ON THE STABILITY AND CONVERGENCE OF FLOW MAP TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Flow maps are a recent method of probabilistic generation that amortize diffusion trajectories into one-step samplers, unlocking efficient generation. While progress has been made into improving their empirical performance, the understanding of the mechanisms underlying their success remains largely unknown. Specifically, flow maps employ stop gradient operators as a crucial part of their training target, which seem necessary to obtain convergent methods. Adaptive weighting also drastically improves the generation capabilities of flow maps. Although many interpretations have been proposed to support the inclusion of these heuristics, none have analyzed the impact of these assumptions on the training dynamics of flow maps themselves. In this work, we address this issue by providing grounded theory explaining why such tricks are required. Analyzing the functional gradient flow induced by the MeanFlow and Lagrangian objective, we show their convergence to the true flow map, settling the validity of these objectives. For finite updates, we highlight a severe roadblock to the usage of these losses in practice; the possible unboundedness of the update. Adaptive normalization not only provides bounded updates, but combined with a specific weighting, additionally provides convergence guarantees towards a neighborhood of the solution. On synthetic data, we show that our functional gradient flow analysis exactly describes the true training dynamics for flow maps. Experiments on standard image datasets demonstrate that a proper reweighting scheme is essential for strong empirical performance, which is only achieved for a specific range of reweighting strengths.

1 INTRODUCTION

Modern deep generative models, such as diffusion (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021) and flow matching models (Lipman et al., 2023; Albergo and Vanden-Eijnden, 2023; Liu et al., 2023), can now generate realistic data across a wide range of modalities. Diffusion and flow matching methods now generate samples that are indistinguishable from human-generated content, including images (Stability AI, 2023; Midjourney, 2023; Black Forest Labs et al., 2025), audio (Borsos et al., 2023), and video (Villegas et al., 2023; Brooks et al., 2024). Diffusion and flow matching have also recently shown great performance on text, either with discrete and continuous approaches (Savinov et al., 2022; Gong et al., 2023; Sahoo et al., 2024; Lee et al., 2026; DiffusionGemma Team et al., 2026).

Once trained, sampling for flow matching models is typically done by numerically solving an ordinary differential equation, *i.e.*, integrating the learned velocity over typically hundreds of steps, hence requiring hundreds of neural network evaluations. This high computational cost at inference has motivated the development of few-step generative models, either by straightening the flow (Liu et al., 2023; Esser et al., 2024), or via distribution matching approaches (Yin et al., 2024; Zhou et al., 2025). Current promising one-step generative modelling approaches rely on distilling the inference trajectories and include techniques such as consistency models (Song et al., 2023), which learn to predict directly the trajectory endpoint of the flow-matching ordinary differential equation from any initial time. In the same spirit, flow maps (Geng et al., 2025; 2026; Boffi et al., 2025; Lu et al., 2026) learn the solution of the ordinary differential equation between any two times.

A variety of approaches for training flow maps have been proposed (Geng et al., 2025; 2026; Boffi et al., 2025; Zhou et al., 2026; Goldstein et al., 2026; Frans et al., 2025; Zhang et al., 2026), due to the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

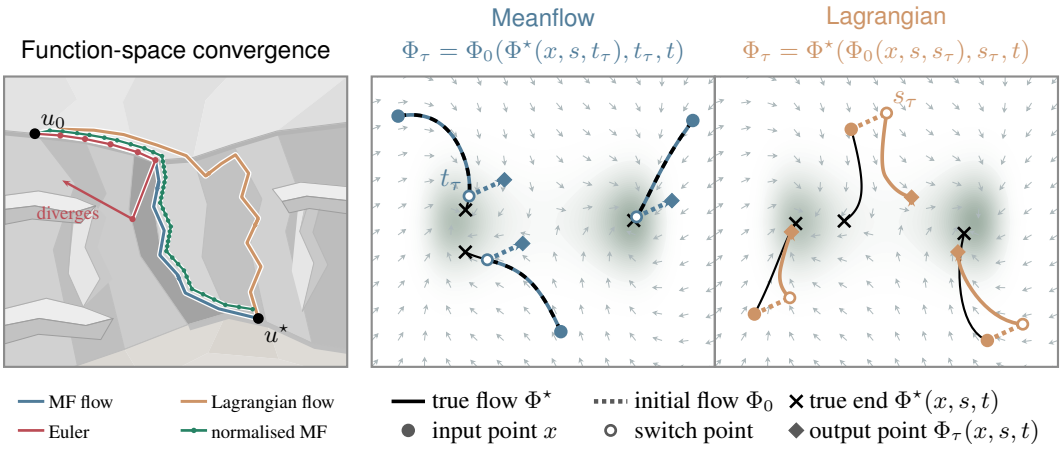


Figure 1: **(Left)** Illustration of the main results of the paper: i) gradient flow on the MeanFlow and Lagrangian objectives provably converge, ii) gradient descent can diverge on both objectives, and iii) normalized gradient descent on MeanFlow provably converges toward a neighborhood of the solution. **(Right)** The ground-truth velocity field $v(\cdot,t)$ is chosen to be constant-in-time. Its associated flow map is $\Phi^\star(\cdot,s,t)$ for fixed s,t. The black curves show the trajectories $(\Phi^\star(x,s,r))_{s\le r\le t}$ for several inputs x (circles). The colored curves show the composite path that the input points (circles) traverse to reach the endpoint $\Phi_\tau(x,s,t)$ (diamonds): for MeanFlow, x follows $\Phi^\star$ until t_τ and then Φ_0; for Lagrangian, it follows Φ_0 until s_τ and then $\Phi^\star$.

abundance of possible training objectives. To stabilize training, several tricks are employed, including stop gradient operators, self-distillation from specialized velocity heads, loss reweighting, gradient normalization, and sampling time pairs from hand-picked distributions (Geng et al., 2025; 2026; Sabour et al., 2025; Lu et al., 2026). *There is little understanding behind why certain losses work in practice*, and whether the tricks used for training are theoretically sound. In particular, the inclusion of stop gradient operators has been found to substantially improve flow map training (Geng et al., 2025); however, it alters gradient directions and has yet to be shown to recover the true flow map.

Contributions. In this paper, we aim to understand the impact of the *stop gradient* operator and of *adaptive normalization* on the learning dynamics of flow maps, and to show how they provide tangible benefits to the convergence and stability of flow map training. We show the following:

1. **Functional convergence of flow maps**. We show that, in a specific functional setting, the gradient flow of the MeanFlow and Lagrangian objectives converge to the desired flow map (Theorem 1). Importantly, the convergence results crucially rely on the careful stop gradient placement in the MeanFlow and Lagrangian losses.
2. **Adaptive normalization**. We show that a fundamental limitation of gradient descent is the unboundedness of the update (Proposition 1), which may amplify arbitrarily small errors. We demonstrate that, crucially, adaptive normalization of the loss, an empirically effective stabilization tool (Geng et al., 2025), counters this blow-up by bounding the update, and provides convergence guarantees under a specific weighting (Theorem 4).
3. **Empirical validation**. In synthetic experiments where the true flow map can be calculated, we show that at intermediate training times, the learned maps obtained through training under the MeanFlow and Lagrangian objectives equal those predicted by our theory. Further experiments on CIFAR-10 and FFHQ 64×64 show that reweighting using adaptive normalization with weights that guarantee stable updates, results in learned flow maps with lower FID.

Organization. Section 2 provides reminders on flow matching and flow map training objectives. Section 3 includes our main results: we analyze the gradient flow and gradient descent dynamics of flow map training from a functional perspective. Section 4 provides a summary of related work, and Section 5 provides experiments on standard image datasets that confirm the theory developed in Section 3.

2 Background and Preliminaries

Flow matching. Let p_0 be the source distribution, typically a Gaussian $\mathcal{N}(0,I)$, and p_{data} the target data distribution. The goal of flow matching (Albergo and Vanden-Eijnden, 2023; Liu et al.,

2023; Lipman et al., 2023) is to find a velocity field $v : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$, such that, if we solve the following ordinary differential equation on $[0,1]$:

$$\begin{cases} \dot{x}(t) = v(x(t), t), \\ x(0) = x_0 \in \mathbb{R}^d, \end{cases} \tag{1}$$

then $x(1) \sim p_{\text{data}}$ for $x_0 \sim p_0$. We denote by p_s the distribution of the $x(s)$ if $x_0 \sim p_0$; namely, $p_1 = p_{\text{data}}$. If we know the velocity field v, new samples can then be generated by sampling x_0 from p_0, solving the ordinary differential equation, and using $x(1)$ as the generated point. Conditional flow matching learns a velocity field by minimizing the following loss:

$$\mathcal{L}^{\text{CFM}}(\theta) = \mathbb{E}_{x_0 \sim p_0, x_1 \sim p_{\text{data}}, t \sim \text{U}(0,1)} \| v_\theta((1-t)x_0 + t x_1, t) - (x_1 - x_0) \|^2. \tag{2}$$

Numerical integration schemes typically rely on a discretization of the time interval. Achieving accurate samples generally requires a fine discretization, often corresponding to hundreds of function evaluations of the neural network. This can lead to prohibitive computational costs, particularly for high-dimensional data such as high-resolution images. Several approaches have therefore been proposed to reduce the computational cost of sampling (Liu et al., 2023; Pooladian et al., 2023; Tong et al., 2024). Below, we briefly review flow maps as an alternative approach to improving sampling efficiency.

Flow maps. Instead of learning the velocity field v, one-step techniques aim to directly learn the *flow*, *i.e.*, the solution of the ordinary differential equation Equation (1). Flow map techniques directly learn the flow $\Phi(x_s, s, t)$, which maps any starting point x_s at time s to its corresponding solution at a later time $t > s$. More precisely, for a given velocity v, the flow map $\Phi^\star(x_s, s, t)$ is defined as the solution at time t of the following ordinary differential equation, initialized at x_s at time s:

$$\Phi^\star(x_s, s, t) \text{ is the solution of the following ODE} \quad \begin{cases} \dot{x}(t) = v(x(t), t) \\ x(s) = x_s \in \mathbb{R}^d \end{cases} \quad \text{at time } t. \tag{3}$$

The velocity v can be either a pre-trained network v_θ or the exact closed-form $v^\star$ of Equation (2) (Bertrand et al., 2025). Once such a flow map is learned, one can sample from p_{data} by sampling $x_0 \sim p_0$ and then computing $\Phi^\star(x_0, 0, 1)$, requiring only a single neural network evaluation.

Learning flow maps. One way to learn flow maps is to find a fixed-point equation that is satisfied by the optimal flow map defined in Equation (3). This can be done by differentiating the definition of the flow map Equation (3) with respect to s: this yields the *Eulerian* identity (Boffi et al. 2025, Prop. 3.6)

$$\partial_s \Phi^\star(x, s, t) + \nabla_x \Phi^\star(x, s, t) v(x, t) = 0 \ . \tag{4}$$

Associated to each flow map Φ is the average velocity u satisfying $\Phi(x, s, t) = x + (t - s)u(x, s, t)$. The parameterizations are equivalent in theory, but in practice the average velocity parameterization ensures $\Phi(x, s, s) = x$ while learning. The popular *MeanFlow* objective (Geng et al., 2025) can be seen as a stop gradient-version of the average velocity u parameterization substituted the Eulerian identity (Equation (4))

$$\begin{aligned} \mathcal{L}^{\text{MF}}(u) &= \frac{1}{2} \mathbb{E}_{x,s,t} \left\| u(x, s, t) - \text{sg}(u^{\text{MF}}_{\text{tgt}}(x, s, t)) \right\|^2 \ , \\ u^{\text{MF}}_{\text{tgt}}(x, s, t) &= v(x, s) + (t - s) \left(\nabla_x u(x, s, t) \cdot v(x, s) + \frac{\partial}{\partial s} u(x, s, t) \right) \quad \text{(MeanFlow)} \ . \end{aligned} \tag{5}$$

Here, the sg denotes the *stop gradient* operator: if the gradient of the loss $\mathcal{L}^{\text{MF}}$ is taken with respect to the average velocity u or parameters that control u, the terms within the sg function have their gradient artificially set to 0. Differentiating the definition of the flow map Equation (3) with respect to t instead yields the *Lagrangian* identity (Boffi et al. 2025, Prop. 3.5)

$$\partial_t \Phi^\star(x, s, t) = v(\Phi^\star(x, s, t), t) \ . \tag{6}$$

The same parametrization $\Phi(x, s, t) = x + (t - s)u(x, s, t)$ with the Lagrangian identity leads to the *Lagrangian* objective. We show in Section B.5 that this recovers the AYF-LMD objective from Sabour et al. (2025), up to reweighting

$$\mathcal{L}^{\mathrm{L}}(u) = \frac{1}{2}\mathbb{E}_{x,s,t}\left\|u(x,s,t) - \mathrm{sg}(u^{\mathrm{L}}_{\mathrm{tgt}}(x,s,t))\right\|^2 \ ,$$
$$u^{\mathrm{L}}_{\mathrm{tgt}}(x,s,t) = v(x+(t-s)u(x,s,t),t) - (t-s)\frac{\partial u}{\partial t}(x,s,t) \ \text{(Lagrangian)} \ . \tag{7}$$

Comments on Equations (5) and (7). Such stop gradient-based losses can be motivated in several ways. Empirically, since all partial derivatives of the network are contained within the stop gradient, this removes the need for double backpropagation. Theoretically, a minimizer of $\mathcal{L}$ is necessarily a flow map, since $\mathcal{L} \geq 0$ with equality if and only if the Eulerian or Lagrangian identity holds. However, although a zero training loss implies being a flow map, *the stop gradient operator might prevent optimization from necessarily decreasing the loss* $\mathcal{L}$: in practice, the MeanFlow loss has been shown not to converge to zero over iterations (Geng et al., 2026). Furthermore, Goldstein et al. (2026) theoretically showed that the model update induced by the MeanFlow loss cannot be the gradient with respect to a scalar loss.

