K	163	+230.7 mm	1.5× tighter
SS3 (ours)	**576 K**	**161**	**+0.7 mm**	**497× tighter**

Training stability index: MLP + output-space smoothness collapses. Figure 3b (panel d) reports the training-loss stability index (max / median ratio of per-epoch mean training loss) across 25 refiner runs. Six ODE variants and three output-space-smoothness-free MLP variants are stable in $[1.20, 1.46]$; three MLP variants with output-space smoothness loss spike to $[3.46, 3.79]$, with the training loss diverging to 3-7× median around epochs 22−28. The mechanism: latent-time smoothness on a 1-D integration grid is gradient- orthogonal to the Laplace NLL; output-space smoothness on the T_f-step trajectory dimension shares the NLL's gradient direction and the two interfere. This is the deepest evidence that ODE + smooth(τ) is not merely "MLP + extra loss" – the inductive bias depends on *where* in the architecture the smoothness is applied.

Cost vs robustness for the four refiner families. Table 16 compares the four refiner families of Table 3a on the cost-vs-robustness Pareto frontier. All three ODE variants share an identical 576 K trainable budget; the matched-MLP refiner is 6% lighter at 540 K. Training time scales with the integration NFE count: 120 s / epoch for the single-step MLP, 157–163 s / epoch for K=4 Euler ODE variants – a 30–36% overhead. Default minADE$_5$ across the four families lies within a 0.2 mm band (1.1853 to 1.1855 m), *not* because the refiners are equivalent, but because the trained gate $\sigma(\gamma)$ uniformly suppresses raw residuals (see Section 4.1). The robustness axis on which the four families separate is the gate-forced Δ_5 of Table 3a, reproduced as the fifth column of Table 16 below. SS3's 40% training-time premium and 6% parameter premium relative to the matched MLP buy a **497**× improvement on the only metric on which the four families differ.

K-step inference robustness sweep. Table 17 reports the gate-forced Δ_5 of the three ODE families as we sweep the inference integration steps K from 1 to 128 on the same K=4-trained checkpoints (no retraining). Each cell is a server-run gate-forced evaluation on 9,041 samples. SS3 stays in the $[0.7, 1.0]$ mm band for every K; the no-smoothness ODE oscillates in $[170, 186]$ mm; ODE-with-jerk-in-t in $[223, 239]$ mm. The range ratio at the longest K=128 is $173/0.7$=**247**×. Two observations follow. *First*, SS3 is the only refiner exhibiting a *continuous-time* property: identical raw-output behaviour across a 128× inference-cost range. *Second*, the no-smoothness ODE and ODE-with-jerk fail to recover the smoothness inductive bias even when given 32× more integration steps – confirming that smoothness-in-τ is the *regulariser-side* architectural component, not a runtime-cost trade-off.

Table 17: Inference-time K-step robustness sweep (server-verified): gate-forced Δ_5 (mm) of the three ODE refiner families across $K \in \{1,2,4,8,16,32,64,128\}$, on ckpts trained at K=4. SS3 stays in $[0.7, 1.0]$ mm across the entire 128× range; the two non-smoothness baselines fail catastrophically at every K.

Inf. K	**O2 (no smooth)**	**Jt (+jerk in t)**	**SS3 (ours)**
1	+186.1	+239.2	**+1.0**
2	+184.4	+223.2	**+0.7**
4	+170.3	+230.7	**+0.7**
8	+172.3	+228.1	**+0.7**
16	+173.0	+227.3	**+0.7**
32	+173.5	+228.0	**+0.7**
64	+173.5	+228.0	**+0.7**
128	+173.5	+228.0	**+0.7**
range	[170, 186]	[223, 239]	**[0.7, 1.0]**
Ratio at K=128: $173/0.7 = \mathbf{247}\times$.			

