Generalised Flow Maps on Riemannian Manifolds

Oscar Davis[*]
University of Oxford

Avishek "Joey" Bose
University of Oxford
Mila – Québec AI Institute

Abstract

Recent advances in generative modelling on Euclidean spaces have shown how to train models that achieve state-of-the-art quality from scratch, while requiring only few function evaluations. In the meanwhile, data supported on Riemannian manifolds, such as protein backbones or geological data, has lagged behind: inference and training on such manifolds remain computationally challenging and numerically unstable due to the need for manifold-specific operations, and require numerous evaluations of a potentially expensive model to obtain samples of high quality. In this paper, we propose Riemannian Flow Maps, a new class of few-step generative models that generalise the flow map framework [Boffi et al., 2024] to arbitrary Riemannian manifolds. We port and design three self-distillation-based training methods: Generalised Lagrange Flow Maps, Generalised Eulerian Flow Maps, and Generalised Progressive Flow Maps, all of which recover their Euclidean counterparts. Empirically, we test Generalised Flow Maps on a host of standard datasets, and achieve, amongst Riemannian generative models, state-of-the-art sample quality for single- and few-step evaluations, and log-likelihoods.

1 Introduction

Data that can be represented on a Riemannian manifold is ubiquitous, and examples include protein backbones on $\mathrm{SE}(3)^N$ [Bose et al., 2024], geological data on $\mathbb{S}^2$ [Mathieu and Nickel, 2020], discrete data using the Fisher-Rao probability simplex [Davis et al., 2024, Cheng et al., 2025]. Currently, the state-of-the-art methods for modelling these [Chen and Lipman, 2024, Bortoli et al., 2022, Huang et al., 2022] involve, at inference time, integrating a learnt vector field, requiring either many (100 or more) integration steps or adaptive samplers, and in some cases even considering stochastic calculus on Riemannian manifolds [Huang et al., 2022, Bortoli et al., 2022], which is non-trivial in theory and in practice, as closed-form solutions often lack and are replaced by approximations introducing biases. As well, each sampling step typically involving a Riemannian operation, such as the exponential or logarithmic map, each step suffers from the inaccuracies induced by their numerical imprecision, especially in high dimensions. Therefore, reducing the number of Riemannian operations, while maintaining this powerful inductive bias, should lead to higher fidelity samples. Concurrently, on Euclidean spaces, recent advances have shown how to stably train state-of-the-art generative models from scratch that only require a few (usually less than ten, and down to one) sampling steps [Boffi et al., 2025, Geng et al., 2025, Zhou et al., 2025, Sabour et al., 2025].

In this work, we propose Generalised flow maps, a method generalising and extending flow maps [Boffi et al., 2025] to Riemannian manifolds. We present three flavours of self-distillation losses (*i.e.*, losses that use older parameters as teacher models [Song et al., 2023, Lu and Song, 2025, Sabour et al., 2025, Geng et al., 2025, Zhou et al., 2025]) that recover the three methods proposed by flow maps on Euclidean spaces. Moreover, our experiments demonstrate the capabilities of our method on a host of datasets supported on Riemannian manifolds, and exhibit state-of-the-art performance for single- and few-step sample quality while achieving competitive negative log-likelihood values.

[*]Correspondence to: `oscar.davis@cs.ox.ac.uk`. Code available at github.com/olsdavis/gfm.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

2 Background and preliminaries

2.1 Riemannian geometry

A smooth manifold, $\mathcal{M} \subseteq \mathbb{R}^d$, $d \in \mathbb{N}^\star$, is a smooth topological space. This smoothness naturally defines tangency of a vector $v \in \mathrm{T}_p\mathcal{M}$ to a point $p \in \mathcal{M}$, where $\mathrm{T}_p\mathcal{M}$ is the tangent space at point p. Each $\mathrm{T}_p\mathcal{M}$ being a vector space, endowing it with an inner product, g_p, makes $\mathrm{T}_p\mathcal{M}$ an inner product space. If g_p is positive semi-definite, symmetric, and bilinear, and varies smoothly with p, then the pair $(\mathcal{M}, g)$ is said to be a Riemannian manifold, and g a metric. This metric in turn defines, on the manifold, distance between points, angles and lengths of vectors. The exponential map, $\exp_p : \mathrm{T}_p\mathcal{M} \to \mathcal{M}$, generalises $p + v$, and amounts to taking a step in the direction of the vector v. The logarithmic map, $\log_p : \mathcal{M} \to \mathrm{T}_p\mathcal{M}$, generalises $q - p$, and provides a vector pointing towards its argument. The shortest path/geodesic between two points $p, q \in \mathcal{M}$ is the curve $\gamma : [0, 1] \to \mathcal{M}$, with $\gamma(0) = p$ and $\gamma(1) = q$, that minimises the energy functional, $\gamma \mapsto \frac{1}{2}\int_0^1 \|\dot\gamma(t)\|^2_{\gamma(t)}\,\mathrm{d}t$.

