Parsimonious Flow Matching for Efficient Image Generation

Tianjiao Ding, Ziqing Xu, Benjamin D. Haeffele, Hongkang Li, and René Vidal

University of Pennsylvania
tjding@upenn.edu

Abstract. Flow matching (FM) models generate data by learning a velocity field that transforms samples from a simple latent distribution, typically an isotropic Gaussian, to the data distribution. However, the geometric mismatch between a unimodal, full-dimensional Gaussian and a multimodal, approximately low-dimensional data distribution leads to a complex velocity field that is costly to learn at training time and to integrate at inference time. In this paper, we propose Parsimonious Flow Matching (PFM), which adopts a mixture of Gaussians (MoG) as the latent distribution whose geometry better aligns with the data. We identify key design choices that enable efficient FM, including an optimal-transport data-latent coupling, MoG estimation via k-means, and eigenvalue regularization of the per-mode covariances. Theoretically, when both data and latent distributions are MoGs, we show that gradient descent for FM with affine velocity fields converges faster when the corresponding modes of the latent and data are more similar and the per-mode covariances are well-conditioned. Further, for data that follows a separated MoG, we prove that replacing the isotropic Gaussian latent with a MoG in FM training accelerates gradient descent convergence and lowers the initial loss. On CIFAR-10 and ImageNet (32×32), PFM achieves the same generation quality as the baseline in fewer training iterations, and produces higher-quality samples for the same number of function evaluations at inference time. Code is available: https://github.com/tianjiaoding/pfm.

Keywords: Flow Matching · Generative Models · Machine Learning

1 Introduction

Flow matching (FM) models [1,26,29] have demonstrated remarkable capabilities in producing realistic images and videos. To generate a data point, these models first sample from a simple distribution in the latent space, often an isotropic Gaussian, and then iteratively apply a learned velocity field to the sample. Despite the simplicity of this idea, training these models to produce high-quality outputs is computationally demanding, often requiring large-scale optimization over high-dimensional velocity fields [12,21]. At inference time, generating a single sample typically requires a large number of function evaluations (NFE) to numerically integrate the learned flow [12,29].

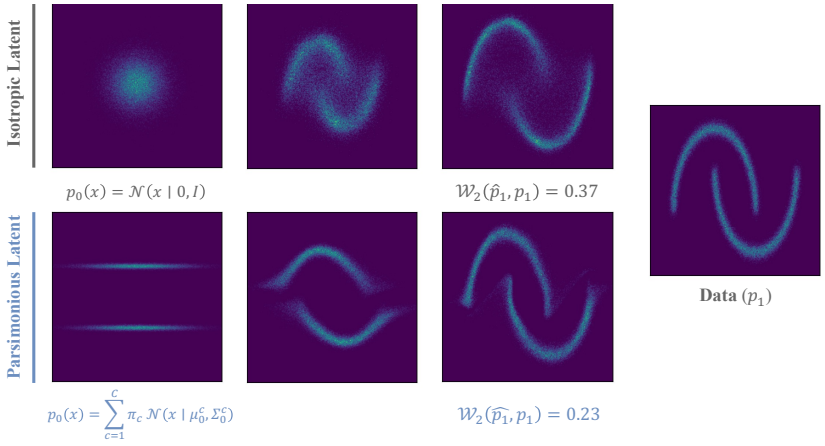


Fig. 1: Using an MoG as the latent distribution instead of an isotropic Gaussian allows for better matching the dimensionality and number of modes of the data distribution, which leads to more efficient training and inference of flow-based generative models. Both methods shown are trained with the same number of iterations.

A fruitful line of research has sought to address these challenges from several angles: reducing the complexity of the data space via autoencoders [7,8], promoting straighter flow trajectories [9, 24, 33, 40], optimizing time [20] and flow [36] discretizations, and designing expressive velocity fields [6, 15, 43]. An orthogonal yet much less explored direction is the design of the *latent distribution* itself, which is the focus of this paper.

One key observation motivating this direction is the geometric and statistical mismatch between an isotropic Gaussian latent distribution and the data distribution. The latter typically has *multiple modes*, each supported on a *low-dimensional* manifold (cf. the *manifold hypothesis* [4, 13, 34]), while an isotropic Gaussian is *unimodal* and *full-dimensional*. This mismatch has two consequences. First, the velocity field must learn to simultaneously create multimodal structure and reduce dimensionality, leading to a poorly conditioned optimization landscape that slows training convergence. Second, the resulting flow trajectories are more complex, requiring a larger NFE at inference time[1]. Motivated by these observations, we ask the following questions:

(Q1) Can we improve the training and generation efficiency of FM by designing a structured latent distribution?

(Q2) How does a structured latent provably improve the training efficiency of the FM optimization problem?

[1] The mode mismatch can even lead to the generated distribution having spurious support that interpolates the true modes [4].

The answer to Q1 appears elusive: Jia et al. [18] showed that adopting a mixture of Gaussians as the latent distribution can improve generation quality, albeit for *diffusion models* on a *subset* of CIFAR-10, while recently Lee et al. [23] offered empirical counter-evidence for FM models. As for Q2, to the best of our knowledge, no prior work has theoretically analyzed how the structure of the *latent distribution* affects the optimization landscape of FM.

Contributions. In this paper, we provide answers to the above questions by making the following contributions.

- *Algorithm* (§3): To better capture the multi-modality and approximate low-dimensionality of the data, we consider using a mixture of Gaussians (MoG) as the latent distribution for FM. In particular, we investigate design choices that enables efficient FM, including an optimal-transport data-latent coupling, MoG estimation via k-means, and eigenvalue regularization of the per-mode covariances. Drawing on the principle of parsimony that has long guided the pursuit of efficient latent representations in machine learning [3,5,31], we coin the resulting method Parsimonious Flow Matching (PFM).
- *Theory* (§4): We theoretically characterize the convergence rate of gradient descent (GD) for FM with affine velocity fields. When both the data and latent distributions are MoGs, we prove that GD converges faster when i) the corresponding modes of the latent and data are similar and ii) the per-mode covariances are well regularized (Theorem 1). When the data distribution is a MoG with two classes, we further show that replacing the isotropic Gaussian latent with an MoG accelerates GD convergence and reduces the initial loss in FM training (Theorem 2).
- *Experiments* (§5): We demonstrate the efficacy of the proposed method on benchmarks including CIFAR and ImageNet (32×32). Under comparable training settings, the proposed method takes $25\% - 50\%$ fewer training iterations to achieve the same generation quality. It further achieves higher generation quality using the same NFE at inference time.

2 Preliminary

In this section, we briefly review the preliminaries of FM that will set the stage for describing our method (§3) and analyzing its theoretical properties (§4).

2.1 Basics of Flow Matching (FM)

The central object in FM is a time-dependent velocity field $u_t : \mathbb{R}^d \to \mathbb{R}^d$ for $t \in [0, 1]$, which defines an ordinary differential equation (ODE) for the flow $\phi_t : \mathbb{R}^d \to \mathbb{R}^d$:

$$\frac{d}{dt}\phi_t(\boldsymbol{x}) = u_t(\phi_t(\boldsymbol{x})), \quad \phi_0(\boldsymbol{x}) = \boldsymbol{x}. \tag{1}$$

The flow pushes an initial distribution p_0 forward so that the marginal at time t is $p_t = (\phi_t)_{\#}p_0$. The goal of FM is to design u_t such that $p_0 \approx q_0$ and $p_1 \approx q_1$,

where q_1 is a latent distribution from which we can easily sample and q_0 is the data distribution.

Conditional FM (CFM) [26, 40] provides a tractable construction. Given a coupling distribution $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ with marginals q_0 and q_1, CFM defines the conditional probability path

$$p_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) = \mathcal{N}\big(\boldsymbol{x} \mid t\boldsymbol{x}_1 + (1-t)\boldsymbol{x}_0,\, \sigma^2 \boldsymbol{I}\big) \tag{2}$$

and the conditional velocity $u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) = \boldsymbol{x}_1 - \boldsymbol{x}_0$. The marginal path $p_t(\boldsymbol{x}) = \int p_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1)\, q(\boldsymbol{x}_0, \boldsymbol{x}_1)\, d\boldsymbol{x}_0\, d\boldsymbol{x}_1$ satisfies $p_0 \approx q_0$ and $p_1 \approx q_1$ as $\sigma \to 0$. A neural network v_θ is trained to approximate the marginal velocity by minimizing the CFM loss:

$$\mathcal{L}^{\mathrm{CFM}}(\theta) = \mathbb{E}_{t,\, q(\boldsymbol{x}_0, \boldsymbol{x}_1),\, p_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1)} \left\| v_\theta(\boldsymbol{x}, t) - u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) \right\|^2 . \tag{3}$$

Crucially, the coupling $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ is the central design choice in CFM: it determines the probability path, the complexity of the velocity field, and hence both training and inference efficiency.

2.2 Mini-batch Optimal Transport in FM

Under independent coupling $q(\boldsymbol{x}_0, \boldsymbol{x}_1) = q_0(\boldsymbol{x}_0)\, q_1(\boldsymbol{x}_1)$, a latent point $\boldsymbol{x}_0$ may be paired with a distant data point $\boldsymbol{x}_1$, leading to crossing flow trajectories and thus a complex velocity field (see for example the work of [29] and Figure 2 therein). To promote straighter trajectories, which simplifies the velocity field and facilitates ODE sampling, one can instead pursue an optimal transport (OT) coupling between the data and latent distribution [29, 33, 40]. Let $\Gamma(q_0, q_1)$ be the set of couplings between q_0 and q_1, i.e., the set of joint distributions whose marginals are q_0 and q_1. For any coupling q denote the transport cost by $\mathcal{J}(q) := \mathbb{E}_{(\boldsymbol{x}_0, \boldsymbol{x}_1) \sim q}\big[\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|^2\big]$. An OT coupling is one that minimizes the transport cost; the OT cost, denoted by $\mathcal{W}_2^2(q_0, q_1)$, is known as the squared Wasserstein-2 distance between q_0 and q_1.

Computing the exact OT coupling is intractable in high dimensions. The mini-batch approximation [33, 40] addresses this as follows: at each training step, one draws a mini-batch of B samples from q_0 and q_1 independently, solves a discrete OT problem on the batch, which is a linear assignment problem computable in $\mathcal{O}(B^3)$ time, and uses the resulting pairs to evaluate the CFM loss (3) and train the velocity field. This adds negligible overhead to training and requires no modification at inference time.

3 Parsimonious Flow Matching (PFM)

As motivated in §1, data distributions typically exhibit multiple modes (e.g., a visual dataset may have multiple semantic classes) and approximate low-dimensionality. To better capture these properties in the latent distribution of FM and thus facilitate learning the velocity field, we consider using an MoG

as the latent distribution. Let $z \in \{1, \dots, C\}$ be a class variable with prior $\pi_c := p(z = c)$. We adopt a latent distribution of the form

$$q_0(\boldsymbol{x}) = \sum_{c=1}^{C} \pi_c \, q(\boldsymbol{x}_0 \mid z = c) = \sum_{c=1}^{C} \pi_c \, \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_0^c, \boldsymbol{\Sigma}_0^c), \tag{4}$$

where $\boldsymbol{\mu}_0^c \in \mathbb{R}^d$ and $\boldsymbol{\Sigma}_0^c \in \mathbb{R}^{d \times d}$ are the mean and covariance of each Gaussian mode. Given this structured latent, three design choices arise: how to couple data and latent (§3.1), how to obtain the MoG parameters from data, and how to regularize the covariance spectrum (§3.2).

3.1 Coupling between the Data and the MoG Latent

A common choice of coupling between the data and latent distributions is independent coupling. However, given the pursuit of a simpler velocity, we instead adopt OT couplings. The reason is illustrated as follows. Suppose the latent perfectly matches the data, i.e., $q_0 = q_1$. Under independent coupling, the velocity field minimizing the CFM loss (3) can still be complicated; see [9, equation 4] for an example. Luckily, under OT coupling between the data and latent, the optimal velocity as per (3) is the zero field, arguably the simplest velocity to learn. This is a core intuition that we will explore in the analysis (§4).

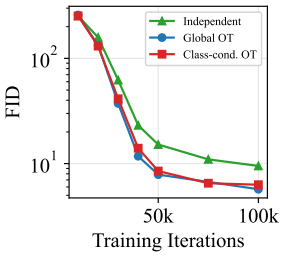
That being said, since the MoG latent (4) carries class structure, a natural question is: at what granularity should we apply the OT coupling (§2.2)?

1. *Global OT*: $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ is the OT coupling of the marginals q_0 and q_1.
2. *Class-conditional OT*: Since the latent MoG is learned from the data, let the class variable z index the modes of both the data and the latent. Then, $q(\boldsymbol{x}_0, \boldsymbol{x}_1) = \sum_{c=1}^{C} \pi_c \, q(\boldsymbol{x}_0, \boldsymbol{x}_1 \mid z = c)$, where $q(\boldsymbol{x}_0, \boldsymbol{x}_1 \mid z = c)$ is the OT coupling of the class-conditionals $q(\boldsymbol{x}_0 \mid z = c)$ and $q(\boldsymbol{x}_1 \mid z = c)$.

In our experiments, the two options often yield comparable generation quality; see Table 1 (a). Class-conditional OT incurs overhead similar to global OT, as it simply groups each mini-batch by class and solves a smaller assignment problem within each group. Our theory (§4) adopts class-conditional OT, as it admits closed-form expressions for the covariances along the interpolation path that facilitate the analysis.

3.2 Obtaining the MoG Parameters

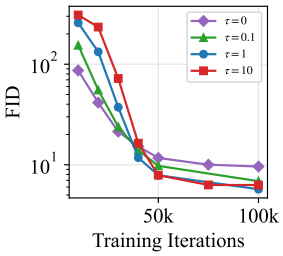
Intuitively, the closer the latent distribution matches the mode structure of the data, the less work the velocity field must do to transform one into the other. This suggests choosing latent distributions whose means align with the data cluster centers, whose covariances span similar subspaces, and whose mixing weights reflect the data class proportions. A natural way to achieve this is to learn the parameters directly from the data, which we describe next.



(a) Data-latent Coupling

MoG	FID
random	5.85
supervised	6.3
10-means	**5.71**

(b) Choice of MoGs



(c) Cov. Regularization

Table 1: Ablation study on the design choices of latent MoG in §3. All models are trained with $100k$ iterations on CIFAR-10, and FIDs are evaluated with NFE=12.

