MC-MF: Multimodal Conditional MeanFlow with Distribution Alignment

Anonymous Author(s)
Affiliation
Address
email

Abstract

Conditional MeanFlow [Geng et al., 2026a] enables efficient one-step generation by learning an average velocity field. However, its regression objective does not explicitly measure the discrepancy between the conditional distribution induced by the one-step map and the target distribution. This limitation can be particularly important in multimodal conditional settings, where it may recover plausible modal regions, yet the probability mass is incorrectly allocated across modes. To address this limitation, we introduce MC-MF: Multimodal Conditional MeanFlow, which augments the standard MeanFlow objective with a condition-aware maximum mean discrepancy (MMD) regularizer between generated and target conditional distributions. This sample-based regularizer requires no knowledge of mode identities, the number of modes, or target mixture proportions, while preserving single function evaluation inference. Our theoretical analysis shows that the proposed MMD decomposes into distribution discrepancies within each condition, and captures discrepancies arising from incorrect probability allocation across modes. We also experimented on a controlled conditional Gaussian mixture and an asymmetric event-conditioned Lorenz system, showing that MC-MF can substantially improve multimodal probability allocation over one-step MeanFlow.

1 Introduction

Diffusion and flow-based models have emerged as powerful frameworks for generative modeling, offering flexible approaches for learning complex data distributions. MeanFlow [Geng et al., 2026a] introduced an average-velocity formulation that enables high-quality generation in a single sampling step, substantially reducing inference cost while maintaining strong generative performance. However, in many real-world applications, generation must satisfy additional requirements specified through conditioning [Ho and Salimans, 2022]. Examples include generating diverse outcomes for a given class or attribute, predicting multiple plausible responses to the same biological perturbation [Bunne et al., 2023, Yu et al., 2025, Luo et al., 2024], representing alternative molecular conformations [Cao et al., 2025, Zhou et al., 2025], or modeling multiple possible trajectories of a dynamical system [Chen et al., 2024, Price et al., 2025]. In these settings, the conditional distribution $p(x \mid c)$ may itself be highly multimodal, and a successful conditional generator must not only recover the individual modes but also reproduce the relative probability mass assigned to each of them.

MeanFlow and its improved variant [Geng et al., 2026b] support conditional generation, but their training objectives are primarily based on regression of an average velocity field. In practice, accurate field regression does not necessarily guarantee that the resulting one-step generated distribution accurately matches the target conditional distribution. In particular, we observe that when the target contains separated modes with unequal probabilities, conditional MeanFlow can recover plausible modal regions while substantially distorting the probability allocated across them, as illustrated by the conditional Gaussian-mixture experiment in Section 5.

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

To address this limitation, we introduce **MC-MF: Multimodal Conditional MeanFlow**, which augments the standard MeanFlow objective with an explicit distribution-level regularizer between the generated and target conditional distributions. Specifically, we use conditional maximum mean discrepancy (MMD) [Gretton et al., 2012] to encourage agreement between $p_{\text{data}}(x \mid c)$ and the one-step generated distribution $p_\theta(x \mid c)$. Unlike mode-specific objectives, the proposed regularizer does not require mode identities, component locations, the number of modes, or target mixture probabilities during training. Instead, it operates directly on samples, providing conditional distribution supervision that complements the transport-field regression of MeanFlow.

This simple yet effective choice has several practical advantages. First, our method can be applied even when the modal structure is unknown. MMD is sample-based and differentiable that does not require explicit density estimation. Second, our approach does not modify the one-step sampling procedure of MeanFlow. The sampling procedure still requires only a single function evaluation at inference time. Our theoretical analysis in Section 3.2 shows that the proposed MMD objective matches the generated and target distributions separately by condition, rather than only matching the aggregate distribution across all conditions. This condition-wise matching is especially useful in multimodal settings, where different conditions may have different mode structures or mode probabilities that would be masked by just aggregate distribution matching.