In conjunction with the concurrent work of Shen et al. (2026), thoroughly discussed in Section 4, this work provides the first (functional) convergence result of the flow map training dynamics for the MeanFlow and Lagrangian objectives.

3 Theoretical Analysis of Functional Flow Maps

We now present our main theoretical results and rigorously analyze the training dynamics of the MeanFlow and the Lagrangian methods. The main tools required to perform functional gradient descent are introduced in Section 3.1. Section 3.2 shows our main convergence results: the MeanFlow and Lagrangian *gradient flow* training dynamics converge towards the true flow map. This uncovers the non-obvious result: *regressing against the stop gradient target not only yields practical benefits, but provably recovers the true flow map*. In Section 3.3, we analyze how, under *gradient descent*, MeanFlow and Lagrangian training fail to recover the true flow map due to the divergence of iterates, and how a specific range of adaptive normalization strengths prevents such a failure.

3.1 Setting

Specific inner product choice. We consider losses of the form of Equation (5) and Equation (7). The choice of how to sample x, s and t induces a density ρ on $\mathbb{R}^{d+2}$. For example, choosing (s,t) uniformly from the lower-diagonal triangle $\Delta = \{0 \leq s \leq t \leq 1\}$ and sampling x from the interpolant distribution at time s yields $\rho(x,s,t) = 2p_s(x)$. Under these assumptions, we can write the losses as

$$\mathcal{L}(u) = \int_{x,s,t} \frac{1}{2}\left\|u(x,s,t) - \mathrm{sg}(u_{\mathrm{tgt}}(x,s,t))\right\|^2 \rho(x,s,t)\mathrm{d}x\mathrm{d}s\mathrm{d}t \ , \tag{8}$$

with u_{tgt} either the MeanFlow target (Equation (5)) or the Lagrangian target (Equation (7)). As the input space of $\mathcal{L}$ is infinite-dimensional, computing the gradient of $\mathcal{L}$ requires the choice of an inner product on the functional space. We use the notion of Fréchet derivatives (see Section A.1) with the $L^2(\mathbb{R}^d \times \Delta)$ inner product. Under this choice of inner product, one can compute the gradient of the loss in Equation (8) to be $\nabla\mathcal{L}(u) = \rho(u - u_{\mathrm{tgt}})$.

Exact gradient (no stochasticity). For most machine learning applications, computing the true deterministic gradient is too expensive, and one has to resort to stochastic gradient algorithms. To simplify our analysis, we disregard the potential effects of stochasticity on the optimization dynamics, and assume access to the true gradients $\nabla_u\mathcal{L}^{\mathrm{MF}}$ and $\nabla_u\mathcal{L}^{\mathrm{L}}$.

Gradient flow. For a loss $\mathcal{L}$ and a fixed step-size $\eta > 0$, the gradient descent updates (u_k) write

$$\tfrac{1}{\eta}(u_{k+1} - u_k) = -\nabla_u\mathcal{L}(u_k) \ . \tag{9}$$

Gradient flow is the continuous-time limit of gradient descent as the step-size $\eta \to 0$, where instead of taking discrete steps, we continuously follow the negative gradient $-\nabla\mathcal{L}$. In terms of the *optimization time* τ, the continuous analogue of the number of steps ($\tau = \eta k$), the gradient flow dynamics satisfies

$$\partial_\tau u_\tau = -\nabla_u\mathcal{L}(u_\tau) \ . \tag{10}$$

The gradient flow setting provides tractable analysis by avoiding finite step size issues, while remaining a relevant model for practical optimization.

3.2 CONVERGENCE RESULTS OF THE GRADIENT FLOW DYNAMICS

We first show that the flow-map training dynamics Φ_τ, induced by gradient flow (Equation (10)) with the MeanFlow and Lagrangian objectives, converge to $\Phi^\star$. Equipped with our functional inner product, the functional gradients of the objectives in Equations (5) and (7) write

$$\nabla_u \mathcal{L}^{\mathrm{MF}}(u)(x,s,t) = \rho(x,s,t)\left(u(x,s,t) - v(x,s) - (t-s)\left((\nabla_x u)v(x,s) + \frac{\partial u}{\partial s}\right)\right) \quad , \tag{11}$$

$$\nabla_u \mathcal{L}^{\mathrm{L}}(u)(x,s,t) = \rho(x,s,t)\left(u(x,s,t) - v(x+(t-s)u(x,s,t),t) + (t-s)\frac{\partial u}{\partial t}\right) \quad . \tag{12}$$

We now solve the gradient flow Equation (10) that arises from these values of $\nabla_u \mathcal{L}$. We equip u with another variable τ, where u_τ *denotes the trained flow map at optimization time* τ. Our main result Theorem 1 requires standard regularity assumptions on the velocity v and the sampling density ρ.

Assumption 1 (Regularity of the velocity v and the sampling density ρ)**.** *We assume that the velocity $v(x,s)$ is locally Lipschitz in both x and s. We assume that the sampling density $\rho(x,s,t)$ is locally Lipschitz in x, s and t for all $(x,s,t) \in \mathbb{R}^d \times \Delta$, and $\rho(x,s,t) > 0$ everywhere.*

Then we need to assume that the initial average velocity u_0 for the gradient flow dynamics has some regularity as well.

Assumption 2 (Regularity of the average velocity initialization u_0)**.** *The average velocity initialization u_0 is C^1 on the interior of $\mathbb{R}^d \times \Delta$, and continuous at the boundary.*

Under these assumptions, we obtain one of our main results: flow map pointwise convergence.

Theorem 1. *Assume the velocity and the sampling density satisfy Assumption 1, and suppose the initial mean flow u_0 satisfies Assumption 2. If the mean flow u_τ satisfies the functional gradient flow Equation (10), then its associated flow map $\Phi_\tau(x,s,t) := x + (t-s)u_\tau(x,s,t)$ has a closed-form solution for the MeanFlow and Lagrangian objectives*

$$\Phi^{\mathrm{MF}}_\tau(x,s,t) = \Phi_0(\Phi^\star(x,s,t_\tau),t_\tau,t) \quad , \qquad \text{if } \mathcal{L} = \mathcal{L}^{\mathrm{MF}} \quad \textit{(MeanFlow)} \quad , \tag{13}$$

$$\Phi^{\mathrm{L}}_\tau(x,s,t) = \Phi^\star(\Phi_0(x,s,s_\tau),s_\tau,t) \quad , \qquad \text{if } \mathcal{L} = \mathcal{L}^{\mathrm{L}} \quad \textit{(Lagrangian)} \quad , \tag{14}$$

where $\Phi_0 = x + (t-s)u_0$ and t_τ (resp. s_τ) is an increasing (resp. decreasing) function in τ with $t_0 = s$ (resp. $s_0 = t$), whose speed depends on ρ. Hence, Φ converges pointwise to the true flow map as $\tau \to \infty$ under $\mathcal{L}^{\mathrm{MF}}$. If $\Phi^\star$ is also continuous in (x,s), the same holds for $\mathcal{L}^{\mathrm{L}}$.

The proof can be found in Section B.1. For the MeanFlow objective, the key idea is that the PDE induced by the MeanFlow target gives a *linear* PDE, hence can be solved by the method of characteristics. Under suitable conditions, the characteristics trace back to the initial conditions, and the speed at which the characteristics are traversed is given by our weighting ρ. For the Lagrangian objective, although the PDE is now non-linear as u appears in the velocity in Equation (12), the same method of characteristics can be used to show convergence.

The closed-form in Theorem 1 also provides insights into how training under the MeanFlow loss structurally modifies the learned map: as training progresses, the MeanFlow map learns to transport samples using the true flow to increasingly later times t_τ, and the initialization Φ_0 estimates the remaining jump to the time t. On the other hand, under the Lagrangian objective, training teaches the flow map to transport samples from increasingly earlier initial times s_τ, with the initial transport up to time s_τ given by our initialization Φ_0. Both objectives learn the flow map, but by replacing the initialization with the true flow map from opposing directions.

Rates of convergence. Using the convergence results of Theorem 1, we can derive convergence rates for the learned maps in the L_∞ (Theorem 2) and L_2-norm (Theorem 3).

Theorem 2. *Assume Assumptions 1 and 2 hold. In addition, assume $\rho(x,s,t) \geq \rho_{\min} > 0$ and $\|u_0\|_\infty, \|v\|_\infty \leq M$. Then the MeanFlow closed-form satisfies*

$$\|\Phi^{\mathrm{MF}}_\tau(x,s,t) - \Phi^\star(x,s,t)\|_\infty \leq 2Me^{-\rho_{\min}\tau} \quad . \tag{15}$$

In addition, if v is uniformly L-Lipschitz for all t, then the Lagrangian closed-form satisfies

$$\|\Phi_\tau^{\mathrm{L}}(x,s,t) - \Phi^\star(x,s,t)\|_\infty \le 2Me^L e^{-\rho_{\min}\tau} \ . \tag{16}$$

Proof of Theorem 2 and examples showing these bounds are tight (up to constants) can be found in Section B.2. The lower bound on ρ is a mild assumption; it holds over compact domains when ρ is positive. Theorem 3 formulates convergence rates with respect to the $L^2(p_s/\rho)$ norm (with the abuse of notation that p_s is of function of x, s, t): for functions $f : \mathbb{R}^d \times \Delta \to \mathbb{R}^d$, its norm $L^2(g)$ with respect to a function $g : \mathbb{R}^d \times \Delta \to \mathbb{R}_{\ge 0}$ is given by

$$\|f\|^2_{L^2(g)} = \int_{\mathbb{R}^d \times \Delta} g(x,s,t)\|f(x,s,t)\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t. \tag{17}$$

The complementary convergence results (Theorem 3) in the $\|\cdot\|_{L^2}$-norm requires additional integrability conditions on the average velocity solution u_τ of the gradient flow dynamics Equation (10).

Assumption 3 (Integrability of the gradient flow average velocities u_τ)**.** *Suppose $u_\tau - u^\star \in C^1$ on the interior of $\mathbb{R}^d \times \Delta$ and $p_s\|u_\tau - u^\star\|^2 + p_s(t-s)|(v\cdot\nabla_x + \partial_s)\|u_\tau - u^\star\|^2|$ is dominated by an integrable function independent of τ for $\tau \in [0,T]$, for all $T < \infty$.*

Under the norm defined in Equation (17) and Assumption 3, we show convergence of the average velocity solution of the gradient flow u_τ to the ground-truth average velocity $u^\star(x,s,t) \mapsto (\Phi^\star(x,s,t) - x)/(t-s)$. Proof of Theorem 3 can be found in Section B.2.

Theorem 3. *Assume Assumptions 1 to 3 hold, and $\rho(x,s,t) \ge \rho_{\min} > 0$. If $\|u_0^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} < \infty$ then the MeanFlow trained average velocity u_τ^{MF} satisfying Equation (10) fulfills:*

$$\|u_\tau^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} \le e^{-\rho_{\min}\tau}\|u_0^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} \ , \tag{18}$$

hence, the average learned velocity u_τ^{MF} converges to $u^\star$ with respect to the $L^2(p_s/\rho)$-norm

Again the assumption $\|u_0^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} < \infty$ is mild as it is true on compact domains when $\rho > 0$. As stated in Section 2, the MeanFlow objective crucially relies on the stop gradient operator, which alters the gradient direction. Specifically, the stop gradient may prevent the dynamics Φ_τ from converging toward the true flow map $\Phi^\star$. However, somewhat remarkably, Theorem 3 shows that the update direction induced by the MeanFlow loss implicitly leads us closer to the true flow map anyway: $\Phi_\tau^{\mathrm{MF}} \to \Phi^\star$.

3.3 Instability and normalization of gradient descent

In practice, one cannot truly follow the gradient flow (Equation (10)): instead one can resort to gradient descent-like algorithms: a learning rate $\eta \ll 1$ is chosen, and a gradient step is taken with step size η

$$u_{k+1} = u_k - \eta\nabla_u\mathcal{L}(u_k) \ , \tag{19}$$

where $\mathcal{L} = \mathcal{L}^{\mathrm{MF}}$ or $\mathcal{L}^{\mathrm{L}}$. *In contrast to the gradient flow results, we can show that gradient descent is not guaranteed to converge, even for the trivial velocity $v = 0$.* The idea is that the differential operators present in both $\nabla_u\mathcal{L}^{\mathrm{MF}}$ and $\nabla_u\mathcal{L}^{\mathrm{L}}$ are unbounded and can amplify errors. One can exhibit initial velocities u_0 such that gradient descent diverges when applied with the MeanFlow and Lagrangian objectives (see Section B.4 for the proof of Proposition 1 and details of the counterexample).