Learning the MoG from Data. The classical approach for fitting a MoG is Expectation-Maximization (EM). However, its per-iteration cost is cubic in the dimension d due to covariance inversions, and existing implementations do not exploit GPUs. Since visual data typically have large d (e.g., CIFAR images have $d = 32 \times 32 \times 3 = 3072$), one either runs a few iterations of EM, which gives poor estimates, or runs many iterations for a long wall-clock time, which defeats the goal of efficiency. Instead, we use k-means, which requires no covariance inversion and admits efficient GPU implementations. For each output cluster c, we estimate π_c as the proportion of data in it, $\boldsymbol{\mu}_1^c$ as its centroid, and $\boldsymbol{\Sigma}_1^c$ as its sample covariance. k-means is run efficiently to produce the MoG parameters, which are then fixed throughout FM training. Implementation details and an empirical comparison of EM and k-means are in §C.2.

Regularizing the MoG Parameters. We observe that the learned per-mode covariances $\boldsymbol{\Sigma}_1^c$ have a rapidly decaying eigenvalue spectrum. Near-degenerate covariances cause two problems: (i) the ground-truth velocity field u_t becomes ill-defined in directions where the covariance vanishes [38]; and (ii) the velocity model receives insufficient training signal in low-density directions, leading to poor generalization [23]. A natural approach to regularize the learnt MoG is probabilistic PCA [39], which models the covariance as a low-rank matrix plus an isotropic residual term. The optimal magnitude of the residual term is determined by the mean of the smallest eigenvalues not captured by the low-rank component. However, we observe that the top few eigenvalues account for nearly all the variance, while the remaining eigenvalues are smaller than 10^{-3}. This results in the residual term too small to meaningfully regularize the degenerate directions. Instead, we regularize the covariances by applying the map $\sigma \mapsto \max(\sigma, \tau)$ element-wise to the eigenvalues of the covariances, where $\tau > 0$ is a hyperparameter. As seen in Table 1 (c), without a small τ for regularization, FID is low in early training but then plateaus at a suboptimal value. A moderate τ yields the best balance between fast early improvement and continued convergence to low FID.

Connection with [23]. A recent work [23] shows that using an MoG as the latent distribution yields much worse generation quality than the baseline. We make the following observations. First, EM was used to estimate the MoG parameters, which tends to yield poor estimates with a compute budget as explained above (see also §C.2). Second, the learned MoGs were not regularized, while we show that properly regularizing the per-mode covariances of MoGs improves the generation quality empirically (Table 1) and theoretically (Theorem 1).

4 Theoretical Analysis of FM Training

This section aims to provide a theoretical justification of the algorithmic design of PFM. We first introduce notations and assumptions that enable the analysis (§4.1), and provide the convergence rate of GD in terms of statistical quantities of the data-latent coupling (§4.2). When the data and latent distributions are MoGs coupled by class-conditional OT, we prove that GD converges faster when i) the mode-conditionals of latent and data are similar and ii) the per-mode covariances are well regularized (§4.3). When the data distribution is a MoG with two classes, we further show that class-conditional OT can approximately achieve global OT, and that an MoG latent distribution achieves faster convergence and smaller initial loss than an isotropic Gaussian (§4.4).

4.1 Problem Setup

With the goal of deriving the rate of convergence of gradient descent on the FM objective (3) from §2.1, in this section we setup basic notations and assumptions.

We first discuss the three main components of (3), which are the conditional probability path, the conditional velocity, and the learnable velocity field. Recall that $\boldsymbol{x}_0, \boldsymbol{x}_1$ are random vectors with the law of $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$; we will specify the law later in §4.3 and §4.4. For ease of exposition, we consider a simplified version of the conditional probability path (2) as follows

$$\boldsymbol{x}_t := (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1, \tag{5}$$

which corresponds to (2) in the limit of $\sigma \to 0$. Denote $\boldsymbol{y}$ as a shorthand for the conditional velocity field $u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) = \boldsymbol{x}_1 - \boldsymbol{x}_0$. In practice, the learnable velocity field $v_\theta(\boldsymbol{x}, t)$ in (3) are typically parameterized by deep networks with advanced architectures (e.g., U-Net and DiTs). However, deep networks are highly non-convex and non-smooth, which makes the theoretical analysis challenging. To facilitate analysis, we adopt an affine[2] velocity $v_\theta(\boldsymbol{x}, t) = \boldsymbol{A}_t\boldsymbol{x}$ for every $t \in [0, 1]$. While this is a gross simplification, we will be compensated by the ability to develop theory that predicts and explains the empirical effectiveness of algorithmic design.

[2] For ease of exposition, §4 restricts attention to affine velocity fields without a bias term and assume $\mathbb{E}[\boldsymbol{x}_0] = \mathbb{E}[\boldsymbol{x}_1] = 0$. The analysis can be extended to the non-centered case in a straightforward manner, which we detail in Appendix B.

Using the above assumptions and notations in the FM objective (3) leads to

$$\mathcal{L}(\{\boldsymbol{A}_t\}_{t\in[0,1]}) = \mathbb{E}_{t\sim\mathrm{Unif}[0,1],q(\boldsymbol{x}_0,\boldsymbol{x}_1)}\left[\left\|\boldsymbol{A}_t\boldsymbol{x}_t-\boldsymbol{y}\right\|_2^2\right]. \tag{6}$$

Optimizing (6) is equivalent to optimizing for every $t\in[0,1]$

$$\mathcal{L}_t(\boldsymbol{A}_t) = \mathbb{E}_{q(\boldsymbol{x}_0,\boldsymbol{x}_1)}\left[\left\|\boldsymbol{A}_t\boldsymbol{x}_t-\boldsymbol{y}\right\|_2^2\right]. \tag{7}$$

Thus, we consider gradient descent (GD) of (7) with a constant step size $\eta_t>0$:

$$\boldsymbol{A}_t^{(k+1)} = \boldsymbol{A}_t^{(k)} - \eta_t\nabla\mathcal{L}_t(\boldsymbol{A}_t^{(k)}). \tag{8}$$

Notations. Below, expectations and covariances are with respect to the law of $q(\boldsymbol{x}_0,\boldsymbol{x}_1)$. For $t\in[0,1]$, denote the mean and covariance of $\boldsymbol{x}_t$ by $\boldsymbol{\mu}_t=\mathbb{E}[\boldsymbol{x}_t]$ and $\boldsymbol{\Sigma}_t=\mathrm{Cov}(\boldsymbol{x}_t)$. Likewise, let $\boldsymbol{\Sigma}_{yy}=\mathrm{Cov}(\boldsymbol{y})$, $\boldsymbol{\Sigma}_{yt}=\mathrm{Cov}(\boldsymbol{y},\boldsymbol{x}_t)$, and $\boldsymbol{\Sigma}_{01}=\mathrm{Cov}(\boldsymbol{x}_0,\boldsymbol{x}_1)$. For any square matrix $\boldsymbol{A}$, denote its largest and smallest eigenvalues by $\lambda_{\max}(\boldsymbol{A})$ and $\lambda_{\min}(\boldsymbol{A})$, and its condition number by $\kappa(\boldsymbol{A})=\frac{\lambda_{\max}(\boldsymbol{A})}{\lambda_{\min}(\boldsymbol{A})}$.

4.2 Basic Convergence Rates of FM Training

Thanks to the simplifications above, FM training (7) is now linear regression with well-known optimization behavior. Indeed, its minimizer is $\boldsymbol{A}_t^*=\boldsymbol{\Sigma}_{yt}\boldsymbol{\Sigma}_t^{-1}$, and GD (8) with the optimal constant step size $\eta_t^*=\frac{1}{\lambda_{\min}(\boldsymbol{\Sigma}_t)+\lambda_{\max}(\boldsymbol{\Sigma}_t)}$ has the following convergence rates (c.f. Lemma 3 and its proof)

$$\|\boldsymbol{A}_t^{(k)}-\boldsymbol{A}_t^*\|_F \le \rho_t^k\|\boldsymbol{A}_t^{(0)}-\boldsymbol{A}_t^*\|_F, \tag{9}$$

$$\mathcal{L}_t(\boldsymbol{A}_t^{(k)})-\mathcal{L}_t^* \le \rho_t^{2k}\big(\mathcal{L}_t(\boldsymbol{A}_t^{(0)})-\mathcal{L}_t^*\big), \tag{10}$$

where $\rho_t:=\frac{\kappa(\boldsymbol{\Sigma}_t)-1}{\kappa(\boldsymbol{\Sigma}_t)+1}$ is the contraction factor and $\mathcal{L}_t^*=\mathcal{L}_t(\boldsymbol{A}_t^*)$ is the minimum objective. In particular, (9) and (10) reveal that GD convergence is governed by two factors: (i) the condition number $\kappa(\boldsymbol{\Sigma}_t)$, which controls ρ_t, and (ii) the initialization errors $\|\boldsymbol{A}_t^{(0)}-\boldsymbol{A}_t^*\|_F$ and $\mathcal{L}_t(\boldsymbol{A}_t^{(0)})-\mathcal{L}_t^*$.

However, the convergence rates above depend on $\boldsymbol{\Sigma}_t$ and $\boldsymbol{\Sigma}_{yt}$, and one may wonder how they connect to the properties of the latent and data distributions.

Lemma 1. *Using the conditional probability path (5), we have that*

$$\begin{aligned}\forall t\in[0,1],\quad \boldsymbol{\Sigma}_t &= (1-t)^2\boldsymbol{\Sigma}_0 + t^2\boldsymbol{\Sigma}_1 + t(1-t)\big(\boldsymbol{\Sigma}_{01}+\boldsymbol{\Sigma}_{01}^\top\big),\\ \boldsymbol{\Sigma}_{yt} &= (1-t)(\boldsymbol{\Sigma}_{01}^\top-\boldsymbol{\Sigma}_0)+t(\boldsymbol{\Sigma}_1-\boldsymbol{\Sigma}_{01}).\end{aligned} \tag{11}$$

Lemma 1 says that $\boldsymbol{\Sigma}_t$ and $\boldsymbol{\Sigma}_{yt}$ depend on the covariances $\boldsymbol{\Sigma}_0,\boldsymbol{\Sigma}_1$ of the latent and data distributions, as well as their cross-covariance $\boldsymbol{\Sigma}_{01}$. That being said, several issues remain. Recall that in PFM (§3), the latent $\boldsymbol{x}_0$ follows a MoG distribution (that is learned from $\boldsymbol{x}_1$), and it is further coupled with the data $\boldsymbol{x}_1$ via OT. First, $\boldsymbol{\Sigma}_0,\boldsymbol{\Sigma}_1,\boldsymbol{\Sigma}_{01}$ are aggregated objects that hide the covariance of each mode as well as the similarity of corresponding modes of latent and data. Second, $\boldsymbol{\Sigma}_{01}$ depends on the OT coupling between $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$, which is difficult to characterize when $\boldsymbol{x}_1$ is a general distribution.

4.3 MoG as Data and Latent Distributions

To address the issues above, we consider a model of $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ where the latent $\boldsymbol{x}_0$ and data $\boldsymbol{x}_1$ follow MoGs coupled with class-conditional OT: i.e., conditioned on each mode, the latent and data are individually Gaussian, and they are coupled by OT. Since the OT map between two Gaussians is affine, such a model allows us to characterize how the per-mode parameters affect the $\boldsymbol{\Sigma}_0, \boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_{01}$ and thus convergence rates. In particular, we show that the convergence rate is faster when i) the corresponding modes of the latent and data are similar and ii) the per-mode covariances are well regularized.

We begin by specifying $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$. Let $z \in \{1, \dots, C\}$ be a categorical variable indexing the modes, with weights $\{\pi_c\}_{c=1}^C$ satisfying $\pi_c \geq 0$ and $\sum_c \pi_c = 1$. Consider Gaussian class-conditionals

$$\boldsymbol{x}_0 \mid z = c \sim \mathcal{N}(\boldsymbol{\mu}_0^c, \boldsymbol{\Sigma}_0^c), \qquad \boldsymbol{x}_1 \mid z = c \sim \mathcal{N}(\boldsymbol{\mu}_1^c, \boldsymbol{\Sigma}_1^c),$$

with corresponding modes related by a single positive-definite matrix $\boldsymbol{B}_c \succ 0$,

$$\boldsymbol{\mu}_1^c = \boldsymbol{B}_c \boldsymbol{\mu}_0^c, \qquad \boldsymbol{\Sigma}_1^c = \boldsymbol{B}_c \boldsymbol{\Sigma}_0^c \boldsymbol{B}_c^\top. \tag{12}$$

Conditioned on $z = c$, we couple $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ by OT. As both class-conditionals are Gaussian, the OT map sending $\boldsymbol{x}_0$ to $\boldsymbol{x}_1$ is affine, which under (12) reduces to the linear map $\boldsymbol{x}_1 = \boldsymbol{B}_c \boldsymbol{x}_0$.

Before moving on to the theorem, we define a few useful quantities. Let $\Delta_c := \boldsymbol{B}_c - \boldsymbol{I}$ measure how $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ differ in their c-th modes. Next, we characterize how the mode means spread out. Define the mean offsets $\delta_j^c := \boldsymbol{\mu}_j^c - \boldsymbol{\mu}_j$ for $j \in \{0, 1\}$ and, under the linear path (5), their interpolated versions $\delta_t^c := (1-t)\delta_0^c + t\delta_1^c$ for $t \in (0, 1)$. To summarize these offsets in a single quantity, define the Gram matrix $G_t \in \mathbb{R}^{C \times C}$ with $(G_t)_{i,j} = \sqrt{\pi_i \pi_j}\, \langle \delta_t^i, \delta_t^j \rangle$. Its norm $\|G_t\|$ equals the largest eigenvalue of the between-class scatter $\sum_c \pi_c \delta_t^c (\delta_t^c)^\top$, i.e., the maximal variance of the mode means about the global mean along any single direction. Hence $\|G_t\|$ is large when i) the mode means lie far from the global mean and ii) the offsets align along a common direction.