2 Background

Flow Matching Flow Matching (FM) learns a time-dependent velocity field that transports samples from a simple prior distribution to the data distribution. Given an interpolation path z_t, the model is trained with $\mathcal{L}_{\text{CFM}} = \mathbb{E}\left[\|v_\theta(z_t, t) - v_t\|_2^2\right]$, and generation typically requires numerical integration of the learned ODE over multiple function evaluations.

Conditional MeanFlow MeanFlow enables one-step generation by learning an average velocity field over a time interval rather than an instantaneous velocity. For $r < t$, the average velocity is defined as

$$u(z_t, r, t) = \frac{1}{t-r}\int_r^t v(z_\tau, \tau)\, d\tau. \tag{1}$$

A neural network u_θ is trained using the MeanFlow regression loss. For conditional generation,u_θ also takes the conditioning variable c,

$$\mathcal{L}_{\text{MF}} = \mathbb{E}\left[\|u_\theta(z_t, r, t, c) - \text{sg}(u_{\text{tgt}})\|_2^2\right], \tag{2}$$

where u_{tgt} is the MeanFlow target derived from the instantaneous velocity field and $\text{sg}(\cdot)$ denotes stop-gradient. Then one-step conditional generation is performed using $\hat{X} = Z + u_\theta(Z, 0, 1, c), Z \sim P_Z$.

3 Methodology

In this section, we introduce MC-MF and provide theoretical motivation for the proposed MMD-based distribution regularizer. We show that our conditional MMD regularizer evaluates distribution matching separately within each condition and is particularly suited to multimodal settings.

Problem setup: Let $C \in \mathcal{C}$ denote a conditioning variable and let $X \sim P_{\text{data}}(X \mid C)$. For theoretical analysis, we represent a multimodal conditional distribution as

$$P_{\text{data}}(X \mid C = c) = \sum_{m=1}^{M_c} \alpha_{c,m} P_{c,m}, \tag{3}$$

where $\alpha_{c,m} \geq 0$ and $\sum_{m=1}^{M_c} \alpha_{c,m} = 1$. Here, $P_{c,m}$ denotes the m-th modal component under condition c, and $\alpha_{c,m}$ is the associated probability mass.

We consider a one-step conditional generator $\hat{X} = G_\theta(Z, C), Z \sim P_Z$, with the goal to learn $P_\theta(\hat{X} \mid C = c) \approx P_{\text{data}}(X \mid C = c), \forall c \in \mathcal{C}$. In the multimodal setting, this requires recovering not only the individual modal distributions, but also the correct relative probability mass assigned to each mode.

3.1 MC-MF: Multimodal Conditional MeanFlow

We introduce **MC-MF: Multimodal Conditional MeanFlow** by augmenting the standard conditional MeanFlow objective with an explicit distribution-level regularizer,

$$\mathcal{L}_{\text{MC-MF}} = \mathcal{L}_{\text{MF}} + \lambda \mathcal{L}_{\text{dist}}, \tag{4}$$

where $\mathcal{L}_{\text{MF}}$ is defined in Eq.2, λ is the regularization coefficient, and $\mathcal{L}_{\text{dist}}$ is defined as a conditional maximum mean discrepancy (MMD):

$$\mathcal{L}_{\text{dist}} = \text{MMD}^2 \left(P_{\text{data}}(C, X), P_\theta(C, \hat{X}) \right). \tag{5}$$

This formulation directly compares the joint distribution of the condition–sample pairs $(C, \hat{X})$ between generated one-step distribution and the target distribution. This provides a conditional distribution-level supervision in addition to the average-velocity regression of MeanFlow.

For the MMD-based kernel, we decompose the kernel into a product kernel $k_C(c, c')$ and a multi-scale radial basis function kernel $k_X(x, x')$ as the following:

$$k\big((c, x), (c', x')\big) = k_C(c, c')\, k_X(x, x'). \tag{6}$$

For discrete conditions, we define a product kernel over the condition–sample pairs to ensure that the distribution matching is performed within each condition:

$$k_C(c, c') = \frac{\mathbf{1}[c = c']}{p(c)}, \tag{7}$$

The indicator acts as a gating function: samples from different conditions have zero kernel similarity, so the target and generated distributions are compared only within the same condition. The factor $1/p(c)$ normalizes for the probability of condition c, so that each condition contributes with weight $p(c)$ to the resulting MMD objective.