Proposition 1. *For every $\eta > 0$, $\varepsilon > 0$, and $\rho \in C^\infty$, there exists u_0 whose initial error $u_0 - u^\star \in L^2(\rho) \cap C_c^\infty$ with $\|u_0 - u^\star\|_\infty \le \varepsilon$, but whose iterates under Equation (19) diverge for either $\mathcal{L} = \mathcal{L}^{\mathrm{MF}}$, or $\mathcal{L}^{\mathrm{L}}$ with $v = 0$.*

However, if we adapt our step size according to the magnitude of the update, we can avoid a blow-up. Since the residual $u_k - u_{\mathrm{tgt},k}$ can become arbitrarily large, we consider an update of the form

$$u_{k+1} = u_k - \eta\rho\frac{r_k}{(\|r_k\|^2_{L^2(\rho)} + c)^p}, \quad \text{with} \quad r_k := u_k - u_{\mathrm{tgt},k} \ , \tag{20}$$

for some small fixed $c > 0$, and a choice of normalization weighting p. We now show that choosing $p \ge 0.5$ enables control over the update magnitude.

Theorem 4. *Assume that the velocity v, the density ρ and u_0 fulfill Assumptions 1 and 2. In addition, assume that the iterates of gradient descent satisfy the discrete equivalent of Assumption 3, and that p_s is bounded above: $p_s(x) \leq p_{\max} < \infty$, ρ is bounded below: $\rho(x, s, t) \geq \rho_{\min} > 0$ and $\|u_0 - u^\star\|^2_{L^2(p_s/\rho)} < \infty$. Consider the update scheme given in Equation (20).*

- *For $p \geq 0.5$, the updates have uniformly bounded norms: $\|u_{k+1} - u_k\|^2_{L^2(p_s/\rho)} \leq M < \infty$.*
- *$p = 1$ yields convergence for the MeanFlow objective: if $\|u_k - u_{\text{tgt},k}\|^2_{L^2(\rho)} \leq M(k+1)$ for some $M > 0$, i.e. the norm of the residual grows at most linearly, then*

$$\limsup_{k\to\infty} \|u_k^{\text{MF}} - u^\star\|^2_{L^2(p_s/\rho)} \leq \eta p_{\max}(\rho_{\min}^{-1} + \eta c^{-1}) \ . \tag{21}$$

A proof is found in Section B.4. Theorem 4 shows that, for $p = 1$ the iterates of the normalized gradient descent (defined in Equation (20)) on the MeanFlow objective remain in an $\mathcal{O}(\eta)$ neighborhood of the true flow map $\Phi^\star$. In MeanFlow (Geng et al., 2025), normalization is done through taking adaptive loss weights $w = 1/(\|u_k - u^{\text{MF}}_{\text{tgt},k}\|_2^2 + c)^p$. Our result is equivalent to showing that, when the regression error is integrated over space, taking adaptive weights with this integrated error and $p = 1.0$ has iterates that remain in an $\mathcal{O}(\eta)$ ball around the true solution. $p = 1.0$ is the value that is found to work the best for from-scratch training on ImageNet 256×256 (Geng et al., 2025).

4 Related work

Trajectory distillation. One of the most relevant related work include *consistency models* (Song et al., 2023; Song and Dhariwal, 2024), which builds a function $f(x_s, s)$ mapping directly to the trajectory endpoint x_1. Continuous-time consistency models are trained with a similar loss to the MeanFlow loss (Lu and Song, 2025; Sabour et al., 2025), apart from with a fixed $t = 1$, and hence are amenable to analysis in exactly the same way. In practice consistency models also require similar tricks such as tangent normalization for effective training (Lu and Song, 2025).

Flow maps. Many works have proposed learning schemes for flow maps. Boffi et al. (2025) derive fundamental equations for flow maps and introduce the LSD and ESD objectives. Geng et al. (2025) introduce the MeanFlow objective, which is further improved in Geng et al. (2026) by replacing the stochastic estimate for v in the calculation of $\mathcal{D}_v u$ with a self-distilled diagonal term. Lu et al. (2026) reparametrize the objective as $u = (\hat{x}(x, s, t) - x)/(1 - t)$. $\hat{x}$ can then be shown to lie on the convex hull of p_1 (Lee et al., 2026). Align Your Flow (Sabour et al., 2025) introduces flow map distillation methods derived from consistency estimates. These are equivalent to the methods presented in this paper, up to reweighting. Terminal velocity matching (Zhou et al., 2026) uses a modified LSD loss, where the stop gradient teacher velocity u is replaced with an EMA version. One of the only theoretical works on flow maps is Goldstein et al. (2026), who argue for an alternative loss to MeanFlow, due to the presence of spurious minimizers. While this logic is valid for the ESD objective, for the stop gradient MeanFlow objective, optimization does not stall in a local minimum, as is shown by our convergence results.

Concurrent work. In very recent work, Shen et al. (2026) derived a regression principle for stop gradient dynamics, and used it to show that functional gradient flow for the MeanFlow and Lagrangian loss can be written in closed-form, as in Theorem 1. Shen et al. (2026) derive the same closed form that we have presented in Theorem 1, through slightly different analysis, and achieve similar L_∞ convergence guarantees. Shen et al. (2026) encompasses a larger class of problem such as reinforcement learning algorithm, but their convergence results are only valid for $\rho = 1$. Our work focuses on flow maps convergence: we assume the existence of a density ρ which alters the gradient dynamics, and show convergence in this broader setting. We also introduce novel L^2 convergence rates, and our results on gradient descent are independent of prior work.

5 Experiments

Experiments on synthetic data (Figure 2) validate the faithfulness of the gradient flow closed-form of the flow maps derived in Theorem 1. Experiments on standard image datasets (Figure 3 and table 1) confirm the importance of proper adaptive normalization (Theorem 4).

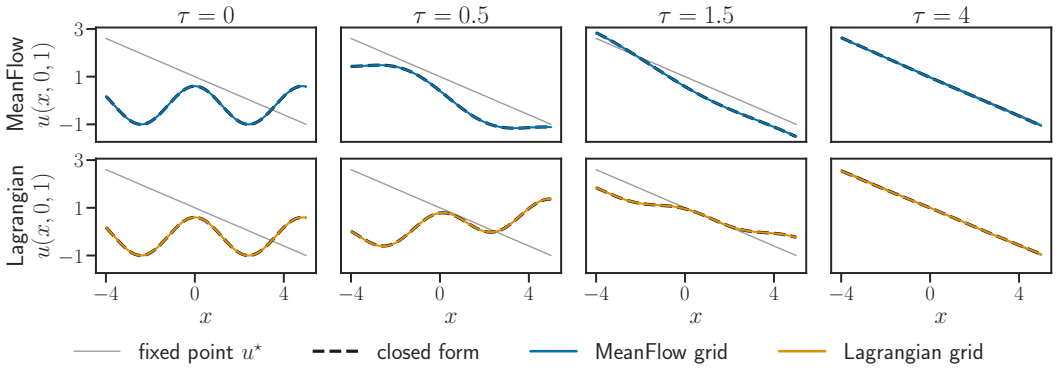


Figure 2: *Synthetic Gaussian-to-Gaussian dataset.* Snapshots of grid function learned average velocities (u_k, solid blue/orange line) versus their theoretically predicted counterparts u_τ at four optimization times τ. The two show strong agreement across iteration times τ and for both the MeanFlow (top) and Lagrangian average velocity target (bottom), and both relax towards the ground truth $u^\star$.

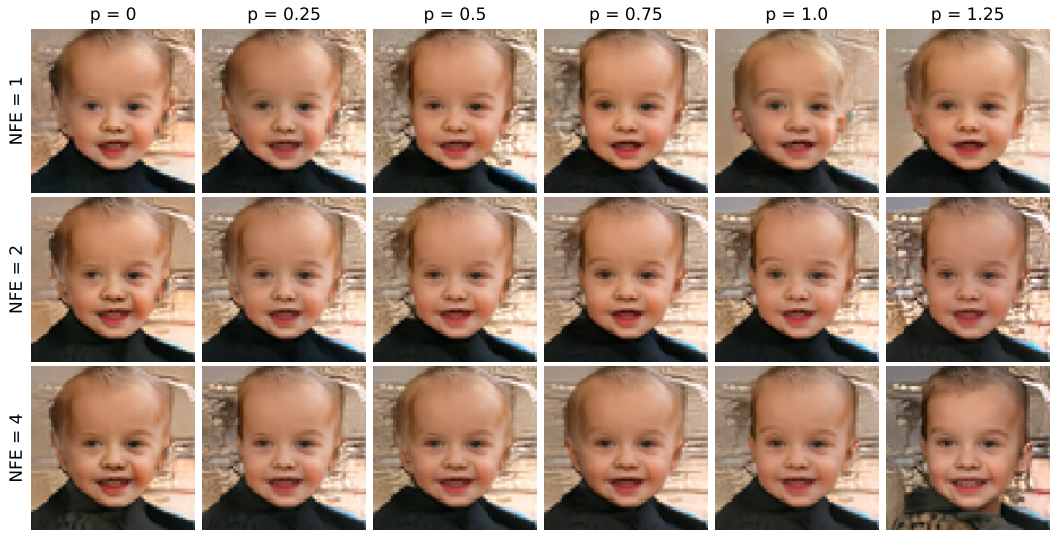


Figure 3: FFHQ dataset. The images are generated from a fixed seed, over different numbers of function evaluations (NFE), with flow maps learned with different normalization exponents p. As predicted by Theorem 4, $p = 0$ and $p = 0.25$ yield poor results.

Synthetic experiments. First, we show that learned flow map training dynamics on synthetic data accurately reproduces the closed-form for the flow maps (Φ_τ^{MF} and Φ_τ^{L}) training dynamics derived in Theorem 1. Theorem 1 assumes functional (*i.e.*, infinite dimensional) optimization: to recreate the assumed functional setting as accurately as possible, in Figure 2 we parametrize the average velocity u pointwise on a discrete grid in $\mathbb{R} \times \Delta$, removing spatial coupling between parameters. Average velocities are trained with grid function backbones to transport $\mathcal{N}(0, 1)$ to $\mathcal{N}(1, 0.36)$ through a linear interpolant path, regressing against the ground truth velocity v. In this setting, Figure 2 displays the learned average velocities (solid blue and orange lines) and the closed-form flow average velocities, u_τ^{MF} and u_τ^{L} (black dashed lines), for multiple optimization times τ. The solid black line represents the ground-truth average velocity u we try to amortize. For completeness, exhaustive empirical details can be found in Section C.1.

Comments on Figure 2. Figure 2 shows that the learned average velocities match their theorized snapshots u_τ^{MF} and u_τ^{L} from Equations (13) and (14): the black dashed line exactly matches the solid blue/orange lines at each optimization time τ. Further experiments analyzing the effect of loss weighting and step size on stability are in Section C.1.

Table 1: FID ↓ for MeanFlow with different normalization exponents p on CIFAR-10 and on FFHQ 64×64. Results are reported as mean ± standard deviation over three runs.

p	Bounded updates	CIFAR-10			FFHQ 64×64		
		NFE 1	NFE 2	NFE 4	NFE 1	NFE 2	NFE 4
0.00	✗	10.62 ± 0.21	10.00 ± 0.69	9.60 ± 0.61	15.38 ± 0.84	12.67 ± 0.60	12.49 ± 0.43
0.25	✗	6.81 ± 0.23	6.70 ± 0.30	6.49 ± 0.18	10.92 ± 0.43	9.46 ± 0.97	9.72 ± 0.68
0.50	✓	5.51 ± 0.06	5.55 ± 0.36	4.93 ± 0.37	8.93 ± 0.21	7.89 ± 0.68	8.15 ± 0.53
0.75	✓	**5.46** ± 0.06	**5.05** ± 0.16	4.78 ± 0.04	**8.67** ± 0.30	6.86 ± 0.91	**7.69** ± 0.74
1.00	✓	5.61 ± 0.20	5.17 ± 0.05	4.69 ± 0.04	10.11 ± 0.21	**5.64** ± 1.11	7.85 ± 0.72
1.25	✓	5.93 ± 0.19	5.25 ± 0.08	**4.46** ± 0.04	12.70 ± 0.04	9.48 ± 4.69	9.89 ± 4.17
1.50	✓	6.32 ± 0.26	5.37 ± 0.54	5.48 ± 1.53	20.25 ± 1.56	57.25 ± 35.14	13.92 ± 0.46

Impact of adaptive gradient normalization. Our developed theory suggests that flow map training is susceptible to instability due to large magnitude updates, and that adaptive normalization offers a solution. We show that the empirical performance of flow maps distilled from pretrained velocity teachers improves when adaptive normalization is used. Here we perform experiments on standard image datasets CIFAR-10 (Krizhevsky and Hinton, 2009) and FFHQ 64×64 (Karras et al., 2019). For CIFAR-10, we used the pretrained velocity of Tong et al. (2024). For FFHQ 64×64, we used the variance exploding denoiser from Karras et al. (2022) converted into a flow matching model. The backbone for the CIFAR-10 and FFHQ 64×64 flow maps is an EDM2 U-Net (Karras et al., 2024) with modified time-embeddings for added stability when computing temporal derivatives (Lu and Song, 2025). We use the MeanFlow loss as in Equation (5), weighted by a factor $w = \mathrm{sg}((\|u_k - u_{\mathrm{tgt},k}\|_2^2 + c)^{-p})$ for various choices of p. Fréchet Inception Distances (Heusel et al., 2017) were computed using torchmetrics (Detlefsen et al., 2022). In Table 1, we display FIDs of the models with exponential moving average obtained after 200K steps of training, for a variety of values of p. Other experimental details, along with results for the Lagrangian loss on CIFAR-10, are in Section D.

Comment on Table 1 and fig. 3. Table 1 compares values of the exponent in adaptive normalization normalization exponent $p = 0$ to $p = 1.5$ and multiple number of function evaluations (NFE 1, 2, 4), and for multiple high dimensional datasets. We observe that $p = 0.5$ to $p = 1$ gives good performances across datasets. For $p < 0.5$, the models have noticeably worse FIDs, which corresponds to the region in which the update is possibly unbounded. For a fixed seed, Figure 3 displays a generated image for multiple values of normalization exponent p and NFE (1, 2, 4). Qualitatively $p \leq 0.25$ and $p = 1.25$ yield the poorest generated image.