Theorem 1 (Convergence Rate for MoG Distributions). *Under the model above, for any $t \in [0, 1]$, we have*

$$\boldsymbol{\Sigma}_t = \sum_{c=1}^C \pi_c \Big((\boldsymbol{I} + t\Delta_c) \boldsymbol{\Sigma}_0^c (\boldsymbol{I} + t\Delta_c)^\top + \delta_t^c (\delta_t^c)^\top \Big), \tag{13}$$

$$\boldsymbol{\Sigma}_{yt} = \sum_{c=1}^C \pi_c \Big(\Delta_c \boldsymbol{\Sigma}_0^c (\boldsymbol{I} + t\Delta_c)^\top + (\delta_1^c - \delta_0^c)(\delta_t^c)^\top \Big). \tag{14}$$

Consider GD on $\mathcal{L}_t(\boldsymbol{A}_t)$ in (7). Assume $\|\Delta_c\| < 1, \forall c \in [C]$, and define

$$\kappa_t^{\mathrm{up}} = \left(\sum_{c=1}^C \pi_c \|\Sigma_0^c\| (1 + t\|\Delta_c\|)^2 + \|G_t\| \right) \left(\sum_{c=1}^C \frac{\pi_c}{\lambda_{\min}(\Sigma_0^c)(1 - t\|\Delta_c\|)^2} \right). \tag{15}$$

The convergence rates in (9) and (10) hold with contraction factor $\rho_t := \frac{\kappa_t^{\mathrm{up}} - 1}{\kappa_t^{\mathrm{up}} + 1}$.

We refer the readers to Appendix B.3 for the proof. Theorem 1 characterizes the covariance matrices for any $t \in [0,1]$ and the corresponding convergence rates. The upper condition number κ_t^{up} admits a natural decomposition into *within-class* and *between-class* contributions. The first factor in κ_t^{up} upper-bounds $\lambda_{\max}(\boldsymbol{\Sigma}_t)$ and consists of a within-class term $\sum_c \pi_c \|\boldsymbol{\Sigma}_0^c\|(1+t\|\Delta_c\|)^2$, which captures the spread of each mode covariance under the interpolation, and a between-class term $\|G_t\|$, which measures the effective separation among the interpolated mode means $\{\delta_t^c\}$. Two qualitative consequences follow.

Smaller covariance mismatch Δ_c leads to faster convergence. As $\|\Delta_c\|$ increases, the factor $(1+t\|\Delta_c\|)^2$ inflates the numerator of κ_t^{up} while $(1-t\|\Delta_c\|)^2$ shrinks its denominator, slowing the convergence. In the limit of $\Delta_c = \mathbf{0}$ for all c, two interesting effects happen. First, the contraction factor ρ_t is governed only by per-mode covaraince and the spread of the mode means alone, since $\boldsymbol{\Sigma}_t$ reduces to $\sum_c \pi_c \boldsymbol{\Sigma}_0^c + \sum_c \pi_c \delta_t^c (\delta_t^c)^\top$. Second, the initialization error vanishes: note that $\delta_0^c = \delta_1^c$ for all $c \in [C]$, thus $\boldsymbol{\Sigma}_{yt} = \mathbf{0}$. Notably, $\boldsymbol{A}_t^* = \mathbf{0}$, and a zero initialization of $\boldsymbol{A}_t^{(0)} = \mathbf{0}$ is already optimal, requiring no GD steps. However, in practice the initialization error is unlikely to be zero, so the contraction factor ρ_t always affects the convergence. Importantly, this means that driving the covariance mismatch $\|\Delta_c\|$ towards zero is not always desirable, because the initialization error and ρ_t are coupled by the shared choice of latent covariance. Namely, pushing $\boldsymbol{\Sigma}_0^c$ close to $\boldsymbol{\Sigma}_1^c$ reduces initialization error but inherits the smallest eigenvalue from data, which can be vanishingly small and blow up ρ_t. This reveals a trade-off between minimizing the mismatch $\|\Delta_c\|$ and regularizing the modes, which justifies the eigenvalue thresholding in our algorithm (§3.2).

Larger between-class separation $\|G_t\|$ leads to slower convergence. In the literature of *subspace clustering*, an important data model is one where the data is from a mixture of zero-mean low-dimensional distributions [11, 25, 41]. In this case, all modes have zero mean, thus $G_t = \mathbf{0}$ and the rate depends only on the within-class covariance structure. On the other hand, when the modes lie far from the global mean and their offsets are aligned, $\|G_t\|$ becomes large, and so does κ_t^{up}, which may slow convergence.

4.4 Comparison between Different Latent Distributions

So far, we have established the convergence rate of GD for FM with affine velocity fields under general data assumptions (Lemma 1) and when both the data and latent distributions follow an MoG (Theorem 1). In practice, however, the default setting for the latent distribution in FM is the isotropic Gaussian distribution. Thus, it remains unclear whether, for a fixed data distribution, choosing an MoG as the latent distribution yields any convergence gain over an isotropic Gaussian.

In this section, we characterize and compare the convergence rate and training loss of FM when the data distribution is an MoG and the latent distribution is either an MoG or an isotropic Gaussian. We refer to these two scenarios as MoG→MoG and MoG→SG, respectively. For simplicity, we restrict the analysis

to symmetric two-mode mixture of Gaussian distributions. Mathematically,

$$\begin{aligned} q_1: \quad & \boldsymbol{x}_1 \sim \frac{1}{2}\mathcal{N}(-\boldsymbol{\mu}, \rho^2\boldsymbol{I}) + \frac{1}{2}\mathcal{N}(\boldsymbol{\mu}, \rho^2\boldsymbol{I}), \\ q_0: \quad & \boldsymbol{x}_0 \sim \begin{cases} \frac{1}{2}\mathcal{N}(-\boldsymbol{\mu}', \rho'^2\boldsymbol{I}) + \frac{1}{2}\mathcal{N}(\boldsymbol{\mu}', \rho'^2\boldsymbol{I}), & \text{MoG} \to \text{MoG}, \\ \mathcal{N}(0, \boldsymbol{I}), & \text{MoG} \to \text{SG}. \end{cases} \end{aligned} \tag{16}$$

We next ask whether a simple coupling between $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ can approximately achieve OT in both the MoG→MoG and MoG→SG scenarios. To this end, we introduce the following *mode-wise affine coupling* $q_{\text{mwac}} \in \Gamma(q_0, q_1)$.

Definition 1 (Mode-wise affine coupling). *For any $(\boldsymbol{x}_0, \boldsymbol{x}_1) \sim q_{mwac}$, given the latent variable $S \sim \text{Unif}\{-1, 1\}$, we have*

$$\boldsymbol{x}_1 \mid (S = s) \sim \mathcal{N}(s \cdot \boldsymbol{\mu}, \rho^2 \cdot \boldsymbol{I}), \tag{17}$$

$$\boldsymbol{x}_0 \mid (\boldsymbol{x}_1, S = s) = \begin{cases} (\boldsymbol{x}_1 - s \cdot \boldsymbol{\mu})\rho'/\rho + s \cdot \boldsymbol{\mu}', & \text{MoG} \to \text{MoG}, \\ (\boldsymbol{x}_1 - s \cdot \boldsymbol{\mu})/\rho, & \text{MoG} \to \text{SG}. \end{cases} \tag{18}$$

$$\boldsymbol{\Sigma}_1^c = \rho'^2\boldsymbol{I}, \quad \boldsymbol{\Sigma}_0^c = \rho'^2\boldsymbol{I} \tag{19}$$

By construction, the marginal distribution of $\boldsymbol{x}_1$ is a symmetric two-mode MoG with means $\pm\boldsymbol{\mu}$ and per-mode covariance $\rho^2\boldsymbol{I}$, while the marginal of $\boldsymbol{x}_0$ is an MoG with parameters $\boldsymbol{\mu}', \rho'$ in the MoG→MoG scenario and the isotropic Gaussian in MoG→SG. A natural question is whether this simple coupling is near-optimal in the transport sense. The following lemma characterize the conditions under which the mode-wise affine coupling defined above can approximately lead to OT in both scenarios.

Lemma 2. *$\forall \epsilon > 0$, if (i) $\boldsymbol{\mu}' = r\boldsymbol{\mu}$, $0 \le r \le R < 1$, (ii) $\|\boldsymbol{\mu}\| \gtrsim \epsilon^{-1}$, and (iii) $0 < \rho, \rho' \le C$ for some $C > 1$, then the mode-wise affine coupling in Definition 1 satisfies that for both* MoG → MoG *and* MoG → SG *from $\boldsymbol{x}_1$ to $\boldsymbol{x}_0$,*

$$\mathcal{J}(q_{\text{mwac}}) \le (1 + O(\epsilon))\mathcal{W}_2^2(q_0, q_1). \tag{20}$$

Lemma 2 shows that when the latent mean is large enough with the covariance upper bounded by a constant, and the mean of the data and latent distributions are collinear (See conditions in Lemma 2), the mode-wise affine coupling (18) from $\boldsymbol{x}_1$ to $\boldsymbol{x}_0$ achieves an ϵ-OT. This means that the transport cost $\mathcal{J}$ (see definition in §2.2) of the coupling exceeds the squared Wasserstein-2 distance by at most an $O(\epsilon)$ factor. The intuition behind this result is that if the mass of the MoG is concentrated around cluster centers that are far apart from each other, then the OT can be approximately realized by a uniform translation and rescaling between the corresponding centers of $\boldsymbol{x}_1$ and $\boldsymbol{x}_0$. The proof of Lemma 2 is in Appendix B.4.

Next, under the setting where a linear coupling can achieve approximate OT, we analyze what distribution parameters can make MoG→MoG yield better

training performance than MoG→SG. Let $\bar{\kappa} := \kappa(\mathbb{E}_{t\sim\mathrm{Unif}[0,1]}\boldsymbol{\Sigma}_t)$. If we consider a time-shared model $\bar{\boldsymbol{A}} = \boldsymbol{A}_t$ for any $t \sim \mathrm{Unif}[0,1]$, then $\bar{\kappa}$ is exactly the condition number of $\bar{\boldsymbol{A}}$ by (6), which has practical significance because real-world velocity networks share the same parameters across different time steps rather than specifying a separate model for each time step. Denote $\bar{\mathcal{L}}^{(0)}$ as the expected loss over $t \sim \mathrm{Unif}[0,1]$ under zero initialization, which equals $\mathbb{E}[\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|^2]$, i.e., the expected squared magnitude of the conditional velocity induced by the coupling. Thus, a smaller initialization loss indicates a shorter and simpler transport path that the velocity model needs to learn. Based on the above notations, we state the following theorem.

Theorem 2. *For any $\epsilon > 0$, if (i) $0 \le r \le \min\{\frac{1}{2}, \frac{4\rho(\rho-1)}{\rho^2+\rho+1}\}$, (ii) $\rho > 1$, (iii) $\rho' \in (\rho, 2\rho-1)$, and conditions in Lemma 2 hold, we have that the mode-wise affine mapping (18) satisfies*

$$\mathcal{J}(q_{\mathrm{mwac}}) \le (1 + O(\epsilon))\mathcal{W}_2^2(q_0, q_1) \quad \textit{and} \quad \begin{cases} \bar{\kappa}_{(MoG\to MoG)} \le \bar{\kappa}_{(MoG\to SG)}, \\ \bar{\mathcal{L}}^{(0)}_{(MoG\to MoG)} \le \bar{\mathcal{L}}^{(0)}_{(MoG\to SG)}, \end{cases} \tag{21}$$

The complete proof of Theorem 2 can be found in Appendix B.5. The main idea is to compute each $\bar{\kappa}$ and $\bar{\mathcal{L}}^{(0)}$, and then solve (21). To provide a clearer understanding of Theorem 2, we present an example of the parameter setting.

Example. The following parameters satisfy conditions (i)-(iii) in Theorem 2: $r = 0.5$, $\rho = 1.4$, $\rho' = 1.47$, $d = 10$, $\boldsymbol{\mu} = 0.2 \cdot \mathbf{1}_d$, and $\boldsymbol{\mu}' = 0.1 \cdot \mathbf{1}_d$. We provide experimental validation of this example in Appendix C.

Remark 1. To simplify the analysis, under the assumption that the means of the data and latent MoG distributions are collinear, Theorem 2 provides parameter conditions under which MoG→MoG achieves a smaller initial loss and faster convergence (i.e., a smaller condition number) than MoG→SG. In particular, conditions (i)-(iii) show that when considering the same model parameter shared across different time steps, if the relative distances between the cluster centers of the input data are large, then using an MoG as the latent distribution obtained via mode-wise affine coupling, where the means are contracted toward the center, and the covariances only change slightly, yields a better-conditioned objective with smaller initial transport complexity than using an isotropic Gaussian as the latent distribution. The intuition is that transporting from $\boldsymbol{\mu}$ to $\boldsymbol{\mu}' = r\boldsymbol{\mu}$ requires a shorter distance than transporting from $\boldsymbol{\mu}$ to 0, which leads MoG→MoG to achieve better performance than MoG→SG.

5 Experiments

5.1 Synthetic Datasets

We first validate PFM on a 2D checkerboard dataset, which exhibits clear multimodal structure. The following latent distributions are compared: an isotropic Gaussian, learned MoGs with $C \in \{2, 4\}$ modes; the latent and data distributions

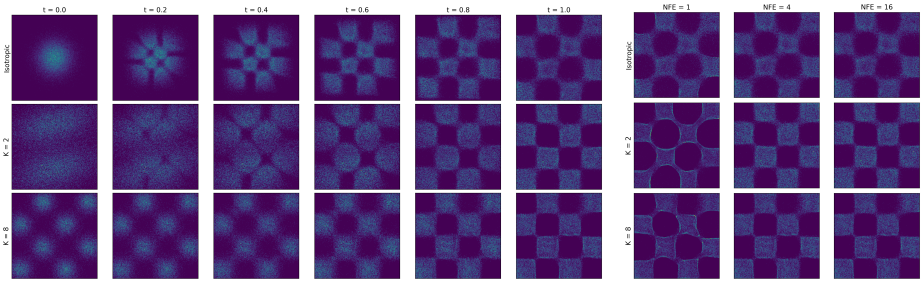


Fig. 2: FM on the checkboard dataset using isotropic Gaussian (top) and learned MoG with $C = 2$ (middle) and 4 (bottom) modes as latent distributions.

are coupled by OT (§2.2). For each variant, we parameterize the velocity field with a multi-layer perceptron and trained it for the same number of iterations. More details are provided in §C.

Figure 2 (left) shows the learned probability paths using an isotropic Gaussian latent (top row) compared with PFM using $C = 2$ (middle) and $C = 4$ (bottom) modes. With the isotropic Gaussian, the generated density ($t = 1$) is blurry. As seen in earlier time steps ($t = 0.2, 0.4$), the velocity field has to 'separate' different modes. In contrast, the MoG latent generates a cleaner density at $t = 1$. One can observe that at $t = 0$, the latent already captures the coarse structure of the checkerboard, enabling the velocity field to perform less *reorganization* during the flow. Figure 2 (right) further demonstrates that PFM provides higher generation quality at low NFE: with NFE=4, the isotropic baseline (top) produces a blurred density, while MoGs with $C = 4$ and $C = 8$ (middle and bottom) both yield a sharp checkerboard pattern.