For the data variable $k_X(x, x')$, we use a multi-scale radial basis function (RBF) kernel, defined as below:

$$k_X(x, x') = \frac{1}{L} \sum_{\ell=1}^{L} \exp\left(-\frac{\|x - x'\|_2^2}{2\sigma_\ell^2} \right), \tag{8}$$

The bandwidths are selected around a reference scale σ, where $\sigma_\ell \in \left\{ \frac{\sigma}{4}, \frac{\sigma}{2}, \sigma, 2\sigma \right\}$ is determined by the median-distance heuristic. In practice, we estimate $\mathcal{L}_{\text{dist}}$ using the biased empirical MMD estimator with the multi-scale RBF kernel.

During training, data samples X and the generated samples $\hat{X}$ are compared under the same conditioning variable C. Importantly, since k_C only determines which conditional distributions are compared, the modal structure itself is unknown to the model. The data kernel k_X measures discrepancies within each condition and is therefore sensitive to differences in mode location, shape, spread, and relative probability mass. Thus, our regularizer does not require access to mode identities, the number of modes, mode locations, or target mixture probabilities.

3.2 Theoretical results

Although the proposed MMD regularizer is simple, we provide a theoretical interpretation of why it is well suited to multimodal conditional generation. In particular, we show that the condition-aware MMD objective matches distributions separately within each condition and is explicitly sensitive to errors in the probability mass allocated across modes, without requiring mode identities or target mixture proportions during training.

For distributions P and Q, the maximum mean discrepancy (MMD) under kernel k is

$$\text{MMD}_k^2(P, Q) = \|\mu_P - \mu_Q\|_{\mathcal{H}}^2, \tag{9}$$

where μ_P and μ_Q are the corresponding kernel mean embeddings in the RKHS $\mathcal{H}$. For a characteristic kernel, $\text{MMD}_k(P, Q) = 0$ if and only if $P = Q$.

Lemma 3.1 (Conditional decomposition)**.** *Let $P_c = P_{\text{data}}(X \mid C = c)$ and $Q_c = P_\theta(X \mid C = c)$, with common condition marginal $p(c) > 0$. Under the kernel defined in Eq. 6, we have the following:*

$$\mathrm{MMD}_k^2 = \sum_c p(c)\,\mathrm{MMD}_{k_X}^2(P_c, Q_c). \tag{10}$$

The lemma shows that the proposed joint MMD is equivalent to a weighted sum of distribution discrepancies evaluated separately within each condition. Thus, errors under one condition cannot be compensated for by samples generated under another condition.

Proposition 3.2 (Sensitivity to multimodal probability allocation)**.** *For each condition c, suppose the target and generated conditional distributions share the same modal components but assign them different probabilities,*

$$P_c = \sum_{m=1}^{M_c} \alpha_{c,m} P_{c,m}, \qquad Q_c = \sum_{m=1}^{M_c} \beta_{c,m} P_{c,m}. \tag{11}$$

Let $\mu_{c,m}$ denote the RKHS mean embedding of $P_{c,m}$ and define $(G_c)_{ij} = \langle \mu_{c,i}, \mu_{c,j} \rangle_{\mathcal{H}}$. Then we can reformulate the MMD as follows: (see Appendix B for proof)

$$\mathrm{MMD}_k^2 = \sum_c p(c)(\alpha_c - \beta_c)^\top G_c(\alpha_c - \beta_c). \tag{12}$$

Here, $\alpha_c - \beta_c$ represents the mode probability mass error under condition c. The matrix G_c captures how distinct the modal distributions are under the kernel through the inner products of their RKHS mean embeddings. Thus, the quadratic form $(\alpha_c - \beta_c)^\top G_c(\alpha_c - \beta_c)$ measures the distributional discrepancy induced by incorrect probability allocation across modes. Importantly, this sensitivity arises without requiring mode labels, the number of modes, or target mixture probabilities during training.