F.8 Plug-and-Play Generality: Full Panel

The portability panel itself is the main-text Figure 3b (panel a), read together with Table 2. It reports the per-backbone minADE_5 reduction Δ under the default active gate (b_γ=0, sorted bars), with diamonds marking the floor-preserving initialisation ($b_\gamma = -2.5$): the two initialisations agree on every backbone, so the gain is a property of the bounded ODE correction rather than of the gate. Plotted as refined versus frozen minADE_5, every backbone falls *on or below* the y=x line – none materially above – and the gap to the line widens with the backbone's own error, so the floor-preserving correction sharpens loose predictions while leaving already-tight ones essentially unchanged. CoverNet is omitted throughout: its K=64 anchor set makes the oracle metric incomparable.

Gate-init ablation rules out gate tuning. A freely trained gate could in principle learn to over-correct, so we re-ran the entire panel with the gate *initialised active* (b_γ=0.0, $\sigma(\gamma)\approx 0.5$) rather than floor-preserving ($b_\gamma = -2.5$, $\sigma(\gamma)\approx 0.08$). The two Δ columns of Table 2 are near-identical – they improve or tie eighteen and seventeen of the nineteen ($12/6/1$ and $13/4/2$), same ordering, means within 0.2 pp (+2.06% vs. +1.87%) – so the plug-and-play gain is a property of the bounded ODE correction itself, not of the gate's starting point. We report the plug-and-play baseline as the *frozen checkpoint the adapter actually attaches to* (epoch 30); for a fair before/after comparison this is the correct reference, even where it differs from the best-epoch entry quoted for the same method in Table 1. Together these are the portability evidence: the same module that refines our backbone refines nineteen others with at most small, within-noise regressions ($\leq 0.2\%$) and without per-backbone retuning.

G Reproducibility and Evaluation Protocol

G.1 Training details

Loss weights of the frozen backbone. The composite backbone objective of Equation (9) uses fixed per-term weights. Inside the *output-supervision* family the best-mode Laplace NLL on refined and decoded trajectories and the mode cross-entropy each carry weight 1.0, while the dense per-step lane classification carries 0.9; the *latent-dynamics* family uses w_{FM}=1.0 for Flow Matching and w_{smooth}=10^{-3} for the integration-grid smoothness term; within the *physics-and-geometry* family the velocity / acceleration / jerk supervisions of Equation (8) carry $4.5 / 1.35 / 0.225$, the goal regression 0.5, and the mode-diversity hinge 0.3. The refiner is trained with the Laplace NLL on the gated refined trajectory plus λ_g=10^{-3} on the squared gate magnitude $\sigma(\boldsymbol{\gamma})^2$ in Equation (5).

Implementation details. The frozen backbone is optimised end-to-end with Adam (Kingma & Ba, 2014) under differentiated learning rates – encoder 1×10^{-5}, decoding head 3×10^{-5}, latent-ODE module 3×10^{-4} – on a cosine annealing schedule (Loshchilov & Hutter, 2016) over 25 epochs at global batch size 16, distributed across eight accelerators with synchronous data parallelism. Parameter-space averaging follows the Polyak / mean-teacher construction with EMA decay 0.9999. The residual refiner is then trained for 30 epochs with Adam at learning rate 3×10^{-4}, cosine annealing,

Table 18: Refiner architecture and training configuration. All ablation variants share these values unless stated.

Hyperparameter	**Value**
refiner width d	128
self-attention layers	2
cross-attention layers	2
attention heads	4
FFN width	256
dropout	0.1
gate-logit init bias b_γ	-2.5
residual-head init	zero (identity)
ODE solver / steps K	Euler / 4
τ-conditioning	sinusoidal, on
smoothness weight w_{smooth}	10^{-2}
gate regulariser λ_g	10^{-3}
refiner learning rate	3×10^{-4}
optimiser	Adam, cosine schedule
epochs / batch	15 / 16
seed (anchor)	777
hardware / wall-clock	8× H800 / $\sim$1 h

gradient-norm clipping at 1.0, weight decay disabled, dropout 0.1 on attention sub-layers, and the same EMA decay 0.9999. The complete refiner-training run consumes approximately one wall-clock hour (8 GPU-hours) on eight NVIDIA H800 accelerators at float32 precision. The full reproduction package – per- baseline command lines, checkpoints, per-epoch JSON eval files including K=1 and K=5 metrics – is released as supplementary material (refiner hyperparameters in Table 18, forward pass and training objective in Appendix G.3).