5.2 Unconditional Image Generation on Realistic Datasets

The goal of the experiments here is to show that FM models benefit from replacing the latent distribution from an isotropic Gaussian to a learned MoG under comparable experiment settings. Meanwhile, we acknowledge that the performance can be further improved by extensively searching the hyperparameters for individual methods, which require significant computational resources and is covered by prior works; see for instance [12, 20, 21].

Method. We evaluate PFM on unconditional image generation using CIFAR-10 and ImageNet, where the input images are of size 32×32 pixels. We train both the baseline OT-CFM and the proposed PFM under identical settings, with the only difference being that the former uses an isotropic Gaussian as the latent distribution and the latter uses a learned MoG. In particular, the models are trained in pixel space and the velocity fields are parameterized by the ADM architecture [10]. For reference, we also include the results reported by existing works, including DDPM [17], Stochastic Interpolants [1], FM [26], I-CFM

and OT-CFM [40], MTC [24], and MFM [33]. We provide more implementation details as well as additional experiments in Appendix C.

Metrics. We report Fréchet Inception Distance (FID) [16] computed over 50K generated samples. At inference, we integrate the learned velocity field using the midpoint ODE solver and vary the NFE to assess sampling efficiency.

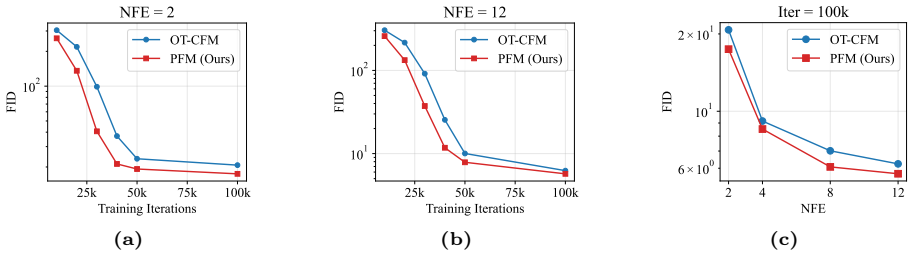
Table 2: Unconditional generation results on CIFAR-10 (left) and ImageNet 32×32 (right). A "*" next to the method indicate the numbers are reproduced since they are not reported in original papers.

Method	Solver	NFE	FID↓
ODE/SDE-based methods			
DDPM	–	1000	3.17
DDPM	dopri5	274	7.48
St. Interpolants	–	–	10.27
FM	dopri5	142	6.35
I-CFM	euler	100	4.46
I-CFM	euler	1000	3.64
I-CFM	dopri5	146	3.66
Improved training			
OT-CFM*	midpoint	4	9.20
OT-CFM*	midpoint	8	7.02
OT-CFM*	midpoint	12	6.24
MTC	heun	5	18.74
Our results			
PFM	midpoint	4	8.52
PFM	midpoint	8	6.07
PFM	midpoint	12	5.71

Method	Solver	NFE	FID↓
ODE/SDE-based methods			
St. Interpolants	–	–	8.49
FM	dopri5	122	5.02
Improved training			
MFM - BatchOT	midpoint	4	17.28
MFM - BatchOT	midpoint	8	8.73
MFM - BatchOT	midpoint	12	7.18
MFM - Stable	midpoint	4	21.82
MFM - Stable	midpoint	8	9.99
MFM - Stable	midpoint	12	7.84
Our results			
PFM	midpoint	4	9.90
PFM	midpoint	8	6.84
PFM	midpoint	12	6.52

Results on CIFAR. Figure 3 and Table 2 (left) summarize the results on CIFAR-10. Notably, under identical training settings, PFM consistently outperforms the OT-CFM baseline across all NFE budgets: at NFE = 4, PFM achieves an FID of 8.52 versus 9.20 for OT-CFM; and at NFE = 12, 5.71 versus 6.24. These improvements are most pronounced at low NFE, which confirms that the MoG latent produces simpler flow trajectories that are easier to integrate with few steps. Further, Figure 3 (a–b) shows that PFM reaches the same FID as OT-CFM in roughly 25–50% fewer training iterations: e.g., at NFE = 2, PFM attains the final OT-CFM quality in 50K iterations, whereas OT-CFM requires the full 100K. Figure 3 (c) displays the FID as a function of NFE at 100K iterations, showing that PFM dominates OT-CFM across the entire NFE range.

Results on ImageNet (32×32). Table 2 (right) reports results on ImageNet (32×32). As seen, PFM improves over alternative methods given the same NFE budget. For instance, it achieves FIDs of 9.90, 6.84, and 6.52 at NFE = 4, 8, and 12, respectively, compared to 17.28, 8.73, and 7.18 for MFM-BatchOT.



(a) **(b)** **(c)**

Fig. 3: PFM yields lower FID than the baseline with the same number of training iterations. Both methods are trained with identical settings on CIFAR-10, except that OT-CFM uses isotropic Gaussian and PFM uses learned MoG as the latent distribution.

6 Conclusion

In this paper, we introduce PFM, which replaces the isotropic Gaussian latent with an MoG whose modes better align with the structure of the data. We investigate practical design choices of the MoG latent distribution that enables efficient FM, and justify them through empirical and theoretical analysis. On CIFAR-10 and ImageNet, PFM matches the generation quality of OT-CFM in much fewer training iterations and produces higher-quality samples for the same NFE. These results confirm that aligning the latent distribution with the data structure is a simple yet effective strategy for improving both training efficiency and sample quality in FM.

Limitations and future work. Our current experiments focus on unconditional generation at moderate resolutions. Extending PFM to class-conditional generation or higher-resolution generation is a natural next step. Additionally, exploring more expressive latent families beyond MoGs, as well as adaptive strategies for selecting the number of mixture components, are promising directions. Finally, integrating PFM with improved ODE solvers and distillation techniques may yield further gains in few-step generation.

Acknowledgements

The authors acknowledge the support of the NSF under grant 2031985, the Simons Foundation under grant 814201, the ONR MURI Program under grant 503405-78051, and the University of Pennsylvania Startup Funds. TD further acknowledges the Penn AI Fellowship.

References

1. Albergo, M.S., Vanden-Eijnden, E.: Building Normalizing Flows with Stochastic Interpolants. In: International Conference on Learning Representations (2023)
2. Benamou, J.D., Brenier, Y.: A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. Numerische Mathematik **84**(3), 375–393 (Jan 2000)

3. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (Aug 2013)
4. Brown, B.C.A., Caterini, A.L., Ross, B.L., Cresswell, J.C., Loaiza-Ganem, G.: Verifying the Union of Manifolds Hypothesis for Image Data. In: International Conference on Learning Representations (2023)
5. Buchanan, S., Pai, D., Wang, P., Ma, Y.: Principles and Practice of Deep Representation Learning. Online (Aug 2025)
6. Chen, H., Zhang, K., Tan, H., Xu, Z., Luan, F., Guibas, L., Wetzstein, G., Bi, S.: Gaussian Mixture Flow Matching Models. In: International Conference on Machine Learning (2025)
7. Chen, H., Han, Y., Chen, F., Li, X., Wang, Y., Wang, J., Wang, Z., Liu, Z., Zou, D., Raj, B.: Masked autoencoders are effective tokenizers for diffusion models. In: International Conference on Machine Learning (2025)
8. Dao, Q., Phung, H., Nguyen, B., Tran, A.: Flow Matching in Latent Space. arXiv:2307.08698 (2023)
9. Davtyan, A., Dadi, L.T., Cevher, V., Favaro, P.: Faster inference of flow-based generative models via improved data-noise coupling. In: International Conference on Learning Representations (2025)
10. Dhariwal, P., Nichol, A.Q.: Diffusion models beat GANs on image synthesis. In: Advances in Neural Information Processing Systems (2021)
11. Ding, T., Lim, D., Vidal, R., Haeffele, B.D.: Understanding Doubly Stochastic Clustering. In: International Conference on Machine Learning. pp. 5153–5165 (2022)
12. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Rombach, R.: Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. In: International Conference on Machine Learning (2024)
13. Fefferman, C., Mitter, S., Narayanan, H.: Testing the manifold hypothesis. Journal of the American Mathematical Society **29**(4), 983–1049 (Feb 2016)
14. Geng, Z., Deng, M., Bai, X., Kolter, J.Z., He, K.: Mean Flows for One-step Generative Modeling. In: Advances in Neural Information Processing Systems. vol. 38 (2025)
15. Guo, P., Schwing, A.G.: Variational Rectified Flow Matching. In: International Conference on Machine Learning (2025)
16. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in Neural Information Processing Systems **30** (2017)
17. Ho, J., Jain, A., Abbeel, P.: Denoising Diffusion Probabilistic Models. In: Advances in Neural Information Processing Systems. vol. 33, pp. 6840–6851 (2020)
18. Jia, N., Zhu, T., Liu, H., Zheng, Z.: Structured Diffusion Models with Mixture of Gaussians as Prior Distribution. arXiv:2410.19149 (2024)
19. Jin, C., Zhang, Y., Balakrishnan, S., Wainwright, M.J., Jordan, M.I.: Local maxima in the likelihood of gaussian mixture models: Structural results and algorithmic consequences. Advances in neural information processing systems **29** (2016)
20. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the Design Space of Diffusion-Based Generative Models. In: Advances in Neural Information Processing Systems. vol. 35 (2022)
21. Karras, T., Aittala, M., Lehtinen, J., Hellsten, J., Aila, T., Laine, S.: Analyzing and improving the training dynamics of diffusion models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24174–24184 (2024)

22. Kornilov, N., Mokrov, P., Gasnikov, A., Korotin, A.: Optimal Flow Matching: Learning Straight Trajectories in Just One Step. In: Advances in Neural Information Processing Systems. vol. 37 (2024)
23. Lee, J., Kim, K., Lee, J.: Is there a better source distribution than gaussian? Exploring source distributions for image flow matching. Transactions on Machine Learning Research (2026)
24. Lee, S., Kim, B., Ye, J.C.: Minimizing trajectory curvature of ODE-based generative models. In: International Conference on Machine Learning (2023)
25. Lerman, G., Maunu, T.: An Overview of Robust Subspace Recovery. Proceedings of the IEEE **106**, 1380–1410 (2018)
26. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow Matching for Generative Modeling. In: International Conference on Learning Representations (2023)
27. Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R.T.Q., Lopez-Paz, D., Ben-Hamu, H., Gat, I.: Flow Matching Guide and Code. arXiv:2412.06264 (2024)
28. Liu, Q.: Rectified Flow: A Marginal Preserving Approach to Optimal Transport. arXiv:2209.14577 (2022)
29. Liu, X., Gong, C., Liu, Q.: Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In: International Conference on Learning Representations (2023)
30. Loaiza-Ganem, G., Ross, B.L., Hosseinzadeh, R., Caterini, A.L., Cresswell, J.C.: Deep Generative Models through the Lens of the Manifold Hypothesis: A Survey and New Connections. Transactions on Machine Learning Research (2024)
31. Luo*, J., Ding*, T., Chan, K.H.R., Thaker, D., Chattopadhyay, A., Callison-Burch, C., Vidal, R.: PaCE: Parsimonious Concept Engineering for Large Language Models. In: Advances in Neural Information Processing Systems (Jun 2024), https://arxiv.org/pdf/2406.04331
32. Nesterov, Y.: Lectures on Convex Optimization, Springer Optimization and Its Applications, vol. 137. Springer International Publishing, Cham (2018)
33. Pooladian, A.A., Ben-Hamu, H., Domingo-Enrich, C., Amos, B., Lipman, Y., Chen, R.T.Q.: Multisample Flow Matching: Straightening Flows with Minibatch Couplings. In: International Conference on Machine Learning (2023)
34. Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., Goldstein, T.: The Intrinsic Dimension of Images and Its Impact on Learning. In: International Conference on Learning Representations (2021)
35. Selim, S.Z., Ismail, M.A.: K-means-type algorithms: A generalized convergence theorem and characterization of local optimality. IEEE Transactions on Pattern Analysis and Machine Intelligence **6**(1), 81–87 (Jan 1984)
36. Shaul, N., Perez, J., Chen, R.T.Q., Thabet, A., Pumarola, A., Lipman, Y.: Bespoke solvers for generative flow models. In: International Conference on Learning Representations (2024)
37. Shireman, E.M., Steinley, D., Brusco, M.J.: Local optima in mixture modeling. Multivariate behavioral research **51**(4), 466–481 (2016)
38. Song, Y., Ermon, S.: Generative Modeling by Estimating Gradients of the Data Distribution. In: Advances in Neural Information Processing Systems. vol. 32, pp. 11895–11907 (2019)
39. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. Journal of the Royal Statistical Society Series B: Statistical Methodology **61**(3), 611–622 (1999)

40. Tong, A., Fatras, K., Malkin, N., Huguet, G., Zhang, Y., Rector-Brooks, J., Wolf, G., Bengio, Y.: Improving and generalizing flow-based generative models with minibatch optimal transport. Transactions on Machine Learning Research (2024)
41. Vidal, R., Ma, Y., Sastry, S.: Generalized Principal Component Analysis, Interdisciplinary Applied Mathematics, vol. 40. Springer New York, New York, NY (2016)
42. Yang, L., Zhang, Z., Zhang, Z., Liu, X., Xu, M., Zhang, W., Meng, C., Ermon, S., Cui, B.: Consistency Flow Matching: Defining Straight Flows with Velocity Consistency. arXiv:2407.02398 (2024)
43. Zhang, Y., Yan, Y., Schwing, A., Zhao, Z.: Towards hierarchical rectified flow. In: International Conference on Learning Representations (2025)

Organization of the Appendix

- In §A, we review prior works in flow matching, focusing on trajectory straightening for FM and a self-contained derivation of FM.
- In §B, we provide proofs for all theoretical results stated in the main paper, as well as extra results including affine velocity field with a bias term.
- In §C, we present extra experiments (simulation and ablation study) as well as details on the experiment setup.
- In §D, we compile all the notations used in the paper.

A Review of Prior Art

In §2.1, we review prior works on trajectory straightening for FM. We are aware that more techniques for improving training and sampling efficiency of FM exist, as briefly pointed out in §1. The proposed PFM method can be viewed as orthogonal improvements, since PFM focuses on the design of the latent distribution. In particular, PFM can be used in conjunction with these approaches.