6 Conclusion

In this paper, we have analyzed the impact of stop gradient operators, and adaptive normalization on flow map training. We have shown how the placement of stop gradients in the MeanFlow and Lagrangian loss lead to a gradient update that, when optimized with *gradient flow*, recover the true flow map, both pointwise and in L^∞ and L^2 norms. In addition to preventing double backpropagation, stop gradient operators provably move learned flow maps closer to the ground truth. However, gradient descent iterations, when applied on the MeanFlow and Lagrangian losses can yield divergence. We demonstrate how adaptive normalization gives us both a theoretical and practical solution, enabling higher fidelity flow map learning on CIFAR-10 and FFHQ 64×64 datasets.

Limitations and future directions. The scope of the results derived in this paper is limited to the losses discussed in Equations (5) and (7). In particular, we do not show convergence for the original ESD and LSD objectives or the Terminal Velocity Matching objective (Boffi et al., 2025; Zhou et al., 2026). Showing convergence for the gradient flow induced by any of these losses is future work. In addition, our convergence results are derived in a specific functional setting limiting the applicability of the derived results to neural networks, whose parametrization impacts the functional gradient. The main theory of this work is for the idealized gradient flow setting, where stochasticity-based challenges are ignored, as well as other tricks such as adaptive weighting and training with a fixed percentage of samples taken at the diagonal $s = t$. Quantifying the effects of these design choices is left for subsequent studies.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools for brainstorming, implementing methods, rephrasing sentences, and creating scientific visuals (e.g., helping with TikZ code for plots, Figure 1, left). AI provided us with critical ingredients for proving mathematical claims and assisted with the checking of proofs.

We have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, assisting in the writing of proofs, proposing or refine hypotheses, designing or provide feedback on research methodology or experiments, assisting with translation, cleaning and reformat dataset, supporting qualitative and thematic data analysis, interpreting results.

We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

M. S. Albergo and E. Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023. 1, 2

Q. Bertrand, A. Gagneux, M. Massias, and R. Emonet. On the closed-form of flow matching: Generalization does not arise from target stochasticity. *NeurIPS*, 2025. 3

Black Forest Labs, S. Batifol, A. Blattmann, F. Boesel, S. Consul, C. Diagne, T. Dockhorn, J. English, Z. English, P. Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. 1

N. M. Boffi, M. S. Albergo, and E. Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *TMLR*, 2025. 1, 3, 7, 9

Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi, et al. Audiolm: a language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023. 1

T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 2024. 1

N. S. Detlefsen, J. Borovec, J. Schock, A. Harsh, T. Koker, L. Di Liello, D. Stancl, C. Quan, M. Grechkin, and W. Falcon. Torchmetrics–measuring reproducibility in pytorch. *Journal of Open Source Software*, 2022. 9

DiffusionGemma Team, A. A. Taïga, J. Assiene, D. Calandriello, R. Chaabouni, J. Gante, T. von Glehn, N. Keating, C. Knutsen, M. Kukla, et al. Diffusiongemma technical report. *arXiv preprint arXiv:2608.00146*, 2026. 1

P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 1

K. Frans, D. Hafner, S. Levine, and P. Abbeel. One step diffusion via shortcut models. In *ICLR*, 2025. 1

Z. Geng, M. Deng, X. Bai, J. Z. Kolter, and K. He. Mean flows for one-step generative modeling. In *NeurIPS*, 2025. 1, 2, 3, 7

Z. Geng, Y. Lu, Z. Wu, E. Shechtman, J. Z. Kolter, and K. He. Improved mean flows: On the challenges of fastforward generative models. In *CVPR*, 2026. 1, 2, 4, 7

M. Goldstein, A. Uppal, R. Singhal, A. M. Puli, and R. Ranganath. Flow map learning via non-gradient vector flow. In *ICLR*, 2026. 1, 4, 7

S. Gong, M. Li, J. Feng, Z. Wu, and L. Kong. Diffuseq: Sequence to sequence text generation with diffusion models. In *ICLR*, 2023. 1

M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017. 9

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1

T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, June 2019. 9

T. Karras, M. Aittala, T. Aila, and S. Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 9, 24

T. Karras, M. Aittala, J. Lehtinen, J. Hellsten, T. Aila, and S. Laine. Analyzing and Improving the Training Dynamics of Diffusion Models . In *CVPR*, 2024. 9

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. 2009. 9

C. Lee, J. Yoo, M. Agarwal, S. Shah, J. Huang, A. Raghunathan, S. Hong, N. M. Boffi, and J. Kim. Flow map language models: One-step language modeling via continuous denoising. *arXiv preprint arXiv:2602.16813*, 2026. 1, 7

Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. In *ICLR*, 2023. 1, 3

X. Liu, C. Gong, and Q. Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 1, 2, 3

C. Lu and Y. Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *ICLR*, 2025. 7, 9

Y. Lu, S. Lu, Q. Sun, H. Zhao, Z. Jiang, X. Wang, T. Li, Z. Geng, and K. He. One-step latent-free image generation with pixel mean flows. In *ICML*, 2026. 1, 2, 7

Midjourney. https://www.midjourney.com/home/, 2023. Accessed: 2023-09-09. 1

A.-A. Pooladian, H. Ben-Hamu, C. Domingo-Enrich, B. Amos, Y. Lipman, and R. T. Q. Chen. Multisample flow matching: Straightening flows with minibatch couplings. In *ICML*, 2023. 3

A. Sabour, S. Fidler, and K. Kreis. Align your flow: Scaling continuous-time flow map distillation. In *NeurIPS*, 2025. 2, 3, 7, 20

S. S. Sahoo, M. Arriola, Y. Schiff, A. Gokaslan, E. Marroquin, J. T. Chiu, A. Rush, and V. Kuleshov. Simple and effective masked diffusion language models. In *NeurIPS*, 2024. 1

N. Savinov, J. Chung, M. Binkowski, E. Elsen, and A. v. d. Oord. Step-unrolled denoising autoencoders for text generation. In *ICLR*, 2022. 1

M. W. Shen, M. Goldstein, Z. Wang, A. Puli, and R. Ranganath. How i learned to stop worrying and love stopgrads: Stationarity, convergence, and a case study on flow map learning, 2026. 4, 7

J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 1

Y. Song and P. Dhariwal. Improved techniques for training consistency models. In *ICLR*, 2024. 7

Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 1

Y. Song, P. Dhariwal, M. Chen, and I. Sutskever. Consistency models. In *ICML*, 2023. 1, 7

Stability AI. https://stability.ai/stablediffusion, 2023. Accessed: 2023-09-09. 1

A. Tong, K. Fatras, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, G. Wolf, and Y. Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *TMLR*, 2024. Expert Certification. 3, 9, 24

R. Villegas, M. Babaeizadeh, P.-J. Kindermans, H. Moraldo, H. Zhang, M. T. Saffar, S. Castro, J. Kunze, and D. Erhan. Phenaki: Variable length video generation from open domain textual descriptions. In *ICLR*, 2023. 1

T. Yin, M. Gharbi, R. Zhang, E. Shechtman, F. Durand, W. T. Freeman, and T. Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024. 1

H. Zhang, A. Siarohin, W. Menapace, M. Vasilkovsky, S. Tulyakov, Q. Qu, and I. Skorokhodov. Alphaflow: Understanding and improving meanflow models. In *ICLR*, 2026. 1

L. Zhou, S. Ermon, and J. Song. Inductive moment matching. In *ICML*, 2025. 1

L. Zhou, M. Parger, A. Haque, and J. Song. Terminal velocity matching. In *ICLR*, 2026. 1, 7, 9

A Additional Reminders

A.1 Fréchet Derivatives

Fréchet derivatives. Our goal is to minimise the objective in Equation (8) over the infinite-dimensional function space $u \in \mathcal{U} \subset C^1$. For such a loss, we can compute its Fréchet derivative with respect to u. Since u_{tgt} is taken under stop gradient, it is considered constant as a function of u. Hence, we can derive

$$D\mathcal{L}(u)[h] = \int_{x,s,t} (u(x,s,t) - u_{\mathrm{tgt}}(x,s,t)) \cdot h(x,s,t)\rho(x,s,t)\mathrm{d}x\mathrm{d}s\mathrm{d}t \ . \tag{22}$$

This corresponds to the infinitesimal change in $\mathcal{L}$ when the input changes infinitesimally in the direction of h. The gradient of $\mathcal{L}$ at u then points in the direction that increases $\mathcal{L}$ the fastest under an infinitesimal change to the input. Mathematically, this is written as $D\mathcal{L}(u)[h] = \langle \nabla \mathcal{L}(u), h \rangle$. Importantly, the gradient *depends on the choice of inner product* on the decision space. To formally define the gradient, we fix a choice of inner product, namely the $L^2(\mathbb{R}^d \times \Delta)$ inner product.

This choice of metric is natural and can be motivated from homogeneity. More precisely, a change in u is felt equally irrespective of its position in space and time. Furthermore, it also does not result in spatial mixing, which is a central ingredient for the success of our analysis. It should be noted that neural network parametrizations do not naturally have this geometry.

Under this inner product, we see that $\nabla \mathcal{L}(u) = \rho(u - u^{\mathrm{tgt}})$.

B Main Proofs

B.1 Proofs of PDE convergence

Theorem 5. *Under the PDE in Equation (10), with $\mathcal{L} = \mathcal{L}^{\mathrm{MF}}$, under Assumptions 1 and 2, u_τ can be written down in closed form:*

$$u_\tau(x,s,t) = u_0(\Phi^\star(x,s,t_\tau), t_\tau, t)F^{-1}(\tau) + \frac{\Phi^\star(x,s,t_\tau) - x}{t-s}, \tag{23}$$

where $t_\tau = t - (t-s)F^{-1}(\tau)$, and where $F^{-1}(\tau)$ is a decreasing function depending on ρ that quantifies how close we are to t. This implies $u \to u^\star$ pointwise.

Proof. The key idea is that this is a linear PDE, hence it may be solved exactly. We fix t and introduce variables $X \in \mathbb{R}^d$ and $S \in [0,1]$ satisfying

$$\frac{\mathrm{d}X}{\mathrm{d}\tau}(\tau) = -\rho(X,S,t)(t-S)v(X,S),$$
$$\frac{\mathrm{d}S}{\mathrm{d}\tau}(\tau) = -\rho(X,S,t)(t-S).$$

Such an S and X exist under the conditions given in Section B.3. Evaluating the total derivative, we find that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\tau}u_\tau(X,S,t) &= \frac{\partial}{\partial\tau}u_\tau(X,S,t) + \nabla_x u_\tau \cdot \frac{\mathrm{d}X}{\mathrm{d}\tau} + \frac{\partial u_\tau}{\partial s}\frac{\mathrm{d}S}{\mathrm{d}\tau} \\ &= -\rho(X,S,t)\left(u_\tau(X,S,t) - v(X,S)\right).\end{aligned}$$

Introducing an integrating factor F satisfying $\mathrm{d}F/\mathrm{d}\tau = \rho F$ and $F(0) = 1$, we may write this as

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\left(Fu\right) = F(\tau)\rho(X,S,t)v(X,S).$$

Prescribing boundary conditions $X(\mathcal{T}) = x$ and $S(\mathcal{T}) = s$, we find that $S(\tau) = t - (t-s)F(\tau)/F(\mathcal{T})$, and $S(0) = t - (t-s)F^{-1}(\mathcal{T}) := t_\mathcal{T}$. Moreover,

$$\frac{\mathrm{d}X}{\mathrm{d}S} = \frac{\mathrm{d}X}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}S} = v(X,S),$$

hence X follows the velocity field along the characteristic curves. This yields $X(\tau) = \Phi^\star(x,s,S(\tau))$. Solving the ODE now shows that, for $s < t$,

$$\begin{aligned}u_\mathcal{T}(x,s,t) &= u_0(\Phi^\star(x,s,t_\mathcal{T}),t_\mathcal{T},t)F^{-1}(\mathcal{T}) + \int_0^\mathcal{T} \frac{F(\tau)\rho(X(\tau),S(\tau),t)}{F(\mathcal{T})}v(X(\tau),S(\tau))\mathrm{d}\tau \\ &= u_0(\Phi^\star(x,s,t_\mathcal{T}),t_\mathcal{T},t)F^{-1}(\mathcal{T}) + \frac{1}{t-s}\int_s^{t_\mathcal{T}} v(\Phi^\star(x,s,S),S)\mathrm{d}S,\end{aligned}$$

which gives closed form solution

$$u_\tau(x,s,t) = u_0(\Phi^\star(x,s,t_\tau),t_\tau,t)F^{-1}(\tau) + \frac{\Phi^\star(x,s,t_\tau) - x}{t-s}. \tag{24}$$

In terms of Φ, we obtain that

$$\begin{aligned}\Phi_\tau(x,s,t) &= x + (t-s)u_\tau(x,s,t) &(25)\\ &= \Phi^\star(x,s,t_\tau) + u_0(\Phi^\star(x,s,t_\tau),t_\tau,t)\underbrace{(t-s)F^{-1}(\tau)}_{t-t_\tau} &(26)\\ &= \Phi_0(\Phi^\star(x,s,t_\tau),t_\tau,t). &(27)\end{aligned}$$

Under mild conditions on ρ, for example if ρ is bounded below on each trajectory $(x_r)_s^t$, which is true since $\rho > 0$ and each trajectory is a compact set, then, $F^{-1}(\tau) \to 0$ as $\tau \to \infty$, since $\dot{F} \geq \rho_{\min}F \implies F(\tau) \geq e^{\rho_{\min}\tau}$ and so if u_0 is locally bounded at the diagonal, the first term disappears and

$$\lim_{\tau\to\infty} u_\tau(x,s,t) = \frac{\Phi^\star(x,s,t) - x}{t-s} = u^\star(x,s,t). \tag{28}$$

which converges to the instantaneous velocity. We have thus shown that ideal training under the meanflow loss converges to the average velocity. Moreover, we have shown how the speed of convergence is dependent on a choice of weighting ρ. □

Φ_τ can be shown to satisfy a similar PDE,

$$\frac{\partial}{\partial\tau}\Phi_\tau(x,s,t) = (t-s)\rho(x,s,t)\left(\nabla_x\Phi_\tau \cdot v(x,s) + \frac{\partial\Phi_\tau}{\partial s}\right), \tag{29}$$

which can be solved in an analogous manner.