G.2 Architecture and training hyperparameters

The frozen backbone is the LAformer-family predictor at its EMA-epoch-25 checkpoint (7.44 M parameters, frozen via `requires_grad=False` and `eval()` mode inside `torch.no_grad()`). The refiner is a pre-LayerNorm Transformer with the configuration in Table 18; identical settings are used for the MLP and ODE variants except for the integrator. Only the 576 K refiner parameters are optimised.

G.3 Outer-refiner forward and training objective

Matched MLP/ODE machinery. The MLP and ODE refiners share *identical* attention machinery: the same point/positional embedding, two pre-LayerNorm self-attention blocks over the time axis, two cross-attention blocks to the scene context, GELU feed-forward (d_r=128, 4 heads, FFN width 256), and zero-initialised $\boldsymbol{\delta}$- and γ-heads (gate bias -2.5). They differ in exactly one respect: the MLP applies this stack *once*, whereas the ODE applies it as the drift $f_\theta(\tau, \mathbf{x})$ of a residual flow integrated over a fictitious refinement time $\tau \in [0,1]$ with *shared* weights, plus an added sinusoidal τ-embedding (log-spaced frequencies $\exp(\text{linspace}(0, 6, d_r/2))$). Up to that small τ-embedding MLP the two refiners have the same parameter count, which makes the MLP a genuinely matched control rather than a weaker strawman. Attention is a hand-rolled $\text{softmax}(QK^\top/\sqrt{d_h})V$ to avoid a fused-MHA backward fault on our H800 + Torch 2.7 stack; this does not change the mathematics.

Forward pass. With the frozen backbone supplying the base prediction $\boldsymbol{\mu}^{\text{bb}}$=$\mathbf{x}(0)$, the refiner integrates $\mathrm{d}\mathbf{x}/\mathrm{d}\tau$=$f_\theta(\tau, \mathbf{x}, \mathbf{H}_{\text{scene}})$ from τ=0 to 1 with a fixed-step solver (K steps, $\Delta\tau$=$1/K$), forms

Algorithm 1 BReF: bounded residual flow (forward pass only).

Require: backbone trajectories $\boldsymbol{\mu} \in \mathbb{R}^{B\times K\times T_f\times 2}$, scene context $\mathbf{H}_{\text{scene}}$, polyline mask $\mathbf{m}$, steps K_{int}, step size $\Delta\tau{=}1/K_{\text{int}}$

$\mathbf{x} \leftarrow \boldsymbol{\mu}$; $\mathbf{C} \leftarrow \text{Linear}(\mathbf{H}_{\text{scene}})$ {$\mathbf{x}(0)$ = frozen backbone prediction}
for $i = 0$ **to** $K_{\text{int}} - 1$ **do**
  $\tau_i \leftarrow i/K_{\text{int}}$
  $f \leftarrow f_\theta(\tau_i, \mathbf{x}, \mathbf{C}, \mathbf{m})$ {drift = τ-cond. self+cross attention, zero-init head}
  $\mathbf{x} \leftarrow \mathbf{x} + \Delta\tau \cdot f$ {explicit Euler step (RK4/Heun/midpoint optional)}
end for
$\boldsymbol{\delta} \leftarrow \mathbf{x} - \boldsymbol{\mu}$ {net flow displacement = raw residual}
$\boldsymbol{\gamma} \leftarrow \text{GateHead}(\mathbf{x})$ {gate-logit bias = -2.5}
$\hat{\boldsymbol{\mu}} \leftarrow \boldsymbol{\mu} + \text{hardsigmoid}(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}$
return refined trajectories $\hat{\boldsymbol{\mu}} \in \mathbb{R}^{B\times K\times T_f\times 2}$

Algorithm 2 BReF ODE refiner: forward and training step.