In §A.2, we review a minimal construction of FM used in the paper. For more derivations and properties of FM, we refer readers to the original papers [1,26,29] and a recent survey [27]. See also a survey connecting generative models to the manifold hypothesis [30].

A.1 Trajectory Straightening for FM

A key insight for efficient flow matching is that straighter flow trajectories require a smaller NFE at inference time. At least two families of approaches arise:

- **Optimal Transport Coupling.** The motivation traces back to the optimization problem of *dynamic OT*, where the optimal flow trajectory is guaranteed to be straight [2, Proposition 1.1]. To learn such a flow, two types of methods emerge. One is called *rectified flow* [28, 29], which involves iteratively learning a new flow using the coupling induced by the previous flow; such a procedure is shown to be a certain alternating descent algorithm on the dynamic OT objective [28, §5.4]. The other method is *mini-batch OT*, which trains the flow with data and latent samples paired by OT on each mini-batch; we covered this in §2.2. It has been shown that mini-batch OT optimizes the dynamic OT objective in the limit under certain assumptions [33, Theorem 4.2] and [40, Proposition 3.4].
- **Velocity Regularization.** Another family of approaches regularizes the velocity field via loss terms or parameterizations. For example, the work of [42] promotes the velocity fields along the flow trajectory to be consistent, or piece-wise so. Meanwhile, the authors of [22] constructs the vector field to be gradients of convex functions and the trajectories as straight lines, and shows that the optimal such velocity field solves the dynamic OT problem. Further, the work of [14] devised a way to learn *average velocity*, which is

the displacement along a flow trajectory divided by the time interval. Such an average velocity is straight in the sense that it allows for computing a partial flow map from any start and end time with one NFE.

A.2 Review of Flow Matching

We use flow-matching to build our framework. The basic element in flow-matching is the velocity field $u_t : \mathbb{R}^d \to \mathbb{R}^d$ for time $t \in [0, 1]$, which defines an ordinary differential equation (ODE) for the flow $\phi_t : \mathbb{R}^d \to \mathbb{R}^d$:

$$\frac{d}{dt}\phi_t(\boldsymbol{x}) = u_t(\phi_t(\boldsymbol{x})), \quad \phi_0(\boldsymbol{x}) = \boldsymbol{x}. \tag{22}$$

The flow ϕ_t is applied to an initial distribution p_0 so that the marginal distribution at time t is $p_t = (\phi_t)_{\#}p_0$. With the above context, the goal of flow-matching is to design a ground-truth u_t such that

$$p_0 \approx q_0 \quad \text{and} \quad p_1 \approx q_1, \tag{23}$$

and learn to estimate u_t via a neural network v_θ.

One major design choice is the coupling distribution $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ of latent $\boldsymbol{x}_0$ and data $\boldsymbol{x}_1$. For now, we assume $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ exists. The conditional flow matching framework [40] provides a way to construct the probability path p_t and velocity field u_t. In particular, it defines a probability path $p_t(\boldsymbol{x})$ as

$$p_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1) = \mathcal{N}\left(\boldsymbol{x}|t\boldsymbol{x}_1 + (1-t)\boldsymbol{x}_0, \sigma^2\right), \tag{24}$$

$$p_t(\boldsymbol{x}) = \int p_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1)q(\boldsymbol{x}_0, \boldsymbol{x}_1)d\boldsymbol{x}_0 d\boldsymbol{x}_1, \tag{25}$$

which satisfies the following property:

Proposition 1. *The probability path p_t corresponding to $q(\boldsymbol{x}_0, \boldsymbol{x}_1)$ satisfies that $p_1 = q_1 * \mathcal{N}\left(0, \sigma^2\right)$ and $p_0 = q_0 * \mathcal{N}\left(0, \sigma^2\right)$, where $*$ is the convolution operator.*

A proof can be adapted from [40]. When $\sigma \to 0$, $\mathcal{N}\left(\boldsymbol{x} \mid 0, \sigma^2\right)$ becomes a Dirac delta function, so $p_1 = q_1$ and $p_0 = q_0$; this satisfies the goal in (23). Now, the velocity field $u_t(\boldsymbol{x})$ is defined by

$$u_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1) = \boldsymbol{x}_1 - \boldsymbol{x}_0, \tag{26}$$

$$u_t(\boldsymbol{x}) = \mathbb{E}_{q(\boldsymbol{x}_0, \boldsymbol{x}_1)} \frac{u_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1)p_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1)}{p_t(\boldsymbol{x})}. \tag{27}$$

It is not hard to show that the so-defined velocity field $u_t(\boldsymbol{x})$ indeed generates the probability path $p_t(\boldsymbol{x})$, i.e., they satisfy the continuity equation.

To learn the velocity field, one minimizes the following objective:

$$\mathcal{L}^{\text{CFM}}(\theta) := \mathbb{E}_{\substack{t, q(\boldsymbol{x}_0, \boldsymbol{x}_1), \\ p_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1)}} \left\| v_\theta(\boldsymbol{x}, t) - u_t(\boldsymbol{x}|\boldsymbol{x}_0, \boldsymbol{x}_1) \right\|^2$$

The training algorithm is given in Algorithm 1.

Algorithm 1 Flow Matching Training

1: **for** each training step **do**
2: Sample $\boldsymbol{x}_0, \boldsymbol{x}_1 \sim q(\boldsymbol{x}_0, \boldsymbol{x}_1)$
3: Sample $t \sim U[0,1]$
4: Compute $\boldsymbol{x}_t = t\boldsymbol{x}_1 + (1-t)\boldsymbol{x}_0$
5: Compute $g = \nabla_\theta \|v_\theta(\boldsymbol{x}_t, t) - (\boldsymbol{x}_1 - \boldsymbol{x}_0)\|_2^2$
6: Update θ using optimizer with g
7: **end for**

B Proofs and Auxiliary Theoretical Results

B.1 Complementary Results

In §4, we presented the key theoretical results of this paper. To complete the picture and provide a more detailed characterization of convergence for flow matching with affine velocity fields, we establish several additional results in this section.

First, we will review the classic convergence results for convergence rate of GD for regression problems. Similar results and proof can be found in [32, Theorem 2.1.15].

Lemma 3 (Linear regression: geometry and rates of convergence). *Let $\boldsymbol{z} \in \mathbb{R}^p$ and $\boldsymbol{y} \in \mathbb{R}^d$ be random vectors with $\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top] := \boldsymbol{Z} \succ 0$. Consider the expected least squares*

$$L(\boldsymbol{W}) := \mathbb{E}\big[\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{y}\|_2^2\big], \qquad \boldsymbol{W} \in \mathbb{R}^{d\times p}.$$

Let $\boldsymbol{C} := \mathbb{E}[\boldsymbol{y}\boldsymbol{z}^\top]$. Then:

1. ***Gradient & Hessian.***

$$\nabla_{\boldsymbol{W}} L(\boldsymbol{W}) = 2\,(\boldsymbol{W}\boldsymbol{Z} - \boldsymbol{C}).$$

The Hessian as a linear map acting on a perturbation $\Delta \in \mathbb{R}^{d\times p}$ is

$$\nabla^2 L(\boldsymbol{W})[\Delta] = 2\,\Delta\,\boldsymbol{Z}.$$

2. ***Strong convexity & smoothness.*** *$L(\boldsymbol{W})$ is μ-strongly convex and L-smooth with respect to $\|\cdot\|_F$, where*

$$\mu = 2\,\lambda_{\min}(\boldsymbol{Z}), \qquad L = 2\,\lambda_{\max}(\boldsymbol{Z}).$$

3. ***Exact dynamics of GD.*** *Let $\boldsymbol{W}_\star$ be the unique minimizer ($\boldsymbol{W}_\star\boldsymbol{Z} = \boldsymbol{C}$). Consider GD with step size $\eta > 0$:*

$$\boldsymbol{W}_{k+1} = \boldsymbol{W}_k - \eta\nabla_{\boldsymbol{W}} L(\boldsymbol{W}_k) = \boldsymbol{W}_k - 2\eta(\boldsymbol{W}_k\boldsymbol{Z} - \boldsymbol{C}).$$

Then the iterates satisfy

$$\boldsymbol{W}_k - \boldsymbol{W}_\star = (\boldsymbol{W}_0 - \boldsymbol{W}_\star)\,(\boldsymbol{I}_p - 2\eta\,\boldsymbol{Z})^k.$$

Further, the objective gap satisfies

$$L(\boldsymbol{W}_k) - L(\boldsymbol{W}_\star) = \sum_{i=1}^{p} \lambda_i \left|1 - 2\eta\lambda_i\right|^{2k} \|\boldsymbol{g}_i\|_2^2,$$

where $\boldsymbol{Z} = \boldsymbol{U}\,\mathrm{diag}(\{\lambda_i\}_{i=1}^p)\,\boldsymbol{U}^\top$ *is the eigendecomposition and* $\boldsymbol{g}_i := (\boldsymbol{W}_0 - \boldsymbol{W}_\star)\,\boldsymbol{u}_i \in \mathbb{R}^d$. *Each eigendirection contracts as* $\|(\boldsymbol{W}_k - \boldsymbol{W}_\star)\,\boldsymbol{u}_i\|_2 = \left|1 - 2\eta\lambda_i\right|^k \|(\boldsymbol{W}_0 - \boldsymbol{W}_\star)\,\boldsymbol{u}_i\|_2$.

4. ***Canonical rate with the optimal fixed step.*** *If the step size is set to* $\eta^\star = \frac{2}{\mu+L} = \frac{1}{\lambda_{\min}(\boldsymbol{Z})+\lambda_{\max}(\boldsymbol{Z})}$, *then*

$$\|\boldsymbol{W}_k - \boldsymbol{W}_\star\|_F \le \rho^k \|\boldsymbol{W}_0 - \boldsymbol{W}_\star\|_F, \tag{28}$$

$$L(\boldsymbol{W}_k) - L(\boldsymbol{W}_\star) \le \rho^{2k} \big(L(\boldsymbol{W}_0) - L(\boldsymbol{W}_\star)\big), \tag{29}$$

where $\rho := \frac{\kappa(\boldsymbol{Z})-1}{\kappa(\boldsymbol{Z})+1}$.

Proof. We prove the parts in order.

Gradient & Hessian. Let us expand the objective:

$$\begin{aligned} L(\boldsymbol{W}) &= \mathbb{E}\big[\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{y}\|_2^2\big] = \mathbb{E}\big[\mathrm{Tr}\big((\boldsymbol{W}\boldsymbol{z} - \boldsymbol{y})(\boldsymbol{W}\boldsymbol{z} - \boldsymbol{y})^\top\big)\big] \\ &= \mathrm{Tr}(\boldsymbol{W}\,\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top]\,\boldsymbol{W}^\top) - 2\,\mathrm{Tr}(\mathbb{E}[\boldsymbol{y}\boldsymbol{z}^\top]\,\boldsymbol{W}^\top) + \mathbb{E}[\|\boldsymbol{y}\|_2^2] \\ &= \mathrm{Tr}(\boldsymbol{W}\boldsymbol{Z}\boldsymbol{W}^\top) - 2\,\mathrm{Tr}(\boldsymbol{C}\boldsymbol{W}^\top) + \mathbb{E}[\|\boldsymbol{y}\|_2^2]. \end{aligned}$$

Therefore, the gradient is given by

$$\nabla_{\boldsymbol{W}} L(\boldsymbol{W}) = 2\boldsymbol{W}\boldsymbol{Z} - 2\boldsymbol{C} = 2(\boldsymbol{W}\boldsymbol{Z} - \boldsymbol{C}).$$

For any perturbation Δ, the differential of the gradient is

$$\nabla^2 L(\boldsymbol{W})[\Delta] = 2\,\Delta\,\boldsymbol{Z}.$$

Strong convexity & smoothness. For any Δ, we have that

$$\begin{aligned} &\langle \Delta, \nabla^2 L(\boldsymbol{W})[\Delta]\rangle = 2\ \mathrm{tr}(\Delta\boldsymbol{Z}\Delta^\top) \\ &\in \big[2\lambda_{\min}(\boldsymbol{Z})\|\Delta\|_F^2,\ 2\lambda_{\max}(\boldsymbol{Z})\|\Delta\|_F^2\big]. \end{aligned} \tag{30}$$

Exact dynamics of GD. Note that

$$\begin{aligned} \boldsymbol{W}_{k+1} &= \boldsymbol{W}_k - \eta\,\nabla_{\boldsymbol{W}} L(\boldsymbol{W}_k) \\ &= \boldsymbol{W}_k - 2\eta(\boldsymbol{W}_k\boldsymbol{Z} - \boldsymbol{C}) = \boldsymbol{W}_k(\boldsymbol{I}_p - 2\eta\boldsymbol{Z}) + 2\eta\boldsymbol{C}. \end{aligned} \tag{31}$$

Further, since the optimum satisfies $\boldsymbol{W}_\star\boldsymbol{Z} = \boldsymbol{C}$, we have

$$\boldsymbol{W}_\star = \boldsymbol{W}_\star(\boldsymbol{I}_p - 2\eta\boldsymbol{Z}) + 2\eta\boldsymbol{C}, \tag{32}$$

$$\Rightarrow\ \boldsymbol{W}_{k+1} - \boldsymbol{W}_\star = (\boldsymbol{W}_k - \boldsymbol{W}_\star)(\boldsymbol{I}_p - 2\eta\boldsymbol{Z}). \tag{33}$$

Iterating the relation yields

$$\boldsymbol{W}_k - \boldsymbol{W}_\star := (\boldsymbol{W}_0 - \boldsymbol{W}_\star)(\boldsymbol{I}_p - 2\eta\boldsymbol{Z})^k. \tag{34}$$