Remark The shared-component assumption in Proposition 3.2 is introduced only to isolate the effect of mode-probability mismatch. In the more general case, the generated modes may also differ from the target in location, shape, or spread. MMD captures these discrepancies jointly by comparing the full conditional distributions, making it a distribution-level complement to the MeanFlow objective. Importantly, the mixture weights appearing in Proposition 3.2 are not inputs to MC-MF. They arise only in the theoretical decomposition of the population distribution. In practice, the relative prevalence of different modes is implicitly encoded in the observed samples, and MMD is estimated directly from real and generated samples without explicitly identifying the modes or their probabilities.

4 Related Work

Multimodal conditional and one-step generation. Conditional generative modeling aims to learn a complex distribution $P(X \mid C)$, where a single condition may correspond to multiple distinct modes of valid outcomes. Accurately modeling such a multimodal distribution requires learning both the structure of these modes and their relative probability mass as described in Eq. 3. Flow-based generative models have substantially advanced distribution learning [Albergo and Vanden-Eijnden, 2022, Lipman et al., 2022, Liu et al., 2022, Pao-Huang et al., 2026, Pooladian et al., 2023], as well as methods for conditional or one-step generations [Frans et al., 2025, Song et al., 2023, Boffi et al., 2024, Fu et al., 2025]. More recently, several methods address the intersection of fast generation in the conditional multimodal settings. SubFlow [Lin et al., 2026] addresses the one-step multimodal conditional generation using flow matching. It explicitly clusters each class into semantic sub-modes and conditions the generator on an additional sub-mode variable S, resulting in a model of the form $P(X \mid C, S)$. In contrast, MC-MF does not require explicit sub-mode identification, additional sub-mode labels, or known mode proportions. RMFlow [Huang et al., 2026] studies multimodal failure in 1-NFE MeanFlow as well by introducing a noise-injection refinement step with likelihood-based training. MM-FM [Luo et al., 2026] models the multimodality through a fitted Gaussian mixture source distribution and mode-dependent coupling in the latent space. In contrast, our approach does not change the source distribution, but directly regularizes the conditional distribution produced by the one-step MeanFlow map.

Table 1: Five-seed results on the conditional three-mode Gaussian mixture. MC-MF substantially improves mode-probability allocation while retaining one-step generation. Note that bridge mass is defined as the fraction of generated samples that are farther than a fixed radius $r = 1.5$ from every target Gaussian mode center.

Method	NFE	Mode-mass TV ↓	Low-density/bridge mass ↓
Conditional FM	50	0.0572 ± 0.0186	$\mathbf{0.1048 \pm 0.0241}$
MeanFlow	1	0.2704 ± 0.0251	0.1923 ± 0.0101
MC-MF(ours)	1	$\mathbf{0.0192 \pm 0.0023}$	0.1568 ± 0.0135

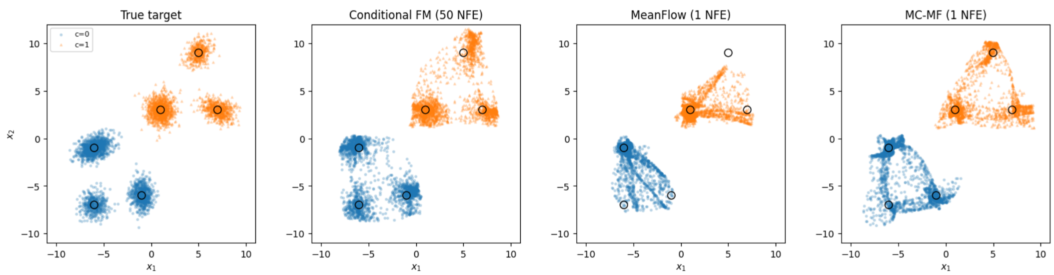

Figure 1: Conditional three-mode Gaussian mixture. Conditional one-step MeanFlow substantially distorts the probability allocated across modes, whereas MC-MF recovers the target mode proportions much more accurately while retaining 1 NFE. CFM-50 provides stronger local geometric fidelity.