Theorem 6. *Under the PDE in Equation (10), with $\mathcal{L} = \mathcal{L}^{\mathrm{L}}$, under Assumptions 1 and 2, Φ_τ can be written down in closed form:*

$$\Phi_\tau(x,s,t) = \Phi^\star(\Phi_0(x,s,s_\tau),s_\tau,t), \tag{30}$$

where $s_\tau = s + (t-s)F^{-1}(\tau)$, and where $F^{-1}(\tau)$ is a decreasing function depending on ρ that quantifies how close we are to s. In particular if $\Phi^\star$ is continuous in (x,s), then Φ converges pointwise to the true flow map as $\tau \to \infty$.

Proof. This is now a non-linear PDE, since the target Φ_τ appears inside the velocity v. However, it can still be solved through characteristic methods. We first determine the PDE for Φ through the chain rule:

$$\begin{aligned}\frac{\partial}{\partial\tau}\Phi_\tau(x,s,t) &= (t-s)\frac{\partial}{\partial\tau}u_\tau(x,s,t)\\ &= -(t-s)\rho(x,s,t)\left(\frac{\partial\Phi_\tau}{\partial t}(x,s,t) - v(\Phi_\tau(x,s,t),t)\right).\end{aligned}$$

Defining $\dot{T} = (T-s)\rho(x,s,T)$, we have

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\Phi_\tau(x,s,T) = (T-s)\rho(x,s,T)v(\Phi_\tau(x,s,T),T).$$

Let F be an integrating factor for ρ with $F(0)=1$, and assume F is finite. If $T(\mathcal{T}) = t$, we see that $T = s + (t-s)F(\tau)/F(\mathcal{T})$, and so $T(0) = s + (t-s)F^{-1}(\mathcal{T}) := s_\mathcal{T}$. Then

$$\frac{\mathrm{d}\Phi_\tau}{\mathrm{d}T} = \frac{\mathrm{d}\Phi_\tau}{\mathrm{d}\tau}\frac{\partial\tau}{\partial T} = v(\Phi_\tau, T),$$

so Φ_τ flows the velocity field v. Since at $\tau = 0$, $T(0) = s_\mathcal{T}$ and $\Phi_\tau(x,s,T) = \Phi_0(x,s,s_\mathcal{T})$, we get that

$$\Phi_\tau(x,s,t) = \Phi^\star(\Phi_0(x,s,s_\tau), s_\tau, t). \tag{31}$$

If $F^{-1}(\tau) \to 0$, then $s_\tau \to s$, and so $\Phi_0(x,s,s_\tau) \to x$ since u_0 is locally bounded. Therefore under the continuity of $\Phi^\star$, $\Phi \to \Phi^\star$ pointwise. □

B.2 Convergence rates

Theorem 2. *Assume Assumptions 1 and 2 hold. In addition, assume $\rho(x,s,t) \geq \rho_{\min} > 0$ and $\|u_0\|_\infty, \|v\|_\infty \leq M$. Then the MeanFlow closed-form satisfies*

$$\|\Phi^{\mathrm{MF}}_\tau(x,s,t) - \Phi^\star(x,s,t)\|_\infty \leq 2Me^{-\rho_{\min}\tau}\ . \tag{15}$$

In addition, if v is uniformly L-Lipschitz for all t, then the Lagrangian closed-form satisfies

$$\|\Phi^{\mathrm{L}}_\tau(x,s,t) - \Phi^\star(x,s,t)\|_\infty \leq 2Me^{L}e^{-\rho_{\min}\tau}\ . \tag{16}$$

Proof. For the MeanFlow closed form, we have that

$$\begin{aligned}|\Phi^{\mathrm{MF}}_\tau(x,s,t) - \Phi^\star(x,s,t)| &= |u_0(\Phi^\star(x,s,t_\tau),t_\tau,t)(t-s)F^{-1}(\tau) + \Phi^\star(x,s,t_\tau) - \Phi^\star(x,s,t)|\\ &\leq (t-s)F^{-1}(\tau)\left(\|u_0\|_\infty + \frac{1}{t-t_\tau}\int_{t_\tau}^{t}\|v(\Phi^\star(x,s,r),r)\|_\infty \mathrm{d}r\right)\\ &\leq 2M(t-s)F^{-1}(\tau).\end{aligned}$$

If $\rho \geq \rho_{\min}$, then $\dot{F} = F\rho \geq F\rho_{\min}$ and $F(0)=1$ gives, by Grönwall, $F(\tau) \geq e^{\rho_{\min}\tau}$, and hence $F^{-1}(\tau) \leq e^{-\rho_{\min}\tau}$. This bound is uniform over all (x,s,t): taking a supremum over all x,s,t finishes. □

A bound on the convergence for the Lagrangian solution requires stronger assumptions. The closed-form solution in Equation (31) shows the Lagrangian-trained flow map is $\Phi^\star(\Phi_0(x,s,s_\tau), s_\tau, t)$. To have this be close to $\Phi^\star(x,s,t) = \Phi^\star(\Phi^\star(x,s,s_\tau), s_\tau, t)$, we need $\Phi_0(x,s,s_\tau)$ close to $\Phi^\star(x,s,s_\tau)$, and a Lipschitz bound on the flow map depending on $\Phi^\star$.

Note that a Lipschitz bound on the flow map follows from a Lipschitz bound on the velocity: if v is uniformly L-Lipschitz in x for all t, then from Cauchy-Schwarz, we can obtain

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}|\Phi^\star(x,s,t) - \Phi^\star(y,s,t)| &\leq |v(\Phi^\star(x,s,t),t) - v(\Phi^\star(y,s,t),t)|\\ &\leq L|\Phi^\star(x,s,t) - \Phi^\star(y,s,t)|,\\ |\Phi^\star(x,s,s) - \Phi^\star(y,s,s)| &= |x-y|,\end{aligned}$$

which tells us that the flow map is $e^{L(t-s)}$ Lipschitz for fixed (s,t) by Grönwall.

Proof. First, we obtain a bound for the closeness of $\Phi_0(x,s,s_\tau)$ and $\Phi^\star(x,s,s_\tau)$, in a similar manner to above:

$$\begin{aligned}|\Phi_0(x,s,s_\tau)-\Phi^\star(x,s,s_\tau)| &\le |\Phi_0(x,s,s_\tau)-x| + |\Phi^\star(x,s,s_\tau)-x| \\ &\le 2M(t-s)F^{-1}(\tau),\end{aligned}$$

from calculations similar to the above. Then the Lipschitzness of $\Phi^\star(\cdot,s_\tau,t)$ gives

$$|\Phi^{\mathrm{L}}_\tau(x,s,t)-\Phi^\star(x,s,t)| \le 2M(t-s)e^{L(t-s_\tau)}F^{-1}(\tau).$$

Again, if $\rho \ge \rho_{\min}$, we get $F^{-1}(\tau) \le e^{-\rho_{\min}\tau}$, which proves the claim. □

We can show that the Lipschitz factor is required for the pointwise convergence of the Lagrangian solution. Take $\rho = 1$, and the velocity

$$v(x,t) = \begin{cases} 1 & x \ge 1/L, \\ Lx & x \in [-1/L, 1/L], \\ -1 & x \le -1/L. \end{cases} \tag{32}$$

The corresponding flow map is $\Phi^\star(x,s,t) = e^{L(t-s)}x$ for $|x| < e^{-L}$. Suppose we have initialization $u_0(x,s,t) = 1$. Then the closed form gives that, for sufficiently large τ,

$$\begin{aligned}\Phi^{\mathrm{L}}_\tau(0,0,1) &= \Phi^\star(\Phi(0,0,e^{-\tau}),e^{-\tau},1) = \Phi^\star(e^{-\tau},e^{-\tau},1) \\ &= e^{L(1-e^{-\tau})}e^{-\tau}\end{aligned}$$

whereas $\Phi^\star(0,0,1) = 0$. This gives separation of $e^L e^{-\tau}$, which is tight against our upper bound up to a constant. We can verify the meanflow map is $\Phi^{\mathrm{MF}}_\tau(0,0,1) = \Phi_0(0,1-e^{-\tau},1) = e^{-\tau}$, which avoids the e^L factor.

L^2 convergence. For the following proof, it will be useful to define the *material derivative* operator $\mathcal{D}_v = (v\cdot\nabla + \partial_s)$. Then the meanflow PDE is equivalent to

$$\partial_\tau u_\tau = -\rho(u_\tau - v - (t-s)\mathcal{D}_v u_\tau). \tag{33}$$

We then note that, for an arbitrary field u (not necessarily satisfying the meanflow PDE),

$$\begin{aligned}\mathcal{D}_v u(\Phi^\star(z,0,s),s) &= v(\Phi^\star(z,0,s),s)\cdot\nabla u(\Phi^\star(z,0,s),s) + \partial_s u(\Phi^\star(z,0,s),s) \\ &= \partial_s\Phi^\star(z,0,s)\cdot\nabla u(\Phi^\star(z,0,s),s) + \partial_s u(\Phi^\star(z,0,s),s) \\ &= \frac{\mathrm{d}}{\mathrm{d}s}u(\Phi^\star(z,0,s),s).\end{aligned}$$

Theorem 3. *Assume Assumptions 1 to 3 hold, and $\rho(x,s,t) \ge \rho_{\min} > 0$. If $\|u_0^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} < \infty$ then the MeanFlow trained average velocity u_τ^{MF} satisfying Equation (10) fulfills:*

$$\|u_\tau^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} \le e^{-\rho_{\min}\tau}\|u_0^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)}\ , \tag{18}$$

hence, the average learned velocity u_τ^{MF} converges to $u^\star$ with respect to the $L^2(p_s/\rho)$-norm

Proof. We note that $u_\tau - u^\star$ follows the PDE

$$\begin{aligned}\frac{\partial}{\partial\tau}(u_\tau - u^\star) &= -\rho\left(u_\tau - v - (t-s)\mathcal{D}_v u_\tau\right) \\ &= -\rho\left(u_\tau - u^\star + (t-s)\mathcal{D}_v u^\star - (t-s)\mathcal{D}_v u_\tau\right) \\ &= -\rho(I - (t-s)\mathcal{D}_v)(u_\tau - u^\star).\end{aligned}$$

Now, we may differentiate under the integral sign, due to the regularity offered by Assumption 3:

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}\tau}\int \frac{p_s(x)}{\rho(x,s,t)}\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t \\ &\quad= -\int 2p_s(x)(I-(t-s)\mathcal{D}_v)(u_\tau-u^\star)\cdot(u_\tau-u^\star)\mathrm{d}x\mathrm{d}s\mathrm{d}t \\ &\quad= -\int p_s(x)(2I-(t-s)\mathcal{D}_v)\|u_\tau-u^\star\|^2\mathrm{d}x\mathrm{d}s\mathrm{d}t \\ &\quad= -2\|u_\tau-u^\star\|^2_{L^2(p_s)} + \int p_s(x)(t-s)\mathcal{D}_v\|u_\tau-u^\star\|^2\mathrm{d}x\mathrm{d}s\mathrm{d}t.\end{aligned}$$

To analyze the last term we apply a change of variable $x = \Phi^\star(z, 0, s)$, noting that if $x \sim p_s$, then $z \sim p_0$:

$$\begin{aligned}
&\int p_s(x)(t-s)\mathcal{D}_v\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t \\
&\qquad = \int p_0(z)(t-s)\mathcal{D}_v\|u_\tau(\Phi^\star(z,0,s),s,t) - u^\star(\Phi^\star(z,0,s),s,t)\|^2 \mathrm{d}z\mathrm{d}s\mathrm{d}t \\
&\qquad = \int p_0(z)(t-s)\frac{\mathrm{d}}{\mathrm{d}s}\|u_\tau(\Phi^\star(z,0,s),s,t) - u^\star(\Phi^\star(z,0,s),s,t)\|^2 \mathrm{d}s\mathrm{d}z\mathrm{d}t \\
&\qquad = \left[\int p_0(z)(t-s)\|u_\tau(\Phi^\star(z,0,s),s,t) - u^\star(\Phi^\star(z,0,s),s,t)\|^2 \mathrm{d}z\mathrm{d}t\right]_{s=0}^{t} \\
&\qquad\quad + \int p_0(z)\|u_\tau(\Phi^\star(z,0,s),s,t) - u^\star(\Phi^\star(z,0,s),s,t)\|^2 \mathrm{d}z\mathrm{d}s\mathrm{d}t \\
&\qquad = -\int t p_0(x)\|u_\tau(x,0,t) - u^\star(x,0,t)\|^2 \mathrm{d}x\mathrm{d}t + \|u_\tau - u^*\|^2_{L^2(p_s)}.
\end{aligned}$$

Combining with the other part gives

$$\begin{aligned}
\partial_\tau\|u_\tau - u^\star\|^2_{L^2(p_s/\rho)} &= \frac{\partial}{\partial\tau}\int \frac{p_s(x)}{\rho(x,s,t)}\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t \\
&= -\int p_s(x)\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t - \int t p_0(x)\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}t \\
&\le -\rho_{\min}\int \frac{p_s(x)}{\rho(x,s,t)}\|u_\tau - u^\star\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t = -\rho_{\min}\|u_\tau - u^\star\|^2_{L^2(p_s/\rho)}.
\end{aligned}$$

Hence we have that $\|u_\tau - u^\star\|^2_{L^2(p_s/\rho)} \le e^{-\rho_{\min}\tau}\|u_0 - u^\star\|^2_{L^2(p_s/\rho)}$. □

Although the assumption that $\rho \ge \rho_{\min} > 0$ cannot hold for a density over $\mathbb{R}^d$, the same inequality holds when integrating over a compact domain $\{(\Phi^*(x,a,s),s,t) \mid x \in K, a \le s \le t \le b\}$. Moreover if $\rho \le \rho_{\max} < \infty$, then $\|u_\tau - u^\star\|^2_{L^2(p_s/\rho)} \ge \rho_{\max}^{-1}\|u_\tau - u^\star\|^2_{L^2(p_s)}$, hence $u_\tau \to u^\star$ in $L^2(p_s)$.