Moreover, starting from the quadratic expansion of L and subtracting the value at the optimum,

$$\begin{aligned}
&L(\boldsymbol{W}) - L(\boldsymbol{W}_\star) && (35)\\
&= \operatorname{tr}(\boldsymbol{W}\boldsymbol{Z}\boldsymbol{W}^\top) - 2\operatorname{tr}(\boldsymbol{C}\boldsymbol{W}^\top) && (36)\\
&\quad - \operatorname{tr}(\boldsymbol{W}_\star\boldsymbol{Z}\boldsymbol{W}_\star^\top) + 2\operatorname{tr}(\boldsymbol{C}\boldsymbol{W}_\star^\top) && (37)\\
&= \operatorname{tr}\left((\boldsymbol{W} - \boldsymbol{W}_\star)\boldsymbol{Z}(\boldsymbol{W} - \boldsymbol{W}_\star)^\top\right) && (38)\\
&\quad + 2\operatorname{tr}\left(\boldsymbol{W}_\star\boldsymbol{Z}(\boldsymbol{W} - \boldsymbol{W}_\star)^\top\right) - 2\operatorname{tr}\left(\boldsymbol{C}(\boldsymbol{W} - \boldsymbol{W}_\star)^\top\right)\\
&= \operatorname{tr}\left((\boldsymbol{W} - \boldsymbol{W}_\star)\boldsymbol{Z}(\boldsymbol{W} - \boldsymbol{W}_\star)^\top\right), && (39)
\end{aligned}$$

where we used $\boldsymbol{W}_\star\boldsymbol{Z} = \boldsymbol{C}$. Plugging in $\boldsymbol{W}_k$, we obtain

$$\begin{aligned}
&L(\boldsymbol{W}_k) - L(\boldsymbol{W}_\star) && (40)\\
&= \operatorname{tr}\left((\boldsymbol{W}_0 - \boldsymbol{W}_\star)(\boldsymbol{I}_p - 2\eta\boldsymbol{Z})^k\,\boldsymbol{Z}\,(\boldsymbol{I}_p - 2\eta\boldsymbol{Z})^k(\boldsymbol{W}_0 - \boldsymbol{W}_\star)^\top\right)\\
&= \operatorname{tr}\left((\boldsymbol{W}_0 - \boldsymbol{W}_\star)\,\boldsymbol{U}\operatorname{diag}\big((1 - 2\eta\lambda_i)^{2k}\lambda_i\big)\,\boldsymbol{U}^\top\,(\boldsymbol{W}_0 - \boldsymbol{W}_\star)^\top\right)\\
&= \operatorname{tr}\left(\boldsymbol{G}\operatorname{diag}\big((1 - 2\eta\lambda_i)^{2k}\lambda_i\big)\,\boldsymbol{G}^\top\right) && (41)\\
&= \sum_i \lambda_i\,|1 - 2\eta\lambda_i|^{2k}\,\|\boldsymbol{g}_i\|_2^2. && (42)
\end{aligned}$$

Thus each eigendirection contracts by $|1 - 2\eta\lambda_i|$ per step. The step size that minimizes $\max_i |1 - 2\eta\lambda_i|$ is $\eta^\star = \frac{1}{\lambda_1 + \lambda_p} = \frac{2}{\mu + L}$, which yields the rates

$$\|\boldsymbol{W}_k - \boldsymbol{W}_\star\|_F \le \rho^k \|\boldsymbol{W}_0 - \boldsymbol{W}_\star\|_F, \tag{43}$$

$$L(\boldsymbol{W}_k) - L(\boldsymbol{W}_\star) \le \rho^{2k}\big(L(\boldsymbol{W}_0) - L(\boldsymbol{W}_\star)\big). \tag{44}$$

where $\rho = \frac{\kappa(\boldsymbol{Z})-1}{\kappa(\boldsymbol{Z})+1}$.

Recall in Lemma 1, we characterize the convergence of GD for centered case. In this section, we provide a more general case where we allow the data and latent distributions to have arbitrary mean. Here, we will consider the following affine velocity models $v_\theta(\boldsymbol{x}, t) = \boldsymbol{A}_t\boldsymbol{x} + \boldsymbol{b}_t$ by minimizing the flow-matching loss

$$\mathcal{L}_t(\boldsymbol{A}_t, \boldsymbol{b}_t) = \mathbb{E}\left[\left\|\boldsymbol{A}_t\boldsymbol{x}_t + \boldsymbol{b}_t - \boldsymbol{y}\right\|_2^2\right]. \tag{45}$$

Proposition 2 (GD: Rate of Convergence (Non-Centered)). *Fix $t \in [0, 1]$ and consider GD on $\mathcal{L}_t(\boldsymbol{A}_t, \boldsymbol{b}_t)$ in (45). Define*

$$\boldsymbol{Z}_{\text{aug}} := \begin{bmatrix} \boldsymbol{\Sigma}_t & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{\mu}_t \\ 1 \end{bmatrix}\begin{bmatrix} \boldsymbol{\mu}_t \\ 1 \end{bmatrix}^\top.$$

With an optimal step size $\eta^\star = \frac{1}{\lambda_{\min}(\boldsymbol{Z}_{\text{aug}})+\lambda_{\max}(\boldsymbol{Z}_{\text{aug}})}$*, the iterates satisfy*

$$\left\|[\boldsymbol{A}_t^{(k)}\ \boldsymbol{b}_t^{(k)}] - [\boldsymbol{A}_t^\star\ \boldsymbol{b}_t^\star]\right\|_F \le \rho_{\text{aug}}^k \left\|[\boldsymbol{A}_t^{(0)}\ \boldsymbol{b}_t^{(0)}] - [\boldsymbol{A}_t^\star\ \boldsymbol{b}_t^\star]\right\|_F, \tag{46}$$

$$\mathcal{L}_t(\boldsymbol{A}_t^{(k)}, \boldsymbol{b}_t^{(k)}) - \mathcal{L}_t^\star \le \rho_{\text{aug}}^{2k}\big(\mathcal{L}_t(\boldsymbol{A}_t^{(0)}, \boldsymbol{b}_t^{(0)}) - \mathcal{L}_t^\star\big), \tag{47}$$

where $\rho_{\text{aug}} := \frac{\kappa(\boldsymbol{Z}_{\text{aug}})-1}{\kappa(\boldsymbol{Z}_{\text{aug}})+1}$.

Proof (Proof of Proposition 2). Set in Lemma 3 the feature and weight to

$$\boldsymbol{z} := \begin{bmatrix}\boldsymbol{x}_t \\ 1\end{bmatrix}, \qquad \boldsymbol{W} := \begin{bmatrix}\boldsymbol{A}\ \boldsymbol{b}\end{bmatrix} \in \mathbb{R}^{d\times(d+1)}.$$

Then $\mathcal{L}_t(\boldsymbol{A}, \boldsymbol{b}) = \mathbb{E}\big[\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{y}\|_2^2\big]$ is exactly of the form in Lemma 3, with second moment

$$\begin{aligned}\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top] &= \begin{bmatrix}\mathbb{E}[\boldsymbol{x}_t\boldsymbol{x}_t^\top] & \mathbb{E}[\boldsymbol{x}_t] \\ \mathbb{E}[\boldsymbol{x}_t]^\top & 1\end{bmatrix} \\ &= \begin{bmatrix}\boldsymbol{\Sigma}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top & \boldsymbol{\mu}_t \\ \boldsymbol{\mu}_t^\top & 1\end{bmatrix} \\ &= \begin{bmatrix}\boldsymbol{\Sigma}_t & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0\end{bmatrix} + \begin{bmatrix}\boldsymbol{\mu}_t \\ 1\end{bmatrix}\begin{bmatrix}\boldsymbol{\mu}_t \\ 1\end{bmatrix}^\top = \boldsymbol{Z}_{\text{aug}}.\end{aligned} \tag{48}$$

By Lemma 3, using the optimal fixed step $\eta^\star = \frac{1}{\lambda_{\min}(\boldsymbol{Z}_{\text{aug}})+\lambda_{\max}(\boldsymbol{Z}_{\text{aug}})}$ yields the stated rates.

Finally, we derive closed-form expression for $\boldsymbol{A}_t^*$ to complete this section.

Proposition 3 ($\mathcal{L}_t$-Optimal Affine Flow). *Assume $\boldsymbol{\Sigma}_t$ is invertible. The minimizer $(\boldsymbol{A}_t^\star, \boldsymbol{b}_t^\star)$ of $\mathcal{L}_t$ in (45) satisfies*

$$\boldsymbol{A}_t^\star = \boldsymbol{\Sigma}_{yt}\boldsymbol{\Sigma}_t^{-1}, \qquad \boldsymbol{b}_t^\star = \boldsymbol{\mu}_y - \boldsymbol{A}_t^\star\boldsymbol{\mu}_t, \tag{49}$$

which yields the optimal velocity field

$$v_t^\star(\boldsymbol{x}) = \boldsymbol{\Sigma}_{yt}\boldsymbol{\Sigma}_t^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_t) + \boldsymbol{\mu}_y. \tag{50}$$

Moreover, the optimal objective value is

$$\mathcal{L}_t^\star = \operatorname{Tr}(\boldsymbol{\Sigma}_{yy}) - \operatorname{Tr}\big(\boldsymbol{\Sigma}_{yt}\boldsymbol{\Sigma}_t^{-1}\boldsymbol{\Sigma}_{yt}^\top\big). \tag{51}$$

This result shows that under an affine velocity model, the optimal velocity field depends only on the first and second order statistics of $(\boldsymbol{x}_t, \boldsymbol{y})$, regardless of how complicated the distribution of $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ may be.

Proof (Proof of Proposition 3). We will use the identities

$$\mathbb{E}[\boldsymbol{x}_t\boldsymbol{x}_t^\top] = \boldsymbol{\Sigma}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top, \qquad \mathbb{E}[\boldsymbol{y}\boldsymbol{x}_t^\top] = \boldsymbol{\Sigma}_{yt} + \boldsymbol{\mu}_y\boldsymbol{\mu}_t^\top,$$

and let $\boldsymbol{\Sigma}_{yy} := \mathrm{Cov}(\boldsymbol{y})$ so that $\mathbb{E}[\boldsymbol{y}\boldsymbol{y}^\top] = \boldsymbol{\Sigma}_{yy} + \boldsymbol{\mu}_y\boldsymbol{\mu}_y^\top$.

$$\begin{aligned}
\mathcal{L}_t(\boldsymbol{A}, \boldsymbol{b}) &= \mathbb{E}\big[\|\boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{b} - \boldsymbol{y}\|_2^2\big] \qquad (52)\\
&= \mathbb{E}\Big[\mathrm{Tr}\big((\boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{b} - \boldsymbol{y})(\boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{b} - \boldsymbol{y})^\top\big)\Big]\\
&= \mathrm{Tr}\big(\boldsymbol{A}\,\mathbb{E}[\boldsymbol{x}_t\boldsymbol{x}_t^\top]\,\boldsymbol{A}^\top\big) + 2\,\mathrm{Tr}\big(\boldsymbol{b}\,\mathbb{E}[\boldsymbol{x}_t]^\top\boldsymbol{A}^\top\big)\\
&\quad - 2\,\mathrm{Tr}\big(\mathbb{E}[\boldsymbol{y}\boldsymbol{x}_t^\top]\,\boldsymbol{A}^\top\big) + \|\boldsymbol{b}\|_2^2 - 2\,\boldsymbol{b}^\top\mathbb{E}[\boldsymbol{y}] + \mathbb{E}[\|\boldsymbol{y}\|_2^2]\\
&= \mathrm{Tr}\big(\boldsymbol{A}(\boldsymbol{\Sigma}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top)\boldsymbol{A}^\top\big) + 2\,\mathrm{Tr}\big((\boldsymbol{b} - \boldsymbol{\mu}_y)\boldsymbol{\mu}_t^\top\boldsymbol{A}^\top\big)\\
&\quad - 2\,\mathrm{Tr}\big(\boldsymbol{\Sigma}_{yt}\boldsymbol{A}^\top\big) + \|\boldsymbol{b} - \boldsymbol{\mu}_y\|_2^2 + \mathrm{Tr}(\boldsymbol{\Sigma}_{yy}). \qquad (53)
\end{aligned}$$

The first-order optimality conditions are

$$\frac{\partial\mathcal{L}_t}{\partial\boldsymbol{b}} = 2(\boldsymbol{A}\boldsymbol{\mu}_t + \boldsymbol{b} - \boldsymbol{\mu}_y) = 0, \qquad (54)$$

$$\frac{\partial\mathcal{L}_t}{\partial\boldsymbol{A}} = 2\Big(\boldsymbol{A}(\boldsymbol{\Sigma}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top) + \boldsymbol{b}\boldsymbol{\mu}_t^\top - (\boldsymbol{\Sigma}_{yt} + \boldsymbol{\mu}_y\boldsymbol{\mu}_t^\top)\Big) = 0. \qquad (55)$$

Hence $\boldsymbol{b}^\star(\boldsymbol{A}) = \boldsymbol{\mu}_y - \boldsymbol{A}\boldsymbol{\mu}_t$, and substituting gives

$$\boldsymbol{A}\,\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_{yt}.$$

If $\boldsymbol{\Sigma}_t$ is invertible, (49) and (50) follow. Plugging $(\boldsymbol{A}, \boldsymbol{b}) = (\boldsymbol{A}_t^\star, \boldsymbol{b}_t^\star)$ into (53), the cross term with $\boldsymbol{\mu}_t$ vanishes and we obtain

$$\begin{aligned}
\mathcal{L}_t^\star &= \mathrm{tr}\big(\boldsymbol{A}_t^\star(\boldsymbol{\Sigma}_t + \boldsymbol{\mu}_t\boldsymbol{\mu}_t^\top)(\boldsymbol{A}_t^\star)^\top\big) - 2\,\mathrm{tr}\big(\boldsymbol{\Sigma}_{yt}(\boldsymbol{A}_t^\star)^\top\big)\\
&\quad + \|\boldsymbol{b}_t^\star - \boldsymbol{\mu}_y\|_2^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{yy})\\
&= \mathrm{tr}\big(\boldsymbol{A}_t^\star\boldsymbol{\Sigma}_t(\boldsymbol{A}_t^\star)^\top\big) - 2\,\mathrm{tr}\big(\boldsymbol{\Sigma}_{yt}(\boldsymbol{A}_t^\star)^\top\big) + 0 + \mathrm{tr}(\boldsymbol{\Sigma}_{yy})\\
&= -\,\mathrm{tr}\big(\boldsymbol{\Sigma}_{yt}(\boldsymbol{A}_t^\star)^\top\big) + \mathrm{tr}(\boldsymbol{\Sigma}_{yy})\\
&= -\,\mathrm{tr}\big(\boldsymbol{\Sigma}_{yt}\,\boldsymbol{\Sigma}_t^{-1}\,\boldsymbol{\Sigma}_{yt}^\top\big) + \mathrm{tr}(\boldsymbol{\Sigma}_{yy}),
\end{aligned}$$

where we used $\boldsymbol{A}_t^\star = \boldsymbol{\Sigma}_{yt}\boldsymbol{\Sigma}_t^{-1}$ and $\boldsymbol{b}_t^\star - \boldsymbol{\mu}_y = \boldsymbol{0}$. This proves (51).