Distribution matching with MMD. Maximum mean discrepancy (MMD) has been widely used in generative modeling. Generative Moment Matching Networks (GMMNs) [Li et al., 2015] directly train a generator by minimizing MMD between real and generated samples, while Conditional Generative Moment Matching Networks [Ren et al., 2016] extend moment matching to conditional generation. MMD-GAN [Li et al., 2017] further improves MMD-based generative modeling through learned feature representations. MMD has also recently been used to construct generative transport dynamics and conditional sampling procedures [Hagemann et al., 2024, Galashov et al., 2025]. Unlike these approaches, where MMD serves as a primary generative objective, MC-MF uses MMD as a distribution-level regularizer that complements the MeanFlow regression loss.

Recently, scDFM [Yu et al., 2026] combined conditional Flow Matching with an MMD regularizer to improve population-level fidelity in single-cell perturbation prediction. In comparison, our approach differs in two important respects. First, MC-MF focuses on one-step MeanFlow and multimodal conditional generation with particular emphasis on preserving relative probability mass across modes. Second, we construct a condition-aware discrepancy over condition–sample pairs (C, X), so that the generated and target distributions are matched separately within each condition. This design requires neither explicit mode identities nor known mixture proportions and provides distribution-level supervision while retaining one-function-evaluation inference.

5 Experiments

2D Conditional Gaussian mixture. First, we consider a controlled multimodal conditional 2D Gaussian mixture example with two classes in which each condition has three separated modes with different mixture weights. Specifically, both conditional distributions use component probabilities $0.5/0.3/0.2$. For example, the target conditional Gaussian mixture for each condition c is $P_{\text{data}}(X \mid C = c) = 0.5\, P_{c,1} + 0.3\, P_{c,2} + 0.2\, P_{c,3}$, where $P_{c,1}$, $P_{c,2}$, and $P_{c,3}$ denote the three Gaussian modes. In this example, we isolate the problem of mode-probability allocation. The goal is to validate the failure case of one-step conditional MeanFlow for unequal multimodal probability mass and show that using our proposed approach with MMD regularizer can reduce errors in this allocation. We compare with a 50-NFE conditional Flow Matching (CFM) [Lipman et al., 2022] and one-step conditional MeanFlow [Geng et al., 2026a].

Table 2: Results for asymmetric event-conditioned Lorenz system with three-seeds. MC-MF substantially improves event recovery and multimodal probability allocation over one-step MeanFlow.

Method	**NFE**	**Event success** $\uparrow$	**Joint TV** $\downarrow$
MeanFlow	1	0.6249 ± 0.0215	0.4101 ± 0.0176
MC-MF(ours)	1	$\mathbf{0.9945 \pm 0.0013}$	0.0718 ± 0.0102
Conditional FM	50	0.9855 ± 0.0018	$\mathbf{0.0198 \pm 0.0007}$

In Table 1, we evaluate conditional mode-probability error using total-variation (TV) distance. One-step conditional MeanFlow shows substantial mode-mass distortion, whereas MC-MF reduces the TV error from 0.2704 to 0.0192, corresponding to a 92.9% reduction. As shown in Figure 1, MeanFlow underrepresents some modes and concentrates excessive probability mass on others relative to the target proportions $0.5/0.3/0.2$. In contrast, MC-MF recovers the relative probability allocation substantially more accurately.

However, note that MC-MF does not fully recover the local geometry of the target mixture. Although the off/bridge mass improves compared with MeanFlow, the CFM-50 baseline remains closer to the local geometry than ours. This suggests that while the MMD regularizer corrects global multimodal probability allocation well, the iterative transport method with more NFE steps remains more effective at resolving fine-grained local geometry.