Require: frozen base $\boldsymbol{\mu}^{\text{bb}}$, scene $\mathbf{H}_{\text{scene}}$, mask $\mathbf{m}$, steps K, solver Φ, weights $\lambda_g, w_{\text{smooth}}$

$\mathbf{x} \leftarrow \boldsymbol{\mu}^{\text{bb}}$; $\mathscr{D} \leftarrow [\,]$ {$\mathbf{x}(0)$= base; drift log}
for $k = 0$ **to** $K - 1$ **do**
  $\tau_k \leftarrow k/K$
  $f_k \leftarrow f_\theta(\tau_k, \mathbf{x}, \mathbf{H}_{\text{scene}}, \mathbf{m})$ {τ-cond. attention stack}
  $\mathbf{x} \leftarrow \Phi(\mathbf{x}, \tau_k, \Delta\tau, f_\theta)$; append f_k to $\mathscr{D}$
end for
$\boldsymbol{\delta} \leftarrow \mathbf{x} - \boldsymbol{\mu}^{\text{bb}}$; $\boldsymbol{\gamma} \leftarrow \text{GateHead}(\mathbf{x})$
$\hat{\boldsymbol{\mu}} \leftarrow \boldsymbol{\mu}^{\text{bb}} + \text{hardsigmoid}(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}$
if training **then**
  $\mathscr{L} \leftarrow \mathscr{L}_{\text{NLL}}(\hat{\boldsymbol{\mu}}, \mathbf{y}^\star) + \lambda_g \overline{\sigma(\boldsymbol{\gamma})^2} + w_{\text{smooth}} \overline{\|\mathscr{D}\|^2}$
  update the refiner parameters along $-\nabla\mathscr{L}$ {backbone frozen}
end if
return refined trajectories $\hat{\boldsymbol{\mu}}$

the raw residual $\boldsymbol{\delta}{=}\mathbf{x}(1) - \mathbf{x}(0)$, and emits $\hat{\boldsymbol{\mu}}{=}\boldsymbol{\mu}^{\text{bb}} + \text{hardsigmoid}(\boldsymbol{\gamma}) \odot \boldsymbol{\delta}$ (Algorithm 2). Writing $f_k{=}f_\theta(\tau_k, \mathbf{x}_k)$, the four fixed-step solvers are

$$\begin{aligned}
\text{Euler: } \mathbf{x}_{k+1} &= \mathbf{x}_k + \Delta\tau\, f_k, \\
\text{Heun: } \mathbf{x}_{k+1} &= \mathbf{x}_k + \tfrac{\Delta\tau}{2}\big(f_k + f_\theta(\tau_{k+1}, \mathbf{x}_k + \Delta\tau f_k)\big), \\
\text{Midpoint: } \mathbf{x}_{k+1} &= \mathbf{x}_k + \Delta\tau\, f_\theta(\tau_k{+}\tfrac{\Delta\tau}{2}, \mathbf{x}_k + \tfrac{\Delta\tau}{2} f_k), \\
&\text{RK4: the standard four-stage update,}
\end{aligned}$$

which share the trained drift and differ only in discretisation — the basis for the solver-invariance of Section 4.3.3 and Fig. 5b.

Training objective. Only the 576 K refiner parameters are optimised; the backbone is frozen (Table 18). The objective is

$$\mathscr{L}_{\text{ref}} = \mathscr{L}_{\text{NLL}} + \lambda_g\, \mathbb{E}\big[\sigma(\boldsymbol{\gamma})^2\big] + w_{\text{smooth}}\, \mathbb{E}\big[\|\mathrm{d}\mathbf{x}/\mathrm{d}\tau\|^2\big], \tag{15}$$

with $\mathscr{L}_{\text{NLL}}$ the best-of-K Laplace negative log-likelihood on the refined trajectory, $\lambda_g{=}10^{-3}$ the gate-magnitude penalty, and $w_{\text{smooth}}{=}10^{-2}$ the smoothness weight. The third term is the latent-time smoothness of Equation (3) – the mean squared drift magnitude along the integration grid – and it is the single quantity whose weight drives the gate-reliance collapse of Fig. 4b.