B.3 Sufficient conditions for convergence

We now state sufficient conditions for which the above proofs are valid. The closed-form and L^∞ proofs rely on the local existence and uniqueness of characteristics X, S and T, and the application of the chain rule. This follows from the Picard-Lindelöf theorem under certain assumptions on the velocity v and the density ρ:

- v is locally Lipschitz in x, s.
- ρ is locally Lipschitz in x, s, t.

This gives local existence and uniqueness. We can extend these local characteristics until they either approach a boundary or $X \to \infty$. The latter cannot happen if the true flow $\Phi^\star$ is well-defined everywhere. We analyse conditions for the former to hold below.

The above derivations only work if the quantity F does not diverge in finite time, otherwise the characteristics hit the boundary $s = t$, where the PDE is unspecified. If ρ is upper bounded by M, then this holds, since F satisfies the PDE $\dot{F} = \rho F \le MF$ and $F(0) = 1$, giving $F(\tau) \le e^{M\tau} < \infty$.

If we suspend the assumption that ρ must be locally Lipschitz everywhere, the following proposition characterizes the choices of ρ for which the characteristics satisfy $s < t$ for all τ.

Proposition 2. *Suppose that $\rho(x,s,t) \asymp g(t-s)$, where g^{-1} is bounded outside of a neighbourhood of 0 and ρ is bounded below on trajectories. Then $F(\tau) < \infty$ for all τ if and only if*

$$\int_0^1 \frac{1}{r g(r)}\mathrm{d}r = \infty. \tag{34}$$

Choosing $g(r) = 1$ ensures convergence while having ρ remain bounded.

Proof. Note that showing $F(\mathcal{T}) < \infty$ is equivalent to showing $S(0) < t$ or $T(0) > s$ for any choice of $\mathcal{T}$, depending on whether we are in the Eulerian or Lagrangian frame. Define a new variable $R(\tau)$ by

$$R(\tau) = \begin{cases} t - S(\mathcal{T} - \tau) & \text{under the Eulerian loss,} \\ T(\mathcal{T} - \tau) - s & \text{under the Lagrangian loss.} \end{cases} \tag{35}$$

Then we can note that in both formulations, $\dot{R} = -\rho(X, S, T)R$ and $R(0) = t - s$. Letting τ_0 be the time until R hits 0 (which is possibly ∞), we obtain that

$$\int_0^{t-s} \frac{1}{r\rho(r)} \mathrm{d}r = \tau_0.$$

Assuming that $\rho(X, S, T) \asymp g(T - S) = g(R)$ gives that

$$\tau_0 \asymp \int_0^{t-s} \frac{1}{rg(r)} \mathrm{d}r,$$

hence τ_0 is finite if and only if the integral is finite, noting that $0 < t - s \leq 1$, and the integral from $t - s$ to 1 is finite by the boundedness of g^{-1} away from 0. □

B.4 Gradient descent proofs

For the gradient descent iterates u_k, let $\Phi_k(x, s, t) = x + (t - s)u_k(x, s, t)$.

Proposition 1. *For every $\eta > 0$, $\varepsilon > 0$, and $\rho \in C^\infty$, there exists u_0 whose initial error $u_0 - u^\star \in L^2(\rho) \cap C_c^\infty$ with $\|u_0 - u^\star\|_\infty \leq \varepsilon$, but whose iterates under Equation (19) diverge for either $\mathcal{L} = \mathcal{L}^{\mathrm{MF}}$, or $\mathcal{L}^{\mathrm{L}}$ with $v = 0$.*

Proof. We first write down the recurrence in terms of the errors $e_k = u_k - u^\star$, using the fact that $u^\star - v - (t - s)\mathcal{D}_v u^\star = 0$, and $u^\star - v(\Phi^\star) + (t - s)\partial_t u^\star = 0$. For the MeanFlow loss,

$$e_{k+1} = e_k - \eta\rho(e_k - (t - s)\mathcal{D}_v e_k), \tag{36}$$

and for the Lagrangian loss,

$$e_{k+1} = e_k - \eta\rho[e_k + (t - s)\partial_t e_k - (v(\Phi_k, t) - v(\Phi^\star, t))]. \tag{37}$$

We give an example of a small error blowing up. Consider $\rho = 1$ and

$$e = e^0 = \varepsilon(t - s)^m a \tag{38}$$

for some value $m \in \mathbb{N}$ with $\eta(m+1) > 2$ and a constant vector a with $\|a\| = 1$. Then $-(t-s)\mathcal{D}_v e = (t-s)\partial_t e = m(t-s)^m = me$, hence $e^{k+1} = (1-\eta(m+1))e^k$ by induction, for any velocity v under the meanflow loss, and for $v = 0$ under the Lagrangian loss. So $e^k = \varepsilon(1 - \eta(m + 1))^k(t - s)^m a$. Since $\eta > 2(m + 1)^{-1}$, this diverges as $k \to \infty$.

To construct $e^0 \in L^2(\rho)$, we can simply multiply by a smooth cutoff function which is equal to one on the neighbourhood of a compact set K. Since this update is local, on K we will still have $e^k = \varepsilon(1 - \eta(m + 1))^k(t - s)^m a$, and hence still obtain divergence in both $L^2(\rho)$ and L^∞.

We now generalize our construction to smooth ρ. We pick some point (x, s, t) on which $\rho > 0$. By continuity, there is a neighborhood around this point on which ρ is positive.

Within this neighborhood, we choose $V = \{(\Phi(x, r, s), s, t) \mid x \in K, s \in [s_1, s_2], t \in [t_1, t_2]\}$, with $t_1 > s_2$. We would like to find an initial error e_0 such that

$$\rho(e_0 - (t - s)\mathcal{D}_v e_0) = me_0 \implies \mathcal{D}_v e_0(x, s, t) = \frac{1 - m/\rho}{(t - s)} e_0(x, s, t), \tag{39}$$

at least in some neighborhood. Consider choosing $e_0(x, s, t) = 1$ for all $(x, s, t) \in \{(\Phi(x, s, s_1), s_1, t) \mid x \in K, t \in [t_1, t_2]\}$. Then, we may define e_0 implicitly on the rest of V since

$$\mathcal{D}_v e_0(\Phi(x, r, s), s, t) = \frac{\mathrm{d}}{\mathrm{d}s} e_0(\Phi(x, r, s), s, t). \tag{40}$$

It can also be defined explicitly through

$$e_0(\Phi(x,r,s),s,t) = C_\varepsilon \exp\left(\int_{s_1}^{s} \frac{1 - m/\rho(\Phi(x,r,s'),s',t)}{t-s'}\mathrm{d}s'\right) a, \tag{41}$$

where a is a constant vector with norm 1, and C_ε is carefully chosen to bound the L_∞ norm of e_0 on V by ϵ. This suffices as an example of e_0 satisfying $\rho(e_0 - (t-s)\mathcal{D}_v e_0) = me_0$, at least within V. Hence we obtain that $e_{k+1} = (1-\eta m)e_k$, and so if $\eta > \frac{2}{m}$, then this diverges within V.

Again, we may make this well-defined by multiplying by a cutoff function that is one on a neighburhood of a compact subset K of V, but now we must also assume this cutoff is 0 on the exterior of some open subset of V, so that the local e_0 counterexample above extends to a globally defined counterexample. Moreover e_0 must be non-zero somewhere, since it is 1 at s_1, so we obtain that its iterates must diverge in both L_∞ and L_2 norms.

A similar proof works for the Lagrangian loss with $v = 0$, which we do not detail here. □

This counterexample for the MeanFlow loss is independent of the flow map we are trying to learn. The phenomenon that increasingly small η is required to fully learn all modes of the solution is due to the inherent *stiffness* of the MeanFlow and Lagrangian PDE.

Theorem 4. *Assume that the velocity v, the density ρ and u_0 fulfill Assumptions 1 and 2. In addition, assume that the iterates of gradient descent satisfy the discrete equivalent of Assumption 3, and that p_s is bounded above: $p_s(x) \le p_{\max} < \infty$, ρ is bounded below: $\rho(x,s,t) \ge \rho_{\min} > 0$ and $\|u_0 - u^\star\|^2_{L^2(p_s/\rho)} < \infty$. Consider the update scheme given in Equation (20).*

- *For $p \ge 0.5$, the updates have uniformly bounded norms: $\|u_{k+1} - u_k\|^2_{L^2(p_s/\rho)} \le M < \infty$.*
- *$p = 1$ yields convergence for the MeanFlow objective: if $\|u_k - u_{\mathrm{tgt},k}\|^2_{L^2(\rho)} \le M(k+1)$ for some $M > 0$, i.e. the norm of the residual grows at most linearly, then*

$$\limsup_{k\to\infty} \|u_k^{\mathrm{MF}} - u^\star\|^2_{L^2(p_s/\rho)} \le \eta p_{\max}(\rho_{\min}^{-1} + \eta c^{-1}) \ . \tag{21}$$

Proposition 3. *Assume Assumptions 1 to 3 hold. Consider the update scheme given in Equation (20). Assume that p_s is bounded above: $p_s(x) \le p_{\max} < \infty$. Suppose $\|u_0 - u^\star\|^2_{L^2(p_s/\rho)} < \infty$. Then for $p \ge 1/2$, the updates have uniformly bounded norms:*

$$\|u_{k+1} - u_k\|^2_{L^2(p_s/\rho)} \le M < \infty. \tag{42}$$

Proof. Note that

$$\|u_{k+1} - u_k\|^2_{L^2(p_s/\rho)} = \left\| \eta\rho \frac{r_k}{(\|r_k\|^2_{L^2(\rho)} + c)^p} \right\|^2_{L^2(p_s/\rho)} \tag{43}$$

$$= \eta^2 \int_{\mathbb{R}^d\times\Delta} \frac{p_s}{\rho}\rho^2 \frac{\|r_k\|^2}{(\|r_k\|^2_{L^2(\rho)} + c)^{2p}} \mathrm{d}x\mathrm{d}s\mathrm{d}t \tag{44}$$

$$\le \eta^2 \frac{p_{\max}}{(\|r_k\|^2_{L^2(\rho)} + c)^{2p}} \int_{\mathbb{R}^d\times\Delta} \rho\|r_k\|^2 \mathrm{d}x\mathrm{d}s\mathrm{d}t \tag{45}$$

$$= \eta^2 p_{\max} \frac{\|r_k\|^2_{L^2(\rho)}}{(\|r_k\|^2_{L^2(\rho)} + c)^{2p}}. \tag{46}$$

The function $f(x) = \frac{x}{(x+c)^{2p}}$ on $x \ge 0$ is bounded for $p \ge 1/2$, since it is continuous with $f(0) = 0$ and $f(x) \le 1$ for $x \ge 1$. □

Proposition 4. *Assume Assumptions 1 to 3 hold. Consider the update scheme given in Equation (20), with $p = 1.0$. Assume that on a subset $K^+ = \{(\Phi^\star(x,a,s),s,t) : x \in K, a \le s \le t \le b\}$, we have that p_s is bounded above: $p_s(x) \le p_{\max} < \infty$, and ρ is bounded below: $0 < \rho_{\min} \le \rho$. Then we obtain*

$$\|u_{k+1} - u^\star\|^2_{L^2(p_s/\rho)} \le \max(\|u_k - u^\star\|^2_{L^2(p_s/\rho)}, \eta p_{\max}(\rho_{\min}^{-1} + \eta c^{-1})). \tag{47}$$

Under the assumption that there exists M for which $\|r_k\|^2_{L^2(\rho)} \leq M(k+1)$ for each k, for all k large enough we obtain $\|u_k - u^\star\|^2_{L^2(p_s/\rho)} \leq \eta p_{\max}(\rho_{\min}^{-1} + \eta c^{-1})$.