Asymmetric Lorenz dynamics. In this example, we evaluate our approach on a nonlinear dynamical system. We use an asymmetric variant of the Lorenz system by introducing a constant symmetry-breaking forcing $\varepsilon \in \{-6, +6\}$ in the x-dynamics, $\dot{x} = \sigma(y - x) + \varepsilon$. For each condition, we simulate 28,000 trajectories and filter them according to an event statistic that characterizes the oscillatory behavior of the x-trajectory. We retain trajectories whose event statistic exceeds a threshold of -0.6. Event-positive trajectories are then classified into left- and right-lobe modes according to the sign of their mean x-coordinate. The resulting left–right lobe proportions are approximately $75/25$ for $\varepsilon = -6$ and 30/70 for $\varepsilon = +6$.

In Table 2, conditional MeanFlow struggles to generate trajectories from the correct event-conditioned $E = 1$ Lorenz distribution with only 62.49% of success rate, whereas MC-MF achieves an event success rate of 99.45%, an improvement of approximately 37 percentage points. Beyond event validity, we evaluate on the joint event–lobe TV, which includes three probability categories: the left-event trajectories, the right-event trajectories, and the invalid/non-event trajectories. This metric simultaneously penalizes both failure selections of the event-conditioned distribution and incorrect allocation between its two left-right modes. MC-MF achieves a reduction of approximately 82.5% relative to the conditional MeanFlow. However, CFM-50 achieves a lower joint TV with 50 steps of NFE.

6 Conclusion

We introduced MC-MF, a simple yet effective approach for one-step multimodal conditional generation. Our condition-aware MMD regularizer complements the conditional MeanFlow regression objective and improves the preservation of relative probability mass across modes without requiring explicit mode labels, mode counts, or target mixture proportions. Our theoretical analysis and experiments demonstrate that MC-MF improves conditional distribution matching while retaining a single function evaluation inference. In future work, we plan to evaluate MC-MF on real-world scientific datasets, including single-cell perturbation data and other applications with complex multimodal conditional distributions.

References

Zhengyang Geng, Mingyang Deng, Xingjian Bai, Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *Advances in Neural Information Processing Systems*, 38:75460–75482, 2026a.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Charlotte Bunne, Stefan G Stark, Gabriele Gut, Jacobo Sarabia Del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature methods*, 20(11):1759–1768, 2023.

Hengshi Yu, Weizhou Qian, Yuxuan Song, and Joshua D Welch. Perturbnet predicts single-cell responses to unseen chemical and genetic perturbations. *Molecular systems biology*, 21(8):960–982, 2025.

Erpai Luo, Minsheng Hao, Lei Wei, and Xuegong Zhang. scdiffusion: conditional generation of high-quality single-cell data using diffusion model. *Bioinformatics*, 40(9):btae518, 2024.

Zhonglin Cao, Mario Geiger, Allan Dos Santos Costa, Danny Reidenbach, Karsten Kreis, Tomas Geffner, Franco Pellegrini, Guoqing Zhou, and Emine Kucukbenli. Efficient molecular conformer generation with so (3)-averaged flow matching and reflow. *arXiv preprint arXiv:2507.09785*, 2025.

Wenyin Zhou, Christopher Iliffe Sprague, Vsevolod Viliuga, Matteo Tadiello, Arne Elofsson, and Hossein Azizpour. Energy-based flow matching for generating 3d molecular structure. *arXiv preprint arXiv:2508.18949*, 2025.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, et al. Probabilistic weather forecasting with machine learning. *Nature*, 637(8044):84–90, 2025.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30467–30476, 2026b.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The journal of machine learning research*, 13:723–773, 2012.

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

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

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Peter Pao-Huang, Xiaojie Qiu, and Stefano Ermon. Generative modeling with flux matching. *arXiv preprint arXiv:2605.07319*, 2026.

Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky TQ Chen. Multisample flow matching: Straightening flows with minibatch couplings. *arXiv preprint arXiv:2304.14772*, 2023.