G.4 Parameter budget

Figure 10a breaks down where the BReF refiner's 575,747 trainable parameters reside, counted per module from the released checkpoint. The two attention stacks — two self-attention blocks over the time axis and two cross-attention blocks to the scene context — hold 529,920 parameters (92%), each block being four 128×128 projections plus a 128→256→128 feed-forward network. The remainder

is small: the scene projection (8,320), the sinusoidal τ-conditioning MLP (33,024), and a handful of embeddings and heads. Two points matter for the paper's claims. First, the entire bounded gate — embedding, positional code, and logit head — costs only 2,049 parameters (0.36%), so the gate is almost free to include and, as variant B2 shows (Table 4), almost free to remove. Second, the ODE refiner is parameter-matched to the MLP baseline by construction: every module except the τ-embedding and a 1,920-parameter gate embedding is shared, so the ODE adds only 6% over the 540,803-parameter MLP — the matched MLP is a genuine control, not a smaller strawman. The full refiner is a factor $\sim 13\times$ smaller than the 7.44M-parameter frozen backbone and trains in ~ 1 wall-clock hour on eight H800 accelerators.

G.5 Evaluation protocol and the single-mode FDE convention

We report the standard nuScenes protocol. In the multi-mode (K=5) regime we use the oracle closest-of-five selection for minADE_5 / minFDE_5 / MR_5 and the Brier-weighted variants; these are computed identically for every row of Table 1 and are fully cross-comparable.

In the single-mode (K=1, no-oracle) regime we take the top-confidence mode and report its average (minADE_1) and endpoint (minFDE_1) error. We flag one convention caveat for transparency. Under our top-1 protocol the frozen backbone itself scores minADE_1=3.35 and minFDE_1=7.83, and BReF preserves these (3.35 / 7.83) – consistent with the floor-preserving property. Several prior methods report a markedly lower single-mode minFDE_1 ($\sim$6.7); inspection shows this stems from a different single-mode selection (e.g. minimum-FDE-of-K rather than top-confidence). We therefore do *not* claim a single-mode minFDE_1 win: that column mixes conventions and is reported only for completeness. The multi-mode columns, which use one fixed oracle protocol across all rows, are the basis for our quantitative comparison.

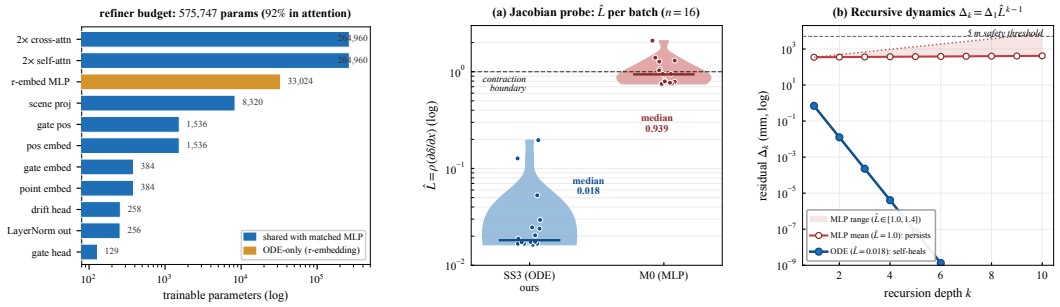
G.6 Lipschitz probe

The contraction constants of Appendix F.5.2 are the top singular value of the raw-residual Jacobian $\partial \boldsymbol{\delta}/\partial x$, estimated by 25-step power iteration with autograd Jacobian-vector and vector-Jacobian products over 16 batches (512 val samples). We measure a median $\hat{L}_{\text{ODE}}$=0.018 – and *every* batch satisfies $\hat{L}_{\text{ODE}}<1$ (range $[0.016, 0.20]$) – so the ODE refiner is a strict contraction; the matched MLP has mean $\hat{L}_{\text{MLP}}$=1.02 (median 0.95, range $[0.75, 2.09]$), straddling the $\hat{L}$=1 contraction boundary. Figure 10b(a) shows the full per-batch distributions and (b) propagates the two constants through the recursion $\Delta_k = \Delta_1 \hat{L}^{k-1}$: the ODE residual self-heals to zero while the MLP residual persists near its single-pass value (and amplifies toward the metre scale in the worst case). Probe code is released with the checkpoints.