B.2 Proof of Lemma 1

Proof. By definition, $\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1$ and $\boldsymbol{y} = \boldsymbol{x}_1 - \boldsymbol{x}_0$. For the covariances, use bilinearity of $\mathrm{Cov}(\cdot,\cdot)$:

$$\begin{aligned}
\boldsymbol{\Sigma}_t &= \mathrm{Cov}\big((1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1\big)\\
&= (1-t)^2\,\mathrm{Cov}(\boldsymbol{x}_0, \boldsymbol{x}_0) + t^2\,\mathrm{Cov}(\boldsymbol{x}_1, \boldsymbol{x}_1)\\
&\quad + t(1-t)\,\big(\mathrm{Cov}(\boldsymbol{x}_0, \boldsymbol{x}_1) + \mathrm{Cov}(\boldsymbol{x}_1, \boldsymbol{x}_0)\big)\\
&= (1-t)^2\,\boldsymbol{\Sigma}_0 + t^2\,\boldsymbol{\Sigma}_1 + t(1-t)\,(\boldsymbol{\Sigma}_{01} + \boldsymbol{\Sigma}_{01}^\top), \qquad (56)
\end{aligned}$$

and

$$\begin{aligned}\boldsymbol{\Sigma}_{yt} &= \mathrm{Cov}\big(\boldsymbol{x}_1 - \boldsymbol{x}_0,\ (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1\big) \\ &= (1-t)\left(\mathrm{Cov}(\boldsymbol{x}_1, \boldsymbol{x}_0) - \mathrm{Cov}(\boldsymbol{x}_0, \boldsymbol{x}_0)\right) \\ &\quad + t\left(\mathrm{Cov}(\boldsymbol{x}_1, \boldsymbol{x}_1) - \mathrm{Cov}(\boldsymbol{x}_0, \boldsymbol{x}_1)\right) \\ &= (1-t)\left(\boldsymbol{\Sigma}_{01}^{\top} - \boldsymbol{\Sigma}_0\right) + t\left(\boldsymbol{\Sigma}_1 - \boldsymbol{\Sigma}_{01}\right). \end{aligned} \tag{57}$$

If $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ are independent, then $\boldsymbol{\Sigma}_{01} = \mathbf{0}$, yielding the stated reductions.

B.3 Proof of Theorem 1

Proof. First, we prove that under the data assumptions in Section 4.3, (13)–(14) hold.

Conditioned on the class $z = c$, one first has

$$\boldsymbol{x}_1 = \boldsymbol{B}_c \boldsymbol{x}_0, \tag{58}$$

$$\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1 \tag{59}$$

$$= (1-t)\boldsymbol{\mu}_0^c + t\boldsymbol{\mu}_1^c + \boldsymbol{M}_{t,c}(\boldsymbol{x}_0 - \boldsymbol{\mu}_0^c), \tag{60}$$

where $\Delta_c := \boldsymbol{B}_c - \boldsymbol{I}$ and $\boldsymbol{M}_{t,c} := \boldsymbol{I} + t\Delta_c = (1-t)\boldsymbol{I} + t\boldsymbol{B}_c$. Therefore

$$\mathbb{E}[\boldsymbol{x}_t \mid z = c] = (1-t)\boldsymbol{\mu}_0^c + t\boldsymbol{\mu}_1^c, \tag{61}$$

$$\mathrm{Cov}(\boldsymbol{x}_t \mid z = c) = \boldsymbol{M}_{t,c}\boldsymbol{\Sigma}_0^c \boldsymbol{M}_{t,c}. \tag{62}$$

Using the expressions above and $\mathbb{E}[\boldsymbol{x}_0 - \boldsymbol{\mu}_0^c \mid z = c] = \mathbf{0}$,

$$\mathrm{Cov}(\boldsymbol{y}, \boldsymbol{x}_t \mid z = c)$$

$$= \Delta_c \mathbb{E}\big[(\boldsymbol{x}_0 - \boldsymbol{\mu}_0^c)(\boldsymbol{x}_0 - \boldsymbol{\mu}_0^c)^{\top} \mid z = c\big]\boldsymbol{M}_{t,c} \tag{63}$$

$$= \Delta_c \boldsymbol{\Sigma}_0^c \boldsymbol{M}_{t,c}. \tag{64}$$

Applying the law of total covariance yields

$$\boldsymbol{\Sigma}_t = \mathbb{E}\big[\mathrm{Cov}(\boldsymbol{x}_t \mid z)\big] + \mathrm{Cov}\big(\mathbb{E}[\boldsymbol{x}_t \mid z]\big) \tag{65}$$

$$= \sum_{c=1}^{C} \pi_c \boldsymbol{M}_{t,c}\boldsymbol{\Sigma}_0^c \boldsymbol{M}_{t,c} + \sum_{c=1}^{C} \pi_c \delta_t^c (\delta_t^c)^{\top}, \tag{66}$$

$$\boldsymbol{\Sigma}_{yt} = \mathbb{E}\big[\mathrm{Cov}(\boldsymbol{y}, \boldsymbol{x}_t \mid z)\big] + \mathrm{Cov}\big(\mathbb{E}[\boldsymbol{y} \mid z], \mathbb{E}[\boldsymbol{x}_t \mid z]\big) \tag{67}$$

$$= \sum_{c=1}^{C} \pi_c \Delta_c \boldsymbol{\Sigma}_0^c \boldsymbol{M}_{t,c} + \sum_{c=1}^{C} \pi_c (\delta_1^c - \delta_0^c)(\delta_t^c)^{\top}. \tag{68}$$

This proves (13)–(14).

Next, we prove the convergence rates in (15). For any fixed $t \in (0,1)$, one can see that the convergence rate is $\frac{\kappa(\boldsymbol{\Sigma}_t)-1}{\kappa(\boldsymbol{\Sigma}_t)+1}$. Thus, it suffices to show that

$$\kappa(\boldsymbol{\Sigma}_t) \le \kappa_t^{\mathrm{up}}. \tag{69}$$

We first prove a lower bound on $\lambda_{\min}(\Sigma_t)$.

$$\begin{aligned}\lambda_{\min}(\Sigma_t) &= \lambda_{\min}(\Sigma_{t,w} + \Sigma_{t,b}) \\ &\geq \lambda_{\min}(\Sigma_{t,w}) \\ &= \lambda_{\min}\left(\sum_{c=1}^{C} \pi_c A_{t,c}\right) \\ &\geq \left(\sum_{c=1}^{C} \frac{\pi_c}{\lambda_{\min}(A_{t,c})}\right)^{-1}, \end{aligned} \tag{70}$$

where we use Lemma 4 in the last inequality.

$$\begin{aligned}\lambda_{\min}(A_{t,c}) &= \lambda_{\min}(M_{t,c}\Sigma_0^c M_{t,c}) \\ &\geq \lambda_{\min}(\Sigma_0^c)\sigma_{\min}^2(M_{t,c}) \\ &= \lambda_{\min}(\Sigma_0^c)\sigma_{\min}^2(\boldsymbol{I} + t\Delta_c) \\ &\geq \lambda_{\min}(\Sigma_0^c)(1 - t\|\Delta_c\|)^2 \end{aligned} \tag{71}$$

Combine Eq. (70) and Eq. (71), one has

$$\lambda_{\min}(\Sigma_t) \geq \left(\sum_{c=1}^{C} \frac{\pi_c}{\lambda_{\min}(\Sigma_0^c)(1 - t\|\Delta_c\|)^2}\right)^{-1}. \tag{72}$$

Now, we derive an upper bound on $\lambda_{\max}(\Sigma_t)$.

$$\begin{aligned}\lambda_{\max}(\Sigma_t) &= \lambda_{\max}(\Sigma_{t,w} + \Sigma_{t,b}) \\ &\leq \lambda_{\max}(\Sigma_{t,w}) + \lambda_{\max}(\Sigma_{t,b}) \\ &\leq \sum_{c=1}^{C} \pi_c \lambda_{\max}(A_{t,c}) + \|\Sigma_{t,b}\| \\ &\leq \sum_{c=1}^{C} \pi_c \|\Sigma_0^c\|(1 + t\|\Delta_c\|)^2 + \|G_t\|. \end{aligned} \tag{73}$$

Thus, one can see that

$$\kappa(\boldsymbol{\Sigma}_t) \leq \kappa_t^{\text{up}}. \tag{74}$$

Lemma 4 (Weighted Harmonic-Mean Eigenvalue Bound). *Let $A_{t,c} \in \mathbb{R}^{d\times d}$, $c = 1, \ldots, C$, be symmetric positive definite matrices, and let $\pi_c > 0$ be positive weights. Then*

$$\lambda_{\min}\left(\sum_{c=1}^{C} \pi_c A_{t,c}\right) \;\geq\; \left(\sum_{c=1}^{C} \frac{\pi_c}{\lambda_{\min}(A_{t,c})}\right)^{-1}. \tag{75}$$

Proof. The proof proceeds in two steps.

First, for each c, since $A_{t,c}$ is symmetric positive definite, we have $A_{t,c} \succeq \lambda_{\min}(A_{t,c})\, I$. Multiplying by $\pi_c > 0$ and summing over $c = 1, \ldots, C$ preserves the PSD order, giving

$$\sum_{c=1}^{C} \pi_c A_{t,c} \;\succeq\; \left(\sum_{c=1}^{C} \pi_c \, \lambda_{\min}(A_{t,c}) \right) I.$$

Since the minimum eigenvalue is monotone with respect to the PSD order, it follows that

$$\lambda_{\min}\left(\sum_{c=1}^{C} \pi_c A_{t,c} \right) \;\geq\; \sum_{c=1}^{C} \pi_c \, \lambda_{\min}(A_{t,c}). \tag{76}$$

Then, let $\mu_c := \lambda_{\min}(A_{t,c}) > 0$. Applying the weighted AM–HM inequality to the positive scalars $\mu_1, \ldots, \mu_C$ with weights $\pi_c > 0$ yields

$$\sum_{c=1}^{C} \pi_c \, \mu_c \;\geq\; \left(\sum_{c=1}^{C} \frac{\pi_c}{\mu_c} \right)^{-1}. \tag{77}$$

Finally, we combine (76) and (77) to get the stated inequality:

$$\lambda_{\min}\left(\sum_{c=1}^{C} \pi_c A_{t,c} \right) \;\geq\; \sum_{c=1}^{C} \pi_c \, \lambda_{\min}(A_{t,c}) \;\geq\; \left(\sum_{c=1}^{C} \frac{\pi_c}{\lambda_{\min}(A_{t,c})} \right)^{-1}.$$

B.4 Proof of Lemma 2

Proof. (a) We first study the case of MoG→MoG.

One can compute that

$$\mathcal{J}(q_{\text{mwac}}) = \mathbb{E}[\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|^2] = (1-r)^2 \|\boldsymbol{\mu}\|^2 + d(\rho - \rho')^2. \tag{78}$$

Note that the covariance matrices of the distribution of q_1 and q_0 are

$$\boldsymbol{\Sigma}_1 = \rho^2 \boldsymbol{I} + \boldsymbol{\mu}\boldsymbol{\mu}^\top, \quad \boldsymbol{\Sigma}_0 = \rho'^2 \boldsymbol{I} + r^2 \boldsymbol{\mu}\boldsymbol{\mu}^\top, \tag{79}$$

respectively. Since that $\boldsymbol{\mu}' = r\boldsymbol{\mu}$, $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_0$ share the same eigenvectors. In the direction of $\boldsymbol{\mu}$, their eigenvalues are $\rho^2 + \|\boldsymbol{\mu}\|^2$ and $\rho'^2 + r^2\|\boldsymbol{\mu}\|^2$, respectively. In the direction of the orthogonal subspace of $\boldsymbol{\mu}$, the eigenvalues are ρ^2 and ρ'^2, respectively. By the covariance lower bound for the squared Wasserstein-2 distance, we have

$$\mathcal{W}_2^2(q_0, q_1) \geq (\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - \sqrt{\rho'^2 + r^2\|\boldsymbol{\mu}\|^2})^2 + (d-1)(\rho - \rho')^2. \tag{80}$$

Since that $\|\boldsymbol{\mu}\| \gtrsim \epsilon^{-1}$, we have

$$(\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - \sqrt{\rho'^2 + r^2\|\boldsymbol{\mu}\|^2})^2 \geq (1 - O(\epsilon))(1-r)^2\|\boldsymbol{\mu}\|^2. \tag{81}$$

Therefore, by combining (78) and $\rho, \rho' \leq C$, we can obtain

$$\mathcal{W}_2^2(q_0, q_1) \geq (1 - O(\epsilon))\mathcal{J}(q_{\text{mwac}}), \tag{82}$$

which is equivalent to

$$\mathcal{J}(q_{\text{mwac}}) \leq (1 + O(\epsilon))\mathcal{W}_2^2(q_0, q_1). \tag{83}$$

(b) We then study the case of MoG→SG.
One can obtain that

$$\mathcal{J}(q_{\text{mwac}}) = \mathbb{E}[\|\boldsymbol{x}_1 - \boldsymbol{x}_0\|^2] = \|\boldsymbol{\mu}\|^2 + d(\rho - 1)^2. \tag{84}$$

The covariance lower bound now gives

$$\mathcal{W}_2^2(q_0, q_1) \geq (\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - 1)^2 + (d - 1)(\rho - 1)^2. \tag{85}$$

One can observe that

$$\begin{aligned} &\mathcal{J}(q_{\text{mwac}}) - (\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - 1)^2 + (d - 1)(\rho - 1)^2 \\ =&2(\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - \rho) \leq 2\|\boldsymbol{\mu}\|. \end{aligned} \tag{86}$$

Since that $\|\boldsymbol{\mu}\| \gtrsim \epsilon^{-1}$, we can obtain that

$$\|\boldsymbol{\mu}\| \lesssim \epsilon\|\boldsymbol{\mu}\|^2. \tag{87}$$

Therefore,

$$\begin{aligned} \mathcal{J}(q_{\text{mwac}}) \leq&\mathcal{W}_2^2(q_0, q_1) + \mathcal{J}(q_{\text{mwac}}) - (\sqrt{\rho^2 + \|\boldsymbol{\mu}\|^2} - 1)^2 + (d - 1)(\rho - 1)^2 \\ \leq&(1 + O(\epsilon))\mathcal{W}_2^2(q_0, q_1). \end{aligned} \tag{88}$$