Kevin Frans, Danijar Hafner, Sergey Levine, and Pieter Abbeel. One step diffusion via shortcut models. In *International Conference on Learning Representations*, volume 2025, pages 34668–34684, 2025.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Nicholas M Boffi, Michael S Albergo, and Eric Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *arXiv preprint arXiv:2406.07507*, 2024.

Yuxiang Fu, Qi Yan, Lele Wang, Ke Li, and Renjie Liao. Moflow: One-step flow matching for human trajectory forecasting via implicit maximum likelihood estimation based distillation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17282–17293. IEEE, 2025.

Yexiong Lin, Jia Shi, Shanshan Ye, Wanyu Wang, Yu Yao, and Tongliang Liu. Subflow: Sub-mode conditioned flow matching for diverse one-step generation. *arXiv preprint arXiv:2604.12273*, 2026.

Yuhao Huang, Shih-Hsin Wang, Andrea Bertozzi, and Bao Wang. Rmflow: Refined mean flow by a noise-injection step for multimodal generation. In *International Conference on Learning Representations*, volume 2026, pages 76267–76286, 2026.

Gaoxiang Luo, Frank Cole, Sihang Zhang, Yuxiang Wan, Yulong Lu, and Ju Sun. Flow matching for multimodal distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23260–23271, 2026.

Yujia Li, Kevin Swersky, and Rich Zemel. Generative moment matching networks. In *International conference on machine learning*, pages 1718–1727. PMLR, 2015.

Yong Ren, Jun Zhu, Jialian Li, and Yucen Luo. Conditional generative moment-matching networks. *Advances in Neural Information Processing Systems*, 29, 2016.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network. *Advances in neural information processing systems*, 30, 2017.

Paul Hagemann, Johannes Hertrich, Fabian Altekrüger, Robert Beinert, Jannis Chemseddine, and Gabriele Steidl. Posterior sampling based on gradient flows of the mmd with negative distance kernel. In *International Conference on Learning Representations*, volume 2024, pages 37821–37853, 2024.

Alexandre Galashov, Valentin De Bortoli, and Arthur Gretton. Deep mmd gradient flow without adversarial training. In *International Conference on Learning Representations*, volume 2025, pages 45918–45952, 2025.

Chenglei Yu, Chuanrui Wang, Bangyan Liao, and Tailin Wu. scdfm: Distributional flow matching model for robust single-cell perturbation prediction. *arXiv preprint arXiv:2602.07103*, 2026.

A Appendix: Proof for lemma 3.1

Lemma 3.1 (Conditional decomposition) states that we have $P_c = P_{\text{data}}(X \mid C = c), Q_c = P_\theta(\hat{X} \mid C = c)$, and suppose the target and generated joint distributions share the same condition marginal $p(c) > 0$. We also defined the product kernel as

$$k\big((c,x),(c',x')\big) = \frac{\mathbf{1}[c = c']}{p(c)}\, k_X(x,x'), \tag{13}$$

Then, we have

$$\mathrm{MMD}_k^2(P,Q) = \sum_c p(c)\ \mathrm{MMD}_{k_X}^2(P_c,Q_c).$$

Proof. By the population definition of squared MMD, we have:

$$\begin{aligned}\mathrm{MMD}_k^2(P,Q) = \ &\mathbb{E}_{(C,X),(C',X')\sim P}[k((C,X),(C',X'))] \\ &+ \mathbb{E}_{(C,\hat{X}),(C',\hat{X}')\sim Q}[k((C,\hat{X}),(C',\hat{X}'))] \\ &- 2\mathbb{E}_{(C,X)\sim P,(C',\hat{X})\sim Q}[k((C,X),(C',\hat{X}))].\end{aligned} \tag{14}$$

For the first term, the product kernel k_c defined in Eq.13 is zero whenever the samples do not belong to the same class $c \neq c'$. Therefore we have,

$$\begin{aligned}&\mathbb{E}_{P,P}[k((C,X),(C',X'))] \\ &= \sum_c p(c)^2 \frac{1}{p(c)} \mathbb{E}_{X,X'\sim P_c}[k_X(X,X')] \\ &= \sum_c p(c)\mathbb{E}_{X,X'\sim P_c}[k_X(X,X')].\end{aligned} \tag{15}$$