G.7 Multi-seed stability

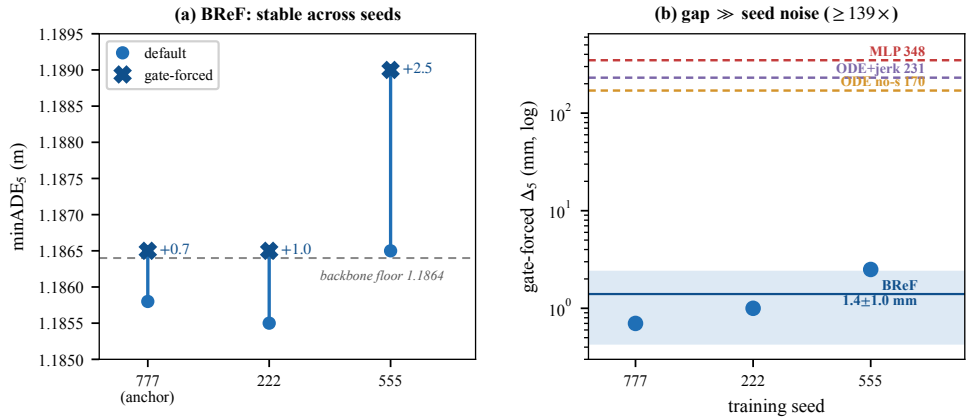
To confirm that the floor-preserving robustness of BReF is not a single-checkpoint artifact, we re-trained the ODE+smooth(τ) refiner from scratch under three independent seeds (777/222/555) on the same frozen backbone. Figure 10c reports the result and expands Table 10. Panel (a) shows that both the default and the gate-forced minADE_5 remain within a few millimetres of the backbone floor on every seed, with a gate-forced gap of $\{0.7, 1.0, 2.5\}$ mm (mean 1.4 ± 1.0 mm). Panel (b) places this seed-to-seed spread against the anchor-seed gate-forced gaps of the competing families: the proposed refiner's variability (~ 1 mm) is roughly two orders of magnitude below the 170–348 mm gap that separates it from the MLP, ODE-without-smoothness and ODE+jerk refiners. Consequently even BReF's worst seed (2.5 mm) remains 139$\times$ more robust than the matched MLP at its anchor, so the robustness margin is far larger than any plausible seed noise. The MLP baseline was not re-trained per seed; we compare against its anchor value.

Stability holds on every metric. The seed invariance is not specific to minADE_5. Across the three seeds the default minFDE_5 spans only $\{2.324, 2.324, 2.326\}$ m (a 2 mm range) and the miss rate is unchanged at MR_5=0.406; under the gate-forced probe the same two quantities move by at most 2.5 mm and 9×10^{-4}. Floor-preservation and gate-independence therefore hold on *every* reported metric, not only the headline minADE_5 – the refined trajectories, their endpoints, and the miss rate are all reproduced to sub-pixel precision regardless of the random seed.



(a) Parameter budget.

(b) Jacobian-probe Lipschitz analysis.



(c) Multi-seed stability.

Figure 10: **Parameter budget, Lipschitz probe, and multi-seed stability** (server-verified; see the surrounding text for the full reading). **(a)** Parameter budget of the BReF refiner (per-module counts from the released checkpoint): the two attention stacks hold 92% of the 575,747 parameters, the τ-conditioning MLP is the principal module the continuous-depth refiner adds over the matched MLP, and the entire gate costs 2,049 parameters (0.36%). **(b)** Jacobian-probe Lipschitz analysis (16 batches): every SS3 batch lies far below the $\hat{L}=1$ contraction boundary (median 0.018), whereas the matched MLP straddles it (median 0.95); under recursion $\Delta_k=\Delta_1\hat{L}^{k-1}$ the ODE residual self-heals to zero while the MLP sustains (worst case amplifies toward) its single-pass error. **(c)** Multi-seed stability (three seeds): both default and gate-forced minADE_5 stay within a few millimetres of the backbone floor on every seed (1.4 ± 1.0 mm), two orders of magnitude below the anchor-seed gaps of the MLP (348 mm), ODE+jerk (231 mm) and ODE-without-smoothness (170 mm).