B.5 Proof of Theorem 2

Proof. For

$$\boldsymbol{x}_1 \sim \frac{1}{2}\mathcal{N}(\boldsymbol{\mu}, \rho^2\boldsymbol{I}) + \frac{1}{2}\mathcal{N}(-\boldsymbol{\mu}, \rho^2\boldsymbol{I}), \tag{89}$$

and

$$\boldsymbol{x}_0 \sim \frac{1}{2}\mathcal{N}(\boldsymbol{\mu}', \rho'^2\boldsymbol{I}) + \frac{1}{2}\mathcal{N}(-\boldsymbol{\mu}', \rho'^2\boldsymbol{I}), \tag{90}$$

we have $\text{Cov}(\boldsymbol{x}_1) = \rho'^2\boldsymbol{I} + \boldsymbol{\mu}'\boldsymbol{\mu}'^\top$. Then, we can derive

$$\begin{aligned} &\boldsymbol{\Sigma}_t \\ =&\frac{1}{2}(\boldsymbol{M}_{t,1}\boldsymbol{\Sigma}_0\boldsymbol{M}_{t,1}^\top + \boldsymbol{M}_{t,2}\boldsymbol{\Sigma}_0\boldsymbol{M}_{t,2}^\top) + ((1 - t)\boldsymbol{\mu}' + t\boldsymbol{\mu})((1 - t)\boldsymbol{\mu}' + t\boldsymbol{\mu})^\top \\ =&(1 + t(\frac{\rho}{\rho'} - 1))^2 \cdot \rho'^2\boldsymbol{I} + (1 + (1 - t)(r - 1))^2\boldsymbol{\mu}\boldsymbol{\mu}^\top, \end{aligned} \tag{91}$$

and

$$\bar{\boldsymbol{\Sigma}}_t := \mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t] = \frac{\rho^2+\rho\rho'+\rho'^2}{3}\boldsymbol{I} + \frac{r^2+r+1}{3}\boldsymbol{\mu}\boldsymbol{\mu}^\top \tag{92}$$

Note that the Hessian of the flow matching loss $\mathbb{E}_{t\in U[0,1]}[\mathcal{L}_t(\boldsymbol{A}_t, \boldsymbol{b}_t)]$ with $\mathcal{L}_t(\boldsymbol{A}_t, \boldsymbol{b}_t)$ defined in (7) is

$$\boldsymbol{H} = 2\mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t \otimes \boldsymbol{I}]. \tag{93}$$

Then, the condition number is computed as

$$\begin{aligned}
&\bar{\kappa}_{(\text{MoG}\to\text{MoG})}\\
=&\frac{\lambda_{\max}\mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t]}{\lambda_{\min}\mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t]}\\
=&\frac{\mathbb{E}_{t\sim U[0,1]}[(1+t(\frac{\rho'}{\rho}-1))^2]\rho'^2 + \mathbb{E}_{t\sim U[0,1]}[(1+(1-t)(r-1))^2]\|\boldsymbol{\mu}\|^2}{\mathbb{E}_{t\sim U[0,1]}[(1+(1-t)(\frac{\rho}{\rho'}-1))^2]\rho'^2}\\
=&1+\frac{r^2+r+1}{\rho^2+\rho\rho'+\rho'^2}\|\boldsymbol{\mu}\|^2.
\end{aligned} \tag{94}$$

At the initialization when $\boldsymbol{A}_t = 0$, $\boldsymbol{b}_t = 0$ for any $t \in [0,1]$, we have

$$\begin{aligned}
\bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{MoG})} = \mathbb{E}_{t\sim U[0,1]}[\mathcal{L}^{(0)}_t(\boldsymbol{A}_t, \boldsymbol{b}_t)] =& \mathbb{E}_{t\sim U[0,1],\boldsymbol{x}_0,\boldsymbol{x}_1}[\|\boldsymbol{x}_1-\boldsymbol{x}_0\|^2]\\
=&(1-r)^2\|\boldsymbol{\mu}\|^2 + (\rho'-\rho)^2 d.
\end{aligned} \tag{95}$$

For $\boldsymbol{x}_0 \sim \mathcal{N}(0, \boldsymbol{I})$ that is computed by $(\boldsymbol{x}_1 - s\cdot\boldsymbol{\mu})/\rho$ given the latent variable $z = s$, we have

$$\begin{aligned}
\boldsymbol{\Sigma}_t =&\mathbb{E}[(\boldsymbol{x}_t - \mathbb{E}[\boldsymbol{x}_t])(\boldsymbol{x}_t - \mathbb{E}[\boldsymbol{x}_t])^\top]\\
=&\mathbb{E}[((1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1)((1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1)^\top]\\
=&t^2(\rho^2\boldsymbol{I} + \boldsymbol{\mu}\boldsymbol{\mu}^\top) + (1-t)^2\boldsymbol{I} + 2t(1-t)\rho\boldsymbol{I}.
\end{aligned} \tag{96}$$

Then, the condition number is computed as

$$\begin{aligned}
\bar{\kappa}_{(\text{MoG}\to\text{SG})} =&\frac{\lambda_{\max}\mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t]}{\lambda_{\min}\mathbb{E}_{t\sim U[0,1]}[\boldsymbol{\Sigma}_t]}\\
=&\frac{\frac{1}{3}(\rho^2+\rho+1+\|\boldsymbol{\mu}\|^2)}{\frac{1}{3}\cdot(\rho^2+\rho+1)}\\
=&1+\frac{1}{\rho^2+\rho+1}\|\boldsymbol{\mu}\|^2.
\end{aligned} \tag{97}$$

At the initialization, we have

$$\begin{aligned}
\bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{SG})} =&\mathbb{E}_{t\sim U[0,1]}[\mathcal{L}^{(0)}_t)(\boldsymbol{A}_t, \boldsymbol{b}_t)] = \quad \mathbb{E}_{t\sim U[0,1],\boldsymbol{x}_0,\boldsymbol{x}_1}[\|\boldsymbol{x}_1-\boldsymbol{x}_0\|^2]\\
=&\|\boldsymbol{\mu}\|^2 + (\rho-1)^2 d.
\end{aligned} \tag{98}$$

Therefore, we need the following condition

$$\begin{cases}
\bar{\kappa}_{(\text{MoG}\to\text{MoG})} \le \bar{\kappa}_{(\text{MoG}\to\text{SG})},\\
\bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{MoG})} \le \bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{SG})},
\end{cases} \tag{99}$$

which holds if

$$\begin{cases} 0 \le r \le \min\{\frac{1}{2}, \frac{4\rho(\rho-1)}{\rho^2+\rho+1}\}, \\ \rho > 1, \\ \rho' \in (\rho, 2\rho-1). \end{cases} \tag{100}$$

This is because $\bar{\kappa}_{(\text{MoG}\to\text{MoG})} \le \bar{\kappa}_{(\text{MoG}\to\text{SG})}$ holds if $\rho, \rho' > 1$ and $0 \le r \le \min\{\frac{1}{2}, \frac{4\rho(\rho-1)}{\rho^2+\rho+1}\}$. $\bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{MoG})} \le \bar{\mathcal{L}}^{(0)}_{(\text{MoG}\to\text{SG})}$ also indicates $\rho' < 2\rho - 1$.

C Experiments

C.1 Additional Simulation of Theorem 2

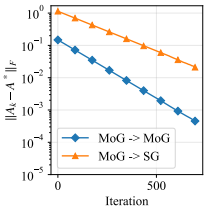
Theorem 2 gives parameter conditions under which MoG→MoG achieves a smaller convergence rate and FM loss than MoG→SG. Below Theorem 2 we provide an example that satisfies these conditions. Here, we conduct a simulation to validate that this example satisfies Theorem 2 and to visualize the convergence rates. Figure 4 (a) and (b) present the update trends of the FM loss and $\mathbb{E}_{t\sim\text{Unif}[0,1]}\|\boldsymbol{A}^k - \boldsymbol{A}^*\|_F$ under the scenarios of MoG→MoG and MoG→SG, respectively. The results show that MoG→MoG has both a smaller initial loss and a smaller converged loss. $\bar{\kappa}_{(\text{MoG}\to\text{MoG})}$ is computed as 1.112 and $\bar{\kappa}_{(\text{MoG}\to\text{SG})} = 1.167$. Therefore, the condition in ((21)) of Theorem 2 holds in this example.

C.2 Additional Ablation Study on Realistic Datasets

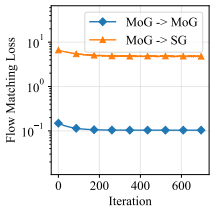
EM vs k-means. In §3.2, we argued that k-means is preferable to EM for learning the MoG parameters because of its lower per-iteration cost. Here, we first explain why this may lead to better estimates, then provide empirical evidence and implementation details. Both EM and k-means solve non-convex optimization problems and are thus prone to local optima [19,35,37]. A standard remedy is to run multiple initializations and keep the solution with the best objective value, and in general the more initializations and iterations per initialization one can afford, the better the resulting estimate. Since k-means has low per-iteration cost, we can run it for more initializations and more iterations within each initialization under the same time budget. Empirically, as seen in Figure 4 (d), for estimating an MoG with $C = 10$ on CIFAR-10, EM is about 165× slower than k-means. Both algorithms are run on the same machine, with the maximum number of iterations set to 50 and the number of initializations set to 1. This gap reflects not only the algorithmic complexity above, but also the fact that efficient implementations of k-means on GPU are widely available in standard libraries[3], while EM in scikit-learn (also used by the work of [23]) is CPU-based and not as efficient.

Choice of C. Here we discuss how to choose the number of components C used in the Gaussian mixture when it is not known a priori. We perform a

[3] See the faiss-gpu https://github.com/facebookresearch/faiss and cuML libraries https://docs.rapids.ai/api/cuml/stable/; this paper uses the former.



(a) Iterate Error



(b) CFM Loss

C	FID at Init.
1	169.1
10	**165.1**
20	227.4
40	225.6

(c) Choice of C

Method	Time (s)
k-means	20
EM	3301

(d) MoG Estimator

Fig. 4: (a)–(b) Additional simulation of the companion example of Theorem 2 in §4.4. (c)–(d) Additional ablation studies on the design choices of latent MoG in §3.

simple grid search: for each $C \in \{1, 10, 20, 40\}$, we run k-means clustering on the training data to obtain the cluster assignments, estimate the per-component means and covariances, and evaluate the FID of samples drawn from the resulting regularized MoG. This FID does not require training the network, and the search over the grid can be run in parallel. As shown in Figure 4 (c), on CIFAR-10, $C = 10$ offers the best FID at initialization, while too large or too small values appear to hurt performance. Thanks to the fast GPU-based k-means, the entire search takes about 10 minutes in total, roughly 1% of the FM training time; thus, selecting C adds negligible overhead to the overall pipeline.

C.3 Details on Synthetic Experiments

We train a small MLP with 4 hidden layers of 128 units each and Swish activations (∼50K parameters), taking as input the concatenation of the 2D spatial coordinate and the scalar time t, and outputting the 2D velocity.

C.4 Details on CIFAR-10 and Imagenet Experiments

For the MoG latent, we run k-means clustering with $C = 10$ components on the training data before training begins; this preprocessing step takes less than 1% of the total training time. The per-component means and covariances are estimated from the cluster assignments, and the eigenvalues of each covariance are regularized with threshold $\tau = 1$ (cf. §3.2 and Table 1). The MoG parameters are fixed throughout training. All models are trained with the Adam optimizer using a learning rate of 2×10^{-4}, batch size of 512, gradient clipping at norm 1.0, and 5,000 warmup steps. We train for 100K iterations on CIFAR-10 and 200K iterations on ImageNet (32×32).

D Notations

For convenience, we summarize the notations used in the paper in Tables 3 and 4.

Table 3: General Notations for the paper.

Notation	Description
d	dimension of the data space
t	time variable of the flow, $t \in [0,1]$
k, K	gradient descent iteration index / total iterations
σ	noise standard deviation in the conditional path
τ	eigenvalue regularization threshold
B	mini-batch size
$\boldsymbol{x}_0$, $\boldsymbol{x}_1$	data / latent sample
$\boldsymbol{x}_t$	interpolated point, defined in (5)
$\boldsymbol{y}$	velocity conditioned on $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ as defined in (5)
p_0, p_1	data / latent marginal distributions
p_t	marginal distribution at time t
$p_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1)$	conditional probability path defined in (2)
$q(\boldsymbol{x}_0, \boldsymbol{x}_1)$	coupling distribution over data–latent pairs
$\Gamma(q_0, q_1)$	set of all couplings between q_0 and q_1
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
C	number of Gaussian components in the MoG
c, z	component index / variable in $\{1, \dots, C\}$
π_c	mixing weight for component c (4)
$\boldsymbol{\mu}_0^c$, $\boldsymbol{\mu}_1^c$	mean of component c (data / latent) (4)
$\boldsymbol{\Sigma}_0^c$, $\boldsymbol{\Sigma}_1^c$	covariance of component c (data / latent) (4)
δ_t^c	interpolated mean offset, $(1-t)\,\delta_0^c + t\,\delta_1^c$
$u_t(\boldsymbol{x})$	ground-truth velocity field defined in (1)
$v_\theta(\boldsymbol{x}, t)$	velocity model with parameters introduced in θ (3)
ϕ_t	flow map induced by $u_t(\boldsymbol{x})$, defined in (1)
$(\phi_t)_\#$	pushforward operator of ϕ_t
$\mathcal{L}^{\mathrm{CFM}}(\theta)$	conditional flow matching loss defined in (3)
$\mathcal{J}(\gamma)$	transport cost for coupling γ
$\mathcal{W}_2^2$	squared Wasserstein-2 distance

Table 4: Notations for theoretical analysis (§4)

Notation	Description
$\boldsymbol{I}$	identity matrix
$\boldsymbol{A}_t$	affine velocity matrix at time t defined above (7)
$\boldsymbol{A}_t^*$	optimal affine velocity matrix at time t
$\mathcal{L}_t(\boldsymbol{A}_t)$	flow matching loss at time t defined in (7)
$\boldsymbol{B}_c$	OT transport map for component c
Δ_c	covariance perturbation, $\boldsymbol{B}_c - \boldsymbol{I}$
$\lambda_{\min}(\cdot)$, $\lambda_{\max}(\cdot)$	minimum / maximum eigenvalue
$\kappa(\cdot)$	condition number, $\lambda_{\max}/\lambda_{\min}$
κ_t^{up}	upper bound on the condition number at time t (15)
ρ_t	GD contraction factor at time t
η_t, η_t^*	step size in (8) and the optimal one in Lemma 1