Proof. Note that

$$e_{k+1} = e_k - \eta\rho \frac{r_k}{\|r_k\|^2_{L^2(\rho)} + c}, \tag{48}$$

so with $\mathbf{x} = (x, s, t)$, we have that

$$\int \frac{p_s}{\rho}\|e_{k+1}\|^2 \mathrm{d}\mathbf{x} = \int \frac{p_s}{\rho}\|e_k\|^2 \mathrm{d}\mathbf{x} - 2\eta \int \frac{p_s}{\|r_k\|^2_{L^2(\rho)} + c} e_k \cdot r_k \mathrm{d}\mathbf{x} \tag{49}$$

$$+ \eta^2 \int p_s \rho \frac{\|r_k\|^2}{(\|r_k\|^2_{L^2(\rho)} + c)^2} \mathrm{d}\mathbf{x}. \tag{50}$$

Under the assumption that $p_s \leq p_{\max}$, the final term can be bounded as $\eta^2 p_{\max}/(\|r_k\|^2_{L^2(\rho)} + c)$. For the former, we can use a change of variables, the expansion $r_k = u_k - v - (t-s)\mathcal{D}_v u_k = e_k - (t-s)\mathcal{D}_v e_k$, and integration by parts to notice that

$$\begin{aligned}\int_{K^+} p_s e_k \cdot (e_k - (t-s)\mathcal{D}_v e_k)\mathrm{d}\mathbf{x} &= \int p_s \|e_k\|^2 \mathrm{d}\mathbf{x} - \frac{1}{2}\int (t-s) p_a \frac{\mathrm{d}}{\mathrm{d}s}\|e_k\|^2 \mathrm{d}\mathbf{x} \\ &= \frac{1}{2}\int p_s \|e_k\|^2 \mathrm{d}\mathbf{x} + \frac{1}{2}\int (t-a) p_a \|e_k\|^2 \mathrm{d}x\mathrm{d}t.\end{aligned}$$

Hence we get the inequality

$$\begin{aligned}\|e_{k+1}\|^2_{L^2(p_s/\rho)} - \|e_k\|^2_{L^2(p_s/\rho)} &\leq -\frac{\eta}{\|r_k\|^2_{L^2(\rho)} + c}\left(\int p_s \|e_k\|^2 \mathrm{d}\mathbf{x} - \eta p_{\max}\right) \\ &\leq -\frac{\eta}{\|r_k\|^2_{L^2(\rho)} + c}\left(\rho_{\min}\int \frac{p_s}{\rho}\|e_k\|^2 \mathrm{d}\mathbf{x} - \eta p_{\max}\right) \\ &\leq -\frac{\eta}{\|r_k\|^2_{L^2(\rho)} + c}\left(\rho_{\min}\|e_k\|^2_{L^2(p_s/\rho)} - \eta p_{\max}\right).\end{aligned}$$

In particular, if $\|e_k\|^2_{L^2(p_s/\rho)} \geq \eta\frac{p_{\max}}{\rho_{\min}}$, then $\|e_{k+1}\|^2_{L^2(p_s/\rho)} \leq \|e_k\|^2_{L^2(p_s/\rho)}$. If $\|e_k\|^2_{L^2(p_s/\rho)} \leq \eta\frac{p_{\max}}{\rho_{\min}}$, then

$$\|e_{k+1}\|^2_{L^2(p_s/\rho)} \leq \|e_k\|^2_{L^2(p_s/\rho)} + \frac{\eta^2 p_{\max}}{\|r^k\|^2_{L^2(\rho)} + c} \leq \eta p_{\max}\left(\frac{1}{\rho_{\min}} + \frac{\eta}{c}\right).$$

Therefore, once we enter the ball with squared radius $\eta p_{\max}(\rho_{\min}^{-1} + \eta c^{-1}) = \mathcal{O}(\eta)$, we always stay within it.

Let $\|e_k\|^2_{L^2(p_s/\rho)} = T_k + \eta p_{\max}\rho_{\min}^{-1}$. Then,

$$T_{k+1} \leq T_k\left(1 - \frac{\eta\rho_{\min}}{\|r_k\|^2_{L^2(\rho)} + c}\right).$$

Suppose $T_k \geq 0$ for all k, otherwise we have already entered the energy ball. Then, we must have $1 - \frac{\eta\rho_{\min}}{\|r_k\|^2_{L^2(\rho)}+c} \geq 0$, and so through telescoping, we can derive the inequality

$$\begin{aligned}T_{k+1} &\leq T_0 \prod_{i=0}^{k}\left(1 - \frac{\eta\rho_{\min}}{\|r_i\|^2_{L^2(\rho)} + c}\right) \leq T_0 \prod_{i=0}^{k}\exp\left(-\frac{\eta\rho_{\min}}{\|r_i\|^2_{L^2(\rho)} + c}\right) \\ &= T_0 \exp\left(-\eta\rho_{\min}\sum_{i=0}^{k}\frac{1}{\|r_i\|^2_{L^2(\rho)} + c}\right).\end{aligned}$$

Hence under the assumption

$$\sum_{i=0}^{\infty} \frac{1}{\|r_i\|^2_{L^2(\rho)} + c} = \infty,$$

we will have that $T_k \to 0$ as $k \to \infty$. Hence E_k is confined to a ball of squared radius $\mathcal{O}(\eta)$. Finally, this inequality holds true if $||r_k||^2_{L^2(\rho)} \leq M(k+1)$ for all k. □

This says that we converge under the assumption that the *effective total step size*

$$\sum_{i=0}^{\infty} \frac{1}{\|r_i\|^2_{L^2(\rho)} + c}$$

is infinite.

For $p > 1$ it is likely possible to formulate similar convergence guarantees, however the assumptions on the residual norms will be stricter.

B.5 Equivalence to Align Your Flow

We show that Equations (5) and (7) have the same gradient update as the losses in Align Your Flow (Sabour et al., 2025). This is already proven for the AYF-EMD objective and the Meanflow loss in Section D.5 of Sabour et al. (2025).

For the AYF-LMD objective, note that

$$\nabla_\theta \mathcal{L}^{\text{AYF-LMD}} = \nabla_\theta \mathbb{E}\left[w(s,t)(t-s) f_\theta(x,s,t)^\top \text{sg}\left(\partial_t f_\theta(x,s,t) - v(f_\theta(x,s,t),t)\right)\right]. \tag{51}$$

Using the parameterisation

$$f_\theta(x,s,t) = x + (t-s)u_\theta(x,s,t),$$

we have

$$\begin{aligned} \nabla_\theta f_\theta &= (t-s)\nabla_\theta u_\theta, \\ \partial_t f_\theta - v(f_\theta, t) &= u_\theta + (t-s)\partial_t u_\theta - v\left(x + (t-s)u_\theta, t\right). \end{aligned}$$

Therefore,

$$\nabla_\theta \mathcal{L}^{\text{AYF-LMD}} = \mathbb{E}\left[w(s,t)(t-s)^2 \nabla_\theta u_\theta^\top \text{sg}\left(u_\theta + (t-s)\partial_t u_\theta - v\left(x + (t-s)u_\theta, t\right)\right)\right]. \tag{52}$$

The term inside the stop gradient is precisely the residual of Equation (7). Hence AYF-LMD and our Lagrangian loss induce the same gradient update, up to their choice of loss weightings.

C Experimental Details

C.1 Additional experiments: validation of theoretical results

All experiments in this section were conducted on CPU.

Equivalence of experimental and theoretical trajectories. Figures 2 and 4 shows $1+2$ dimensional lookup tables with $129 \times 33 \times 33$ parameters trained from a cosine initialisation relaxing towards the true flow map. We choose a bounded region of space $x \in [-4, 5]$, and initialise $u_0(x,s,t) = 0.8\cos(1.3x + 2s) - 0.2t$. Choosing $p_0 = \mathcal{N}(0,1)$, $p_1 = \mathcal{N}(1, 0.6^2)$, the linear interpolant has $p_t \sim \mathcal{N}(t, (1-t)^2 + t^2 0.6^2)$. We take $\rho = 1$, and use learning rate 3×10^{-5} for both methods.

Effect of loss weighting on convergence. In order to see how loss weighting affects convergence, we redo the same convergence experiments with $\rho(x,s,t) = (t-s)^a$, for various values of a. In order for the updates to remain stable at $a = -1$, we use a coarser grid, of size $65 \times 17 \times 17$. Results are plotted in Figure 5. We choose an LR of 10^{-5} for $a = -1$, 3×10^{-5} for $a = -0.5$, and 3×10^{-4} for $a = 0$ and 1.

We can see that our derived convergence rates hold in practice. Moreover, for choices of $a < 0$ when interior characteristics are predicted to hit the boundary at finite time, convergence to the error floor happens at or shortly after this hitting time.

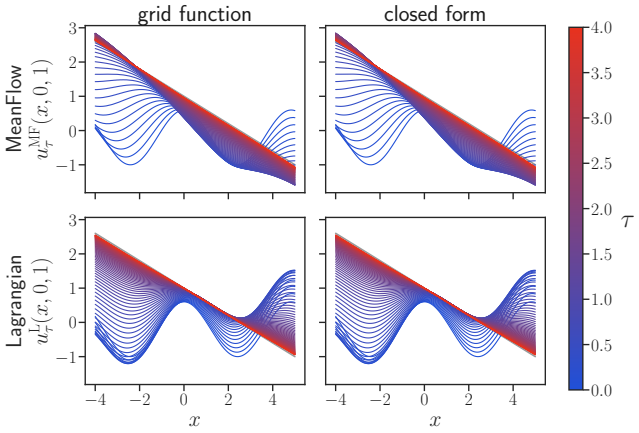


Figure 4: *Synthetic Gaussian-to-Gaussian dataset.* Each line corresponds to an optimization time τ for the empirically learned average velocity (left) and the derived closed-form average velocity (right). As the optimization time increases, the average velocities converge towards the true one $u^\star$. The two average velocities, empirically learned and the closed-form, show strong agreement across iteration times τ and for both the MeanFlow (top) and Lagrangian target (bottom).

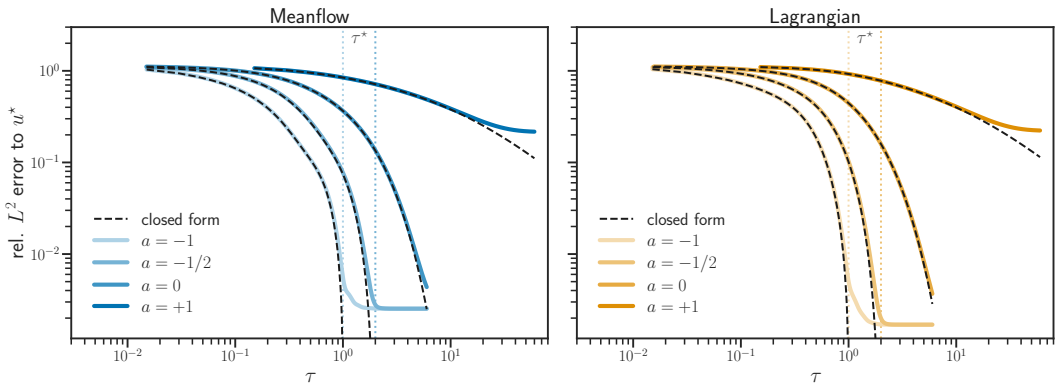


Figure 5: A comparison of the convergence rates of flow maps trained with different loss weightings $\rho = (t-s)^a$, versus their theorised trajectories. For $a = -1$ and $a = -0.5$, the theoretical last diagonal hitting time is marked.

To quantify the dependence of each loss weighting to the diagonal, we consider a situation where the diagonal is not τ-independent. Using the same experimental setup as with Figure 5, we now let $u_\tau(x,t,t) = v^\star(x,t) + \sin(\pi\tau)/2$, where $v^\star$ is the ground truth linear interpolant velocity.

Figure 6 shows a comparison of the L^2 errors for $a \leq 0$. We see that for $a = -0.5$ and $a = -1$, the loss is consistently higher than the $a = 0$ loss, which matches our prediction that the interior of the PDE tracks the boundary for $a < 0$, whereas for $a = 0$ the solution of the PDE in the interior should be independent of the evolution of the boundary. The failure for the $a = 0$ runs to recover the true interior map is a discretisation error that arises from our lookup table approximation.

Mode-learning dependence on η. We now show that learning higher modes is impossible with a fixed learning rate. Returning to the linear interpolant between $\mathcal{N}(0,1)$ and $\mathcal{N}(0, 0.6^2)$, we consider initialising our network with error $(t-s)^k$, for choices of $k = 2$, 8 and 24. We evaluate the loss of the meanflow evolution on a $64 \times 257 \times 257$ grid for different values of η.

Results are found in Figure 7. We see that if η is too large compared to $1/k$, the table is unable to learn, and the error grows.

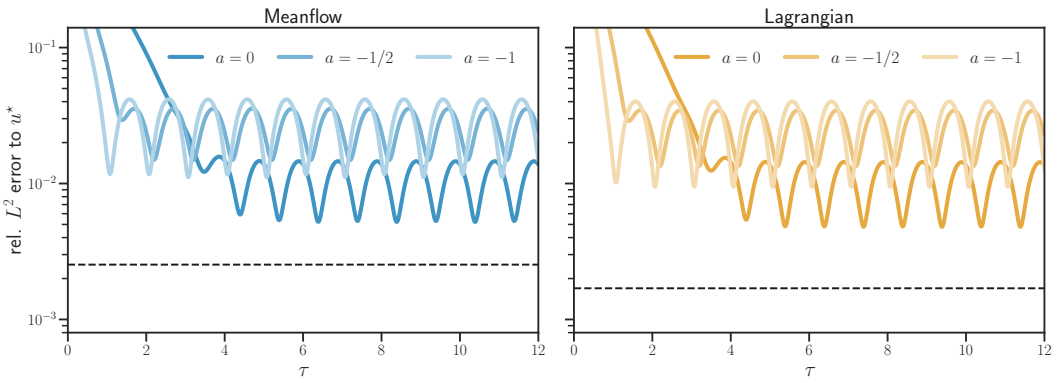


Figure 6: Instead of keeping the diagonal $u(x, s, s)$ fixed, we vary it with optimisation time, keeping it centred at the true velocity. The learnt flow maps for $a < 0$ have a higher MSE against the true flow map, compared to $a = 0$. We plot the errors achieved when the diagonal is fixed for comparison.