Effect size. Read as a standardised separation the margin is decisive: the gate-forced gap of the three non-smoothness families (170–348 mm) exceeds BReF's per-seed standard deviation ($\sigma_{\text{seed}} \approx 1$ mm) by more than two orders of magnitude, i.e. a distance of $> 100\,\sigma_{\text{seed}}$ between BReF and every alternative refiner. A gap this large cannot be manufactured by seed variation, which is precisely why a single anchor checkpoint suffices for the headline comparison and per-seed re-training of the baselines is unnecessary. It is the smoothness-in-τ inductive bias, not a fortunate initialisation, that places BReF on the backbone floor. All three seed checkpoints, the gate-forced and Jacobian-probe scripts, and the per-seed evaluation logs are released so that every figure in this appendix is independently reproducible.

G.8 The design space collapses onto the robustness axis

The robustness evidence of this appendix can be read at a glance in a single design-space view (Fig. 11), which plots *every* refiner we trained on two axes: the default minADE_5 by which it is conventionally judged (horizontal) and the gate-forced Δ_5 that exposes the validity of its raw residual (vertical, log). Three observations follow.

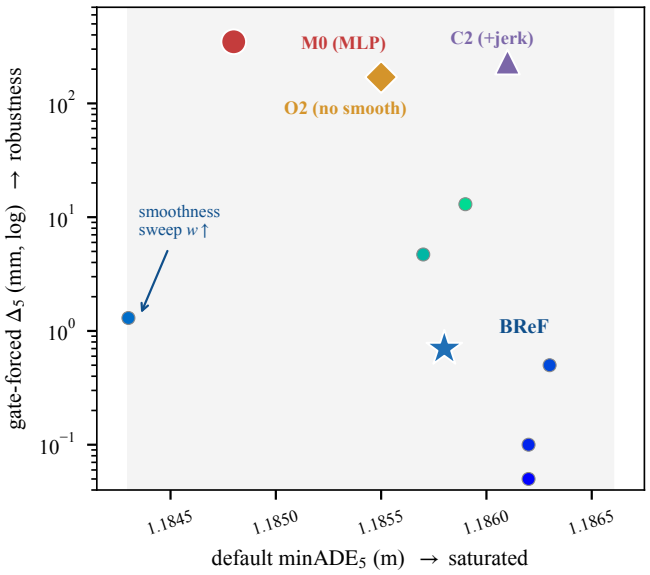


Figure 11: Design-space view (server-verified). Every trained refiner plotted by default minADE_5 (horizontal: a 2.3 mm saturated band) against gate-forced Δ_5 (vertical, log: the validity of the raw residual). The four families and the eight smoothness-weight settings share one default yet span three orders of magnitude in robustness; increasing w_{smooth} (coloured sweep) slides BReF to the robust bottom of the axis. The default leaderboard collapses the design space to a point; the gate-forced axis resolves it.

The horizontal axis is degenerate. Every variant — the matched MLP, the bare ODE, ODE+jerk, all eight smoothness-weight settings, and BReF — lands in a band only 2.3 mm wide (1.1843–1.1866 m). Read on the metric the field reports, the entire design space is effectively a single point: the default leaderboard cannot tell these architectures apart, which is exactly the gate-compression saturation derived in Appendix C.2.

The vertical axis spans three orders of magnitude. The same variants range from 0.1 mm to 348 mm on the gate-forced axis – the dimension along which the architectures genuinely differ, and the one the default metric cannot see. The matched MLP sits at the top (348 mm): its raw correction is large and the outer gate carries it. BReF sits at the bottom (0.7 mm): its raw drift field is already valid, so the gate is near-removable — the 0.4 mm cost of deleting it entirely (variant B2, Table 4) is just the short vertical hop from BReF to the backbone floor.