Similarly, for the second term, we have:

$$\mathbb{E}_{Q,Q}[k((C,\hat{X}),(C',\hat{X}'))] = \sum_c p(c)\mathbb{E}_{\hat{X},\hat{X}'\sim Q_c}[k_X(\hat{X},\hat{X}')], \tag{16}$$

and for the third term, we have:

$$\mathbb{E}_{P,Q}[k((C,X),(C',\hat{X}))] = \sum_c p(c)\mathbb{E}_{X\sim P_c,\hat{X}\sim Q_c}[k_X(X,\hat{X})]. \tag{17}$$

Combining the three terms gives

$$\begin{aligned}\mathrm{MMD}_k^2(P,Q) &= \sum_c p(c)\Big(\mathbb{E}_{P_c,P_c}[k_X] + \mathbb{E}_{Q_c,Q_c}[k_X] - 2\mathbb{E}_{P_c,Q_c}[k_X]\Big) \\ &= \sum_c p(c)\ \mathrm{MMD}_{k_X}^2(P_c,Q_c),\end{aligned} \tag{18}$$

which proves the result. □

B Appendix: Proof for proposition 3.2

In Proposition 3.2, we state that we can reformulate the MMD regularizer as

$$\mathrm{MMD}_k^2 = \sum_c p(c)(\alpha_c - \beta_c)^\top G_c(\alpha_c - \beta_c). \tag{19}$$

Proof. First, given the conditional decomposition from Lemma 3.1, we have

$$\mathrm{MMD}_k^2 = \sum_c p(c)\ \mathrm{MMD}_{k_X}^2(P_c,Q_c). \tag{20}$$

For a fixed condition c, the target and generated distributions are

$$P_c = \sum_{m=1}^{M_c} \alpha_{c,m} P_{c,m}, \qquad Q_c = \sum_{m=1}^{M_c} \beta_{c,m} P_{c,m}. \tag{21}$$

Also by the linearity of kernel mean embeddings, we can write the following:

$$\mu_{P_c} = \sum_{m=1}^{M_c} \alpha_{c,m} \mu_{c,m}, \qquad \mu_{Q_c} = \sum_{m=1}^{M_c} \beta_{c,m} \mu_{c,m}. \tag{22}$$

Hence, we have:

$$\mu_{P_c} - \mu_{Q_c} = \sum_{m=1}^{M_c} (\alpha_{c,m} - \beta_{c,m}) \mu_{c,m}. \tag{23}$$

Using $\mathrm{MMD}^2_{k_X}(P_c, Q_c) = \|\mu_{P_c} - \mu_{Q_c}\|^2_{\mathcal{H}}$, we can then obtain

$$\begin{aligned} \mathrm{MMD}^2_{k_X}(P_c, Q_c) &= \left\| \sum_m (\alpha_{c,m} - \beta_{c,m}) \mu_{c,m} \right\|^2_{\mathcal{H}} & (24) \\ &= \sum_{i,j} (\alpha_{c,i} - \beta_{c,i})(\alpha_{c,j} - \beta_{c,j}) \langle \mu_{c,i}, \mu_{c,j} \rangle_{\mathcal{H}}. & (25) \end{aligned}$$

By definition, we have

$$(G_c)_{ij} = \langle \mu_{c,i}, \mu_{c,j} \rangle_{\mathcal{H}}, \tag{26}$$

Then Eq.24 becomes

$$\mathrm{MMD}^2_{k_X}(P_c, Q_c) = (\alpha_c - \beta_c)^\top G_c (\alpha_c - \beta_c). \tag{27}$$

Substituting into the conditional decomposition in Eq.20 yields

$$\mathrm{MMD}^2_k = \sum_c p(c) (\alpha_c - \beta_c)^\top G_c (\alpha_c - \beta_c), \tag{28}$$

which proves the result. □