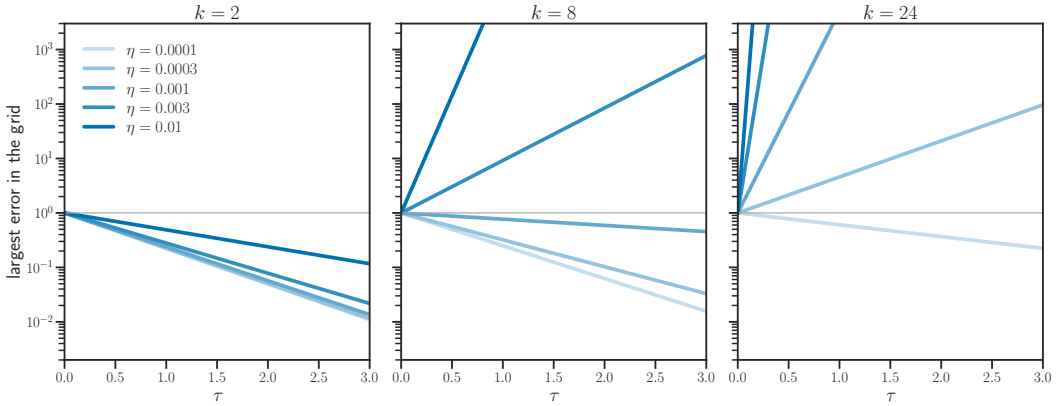


Figure 7: The choice of η impacts whether we learn an error of the form $(t - s)^k$. As k increases, a smaller η must be taken to ensure that gradient descent does not accumulate errors.

C.2 Additional experiments: qualitative behaviour

We aim to show a difference in practice between learning using the MeanFlow and the Lagrangian objective. All of the following experiments ran in 90 minutes on a single RTX 5090.

Dependence on velocity Lipschitzness. We have shown that pointwise and L^2 convergence rates for the meanflow objective are independent of the Lipschitz constant of the velocity v, whereas in the Lagrangian objective the convergence guarantees are exponentially weakened by the Lipschitz constant L. To test whether this effect is visible in practice, we train neural flow map models to follow increasingly expansive velocities using both objectives, and measure their relative L^2 error to the true average velocity $u^\star$.

We train MLPs with Adam to learn the flow map associated to the velocity $v(x, t) = \lambda x$ for values of λ in $[0, 4]$, starting from the 64 dimensional Gaussian distribution $\mathcal{N}(0, I)$. We plot the mean relative L^2 error obtained over 5 seeds along with min-max bands in Figure 8. The meanflow objective achieves lower error on this task than the Lagrangian objective over all values of λ considered, which is consistent with the convergence rate theory that we have developed. For values of λ above 3.5, the Lagrangian objective fails to learn the flow map.

Dependence on velocity Lipschitzness. We have shown that pointwise convergence for the meanflow objective is independent of the Lipschitz constant of the velocity v, whereas in the Lagrangian objective the convergence guarantees are exponentially weakened by the Lipschitz constant L.

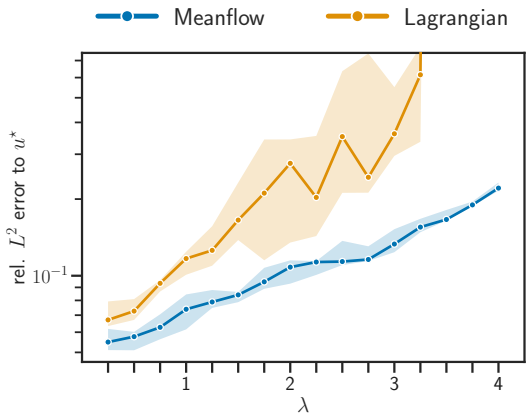


Figure 8: For the linear flow $v(x,t) = \lambda x$, we report the average relative $L^2(p_s)$ error over 5 seeds for average velocity networks against the ground truth $u^\star = (e^{\lambda(t-s)} - 1)x/(t-s)$. Lagrangian is worse than Meanflow for every $\lambda > 0$, and is unstable at high λ.

Table 2: Training setup for Figure 8

Component	**Setting**
Problem	$p_0 = \mathcal{N}(0, I_{64})$, $v(x,t) = \lambda x$, $\lambda \in \{0.25, 0.5 \ldots, 4\}$
Architecture	MLP 3×256, SiLU, sinusoidal s, t embeddings
Optimisation	Adam, lr 10^{-4}, batch 1024, 60k steps
Time sampling	Uniform on $0 \le s \le t \le 1$
State sampling	$x_s \sim \mathcal{N}(0, e^{2s\lambda} I)$
Normalisation	Scale $= e^{\lambda}$

We provide here the experimental details for the MLP experiments from Section 5. Notably we scale the inputs by $e^{-\lambda}$ to make them unit variance, and scale the network outputs by e^{λ}. The L^2 error is computed over 30k samples sampled from p_s, fixed over seeds and objective.

Mode recovery. A realistic example of non-Lipschitz flows occurs when the data distribution consists of distinct modes, separated by very low density regions. Points in between the modes will get mapped onto the high density regions, making the one-step map have high Lipschitz constant near the boundary of each mode's preimage.

To test this, we compared the meanflow and Lagrangian training schemes on a 64 dimensional mixture of 8 Gaussians, which have maximally separated means at radius 8, and with standard deviation 0.1. The Gaussians are weighted with probabilities $[8/30, 8/30, 4/30, 4/30, 2/30, 2/30, 1/30, 1/30]$, to test the effect that unequal weighting has on the recovery of modes.

The L^2 loss and mode recovery is measured on 1500 samples, where the correct mode is determined by integrating the closed form velocity over 256 steps.

Figure 9 shows t-SNE plots of the trained one-step maps. We also vary the diagonal percentage. When no samples are explicitly taken at the diagonal, the meanflow trained model has significantly better mode recall, which assesses the percentage of points for which the closest mode according to the model matches the closest mode under the true flow map. Meanflow takes 99.0% of points to their own mode, while only 81.9% are transported to the correct mode under Lagrangian training. However, the points under the meanflow loss are considerably more dispersed than in the Lagrangian case, leading to a larger relative L^2.

When training with 50% of the points at the diagonal, the relative L^2 of the meanflow model decreases substantially, overtaking the Lagrangian model. Hence a larger fraction of points at the diagonal during training can be correlated with more concentrated samples. This is while only misrouting an additional 0.8% of all points. While the L^2 and mode recall of the Lagrangian method improves, it lags behind the meanflow model.

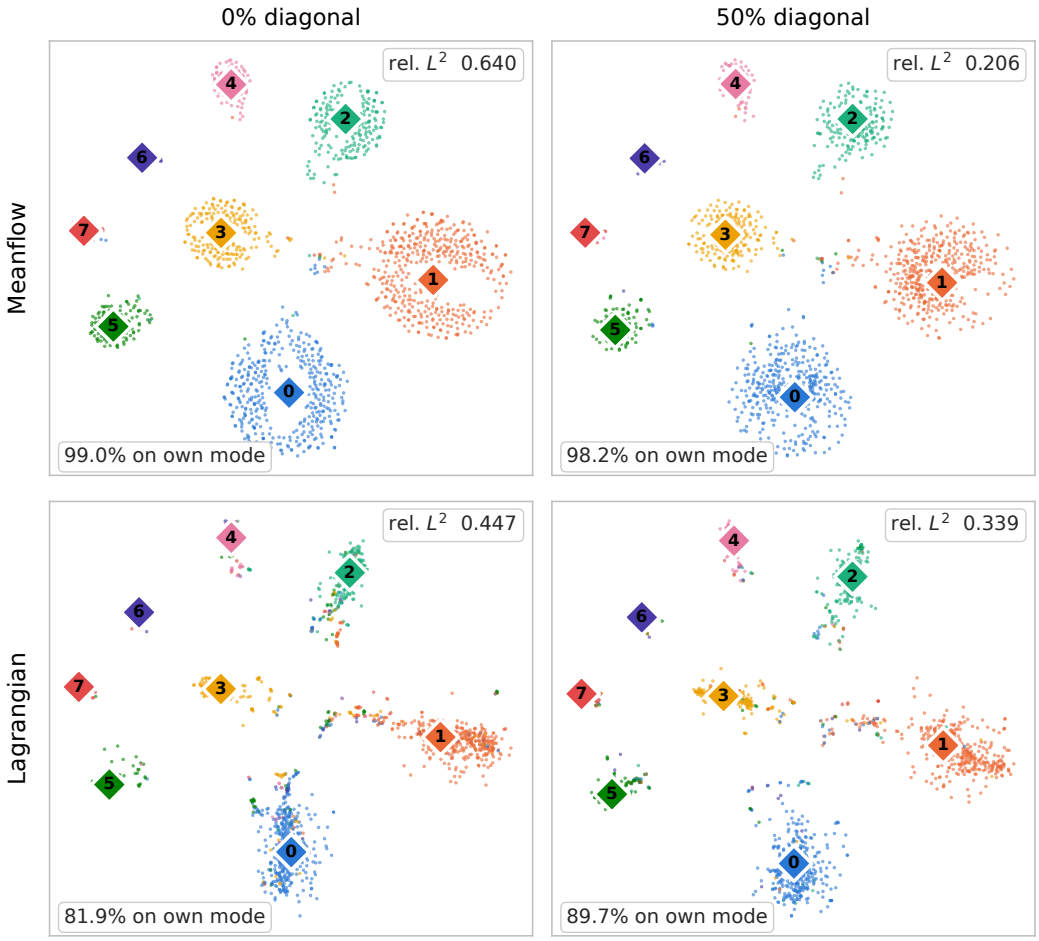


Figure 9: Comparison of the performance of the meanflow and Lagrangian training methods on an unequal mixture of eight Gaussians, split by % of diagonal samples taken. Lagrangian is closer to the centre of each mode, while routing more points to the incorrect mode. Increasing the % of samples at the diagonal makes the meanflow samples more concentrated around each mode.

Table 3: Training setup for the multimodal experiment.

Component	**Setting**
Architecture	MLP 3×256, SiLU, 4-frequency time embeddings
Optimisation	Adam, lr $2 \times 10^{-3} \to 10^{-5}$ cosine, batch 1024, 20k steps
Time sampling	$s \sim \mathcal{U}[0,1]$, $t = s + (1-s)\xi$, $\xi \sim \mathcal{U}[0,1]$

D ADDITION EXPERIMENTAL DETAILS FOR LARGE SCALE EXPERIMENTS

We provide the hyperparameters used for training in Table 5. Note that we ablate over the p parameter that decides the strength of the adaptive normalization. We either use the MeanFlow or the Lagrangian loss with the OT-CFM teacher (Tong et al., 2024) for CIFAR-10, or the EDM teacher (Karras et al., 2022) for FFHQ 64×64. Times are measured for the MeanFlow loss. Each entry of Tables 1 and 4, displays the mean and the standard deviation over 3 training runs.

We include the performance of Lagrangian-trained flow maps on CIFAR-10 in Table 4. Similarly to the MeanFlow trained flow maps, $p \geq 0.5$ produced the lowest FIDs, and $p = 0.0$ gave the worst performance.

Table 4: FID on CIFAR-10 for the Lagrangian objective with different normalization exponents p. Results are reported as mean $\pm$ standard deviation over three runs.

	FID ↓		
p	NFE 1	NFE 2	NFE 4
0.00	74.01 ± 8.71	25.82 ± 3.51	9.75 ± 1.16
0.25	56.61 ± 1.46	21.51 ± 0.85	8.14 ± 0.71
0.50	51.99 ± 8.56	19.24 ± 1.30	6.80 ± 0.15
0.75	42.54 ± 1.87	15.25 ± 1.06	$\mathbf{5.94 \pm 0.24}$
1.00	$\mathbf{39.58 \pm 5.96}$	$\mathbf{13.94 \pm 2.17}$	6.35 ± 0.67
1.25	54.33 ± 7.59	21.00 ± 3.00	8.06 ± 1.17
1.50	45.85 ± 2.94	20.32 ± 1.93	8.47 ± 1.05

Table 5: Training configuration for the CIFAR-10 and FFHQ experiments.

	CIFAR distilled	**FFHQ distilled**
Teacher	CFM (EMA)	EDM VE (EMA)
Architecture	EDM2-style U-Net	EDM2-style U-Net
Parameters	54.7M	59.9M
Model Channels	128	128
Channel Mults	[2, 2, 2]	[1, 2, 2, 2]
Attention Dims	16, 8	16, 8
Training time (1×H100)	13.5h	28.3h
Seeds	0, 1, 2	0, 1, 2
Diagonal %	0.5	0.5
Optimizer	Adam	Adam
Learning rate	2×10^{-3}	1×10^{-3}
Batch size	256	256
Training steps	200k	200k
Warm-up	2.5k	2.5k
EMA	0.9999	0.9999
Precision	bf16 AMP	bf16 AMP