Smoothness-in-τ is the coordinate. The eight sweep points trace a near-vertical descent: raising w_{smooth} from 0 to 10^{-2} slides a refiner down the robustness axis from 170 mm to 0.7 mm *without* moving it horizontally. The one knob the field has overlooked — because it does not register on the default metric — is precisely the one that converts a gate-dependent corrector into a gate-independent one. The paper's central claim is thus recoverable as the geometry of the design space: default accuracy is saturated, robustness is the operative axis, and the latent-time smoothness penalty is the coordinate along it.

What the appendix establishes. Read end to end, the diagnostics assembled here close the inferential gap between the saturated default metric and the architectural claim of the paper, and they do so along a single coherent thread. Appendix C first proves the floor-preserving property — the bounded, identity-initialised gate makes the refined prediction equal to the frozen backbone at the first optimiser step — and then derives the gate-compression identity that explains *why* four architecturally distinct refiners collapse into a 1.3 mm default band, motivating a diagnostic that bypasses the gate. Appendix D opens exactly that axis. The twelve-variant loss ablation isolates smoothness-in-τ as the unique driver of gate-forced robustness, with no other physical loss reproducing it (Table 4); the smoothness-weight sweep shows the gate-reliance gap collapsing 243×, from 170.3 mm to 0.7 mm, as that single coordinate is turned up (Fig. 4b); the solver- and step-sweeps show the resulting drift field integrates to the same trajectory under any discretisation, the signature of a genuine continuous-time vector field (Fig. 5b); the Jacobian probe certifies that field as a strict

contraction, $\hat{L}_{\mathrm{ODE}}=0.018\ll 1$ on every batch against $\hat{L}_{\mathrm{MLP}}\approx 1.0$ (Fig. 10b); and the three independent re-trainings confirm the entire effect is reproducible to ± 1 mm across seeds (Fig. 10c). Algorithm 2 and Equation (15) pin the implementation to the exact integrator and objective that produce these numbers, and the design-space view of Fig. 11 folds the whole body of evidence into one geometric statement. Every value reported in this appendix is server-verified, and the seed checkpoints, the gate-forced and Jacobian-probe scripts, and the per-seed evaluation logs are released so that each figure and table can be reproduced independently.

Scope of the diagnostics. Two caveats bound these claims. First, the gate-forced probe measures the validity of the *raw* single-pass residual: it is a stress test of the learned field, not a closed-loop or multi-step-rollout metric, so the $497\times$ ratio is evidence about the inductive bias rather than a deployment-time guarantee. Second, the multi-seed study uses three seeds and is reported as a stability check — sufficient to exclude single-checkpoint coincidence given the $>100\,\sigma_{\mathrm{seed}}$ separation, but not a full sampling distribution. Finally, the design-space geometry of Fig. 11 is established for the frozen LAformer-family backbone studied here; whether the same saturated-default, open-robustness structure holds for other backbones, datasets, or longer horizons is the subject of the transfer study left to future work. None of these caveats affects the reproducibility of the reported numbers, each of which is regenerated by the released scripts from the public checkpoints.

Released artifacts. To support exact reproduction we release the frozen backbone checkpoint (EMA epoch 25); the three BReF refiner checkpoints for seeds 777/222/555 together with the eight smoothness-weight and the solver/K ablation checkpoints; the evaluation harness with the gate-forced ($\sigma(\boldsymbol{\gamma})\equiv 1$), inference-$K$, solver-swap, and Jacobian-probe entry points used throughout Appendices D–G; and the per-epoch and per-seed evaluation logs in JSON. Every figure and table in this appendix is regenerated by a single command from these artifacts under the configuration of Table 18, with the random seed, solver, and step count recorded in each log so that the gate-forced Δ_5 values reproduce to the reported precision. The probe code in particular — the 25-step power iteration of Appendix G.6 and the gate-forced override — is deliberately small and dependency-free, so that the central robustness claim can be audited independently of our training pipeline.