2.2 Flow Maps

Let ρ_0, ρ_1 be distributions over $\mathbb{R}^d$, $d \in \mathbb{N}^\star$. A stochastic interpolant is defined by $I : [0, 1] \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d, (t, x_0, x_1) \mapsto I_t(x_0, x_1) = \alpha_t x_0 + \beta_t x_1$, for some $\alpha, \beta \in \mathrm{C}^1([0, 1], [0, 1])$, such that $\alpha_0 = 1$, $\alpha_1 = 0$, $\beta_0 = 0$ and $\beta_1 = 1$. Drawing $(x_0, x_1) \sim \rho(x_0, x_1)$ with marginals ρ_0 and ρ_1, the interpolant defines a probability path, $(\rho_t)_{t\in[0,1]}$, between ρ_0 and ρ_1. For any $x \in \mathbb{R}^d$, $t \in (0, 1)$, $v_t(x) = \mathbb{E}[\dot I_t \mid I_t = x]$, $\dot x_t = v_t(x_t)$ with $x_0 \sim \rho_0$, defines a valid probability flow. A tractable alternative to learn v_t is to learn the conditional velocity field, $v_t(\cdot|x_0, x_1)$, everywhere along the interpolant path, which can then be used to transport ρ_0 to ρ_1 with an ODE [Albergo et al., 2023, Lipman et al., 2023].

Flow maps as introduced in Boffi et al. [2024, 2025] are functions $X : [0, 1]^2 \times \mathbb{R}^d \to \mathbb{R}^d$ that satisfy the *jump condition*: $X_{s,t}(x_s) = x_t$, where $(x_t)_{t\in[0,1]}$ is any solution of the probability flow. One can thus sample from ρ_1 by first sampling $x_0 \sim \rho_0$ and then applying $X_{0,1}(x_0) \sim \rho_1$. These are parametrised as $X_{s,t}(x) = x + (t - s)v_{s,t}(x)$. To learn a flow map, we can use any flavour of self-distillation alongside the typical flow matching loss [Albergo et al., 2023, Tong et al., 2023, Lipman et al., 2023], $\mathcal{L}(\hat v) = \mathcal{L}_{\mathrm{SD}}(\hat v) + \mathcal{L}_{\mathrm{FM}}(\hat v)$, and we state all three proposed in Boffi et al. [2025] in appendix A.1.

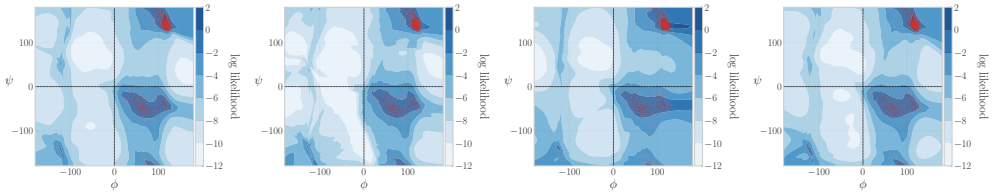
3 Method

We are interested in the generative modelling task, where we intend to model a distribution, p_{data}, of which we only have $n \in \mathbb{N}^\star$ samples, by using a parametric family $(p_\theta)_{\theta\in\Theta}$, where the choice of parameters θ is typically updated with a variant of gradient descent. Moreover, here, we consider the support of the distribution to be a *known* Riemannian manifold, $(\mathcal{M}, g)$. Finally, we denote by sg the stop-gradient operator

We adapt each of the self-distillation objectives to the Riemannian setting. Hereinafter, $\hat X$ and $\hat v$ denote our approximations of X and v respectively, and $I_s \equiv I_s(x_0, x_1)$. Following Chen and Lipman [2024], we constrain our vector field to lie on the tangent plane at the point that it is evaluated. Specifically, letting $f^\theta : [0, 1]^2 \times M \to \mathbb{R}^d$ be our underlying neural network, we compute its tangential projection at $p \in \mathcal{M}$; that is to say, $\forall 0 \le s \le t \le 1, \forall p \in \mathcal{M}, v^\theta_{s,t}(p) = \mathrm{proj}_{\mathrm{T}_p\mathcal{M}}\left(f^\theta_{s,t}(p)\right)$. This ensures that all the usual Riemannian manifold operations required are not ill-defined. Furthermore, we now also re-define X (and by extension $\hat X$) as, for any $0 \le s \le t \le 1, x \in \mathcal{M}$, $X_{s,t}(x) = \exp_x((t - s)v_{s,t}(x)) \in \mathcal{M}$, and similarly, for all $x_0, x_1 \in \mathcal{M}$, $I_s(x_0, x_1) = \exp_{x_0}(\beta_s \log_{x_0}(x_1))$ (*i.e.*, the geodesic interpolant between x_0 and x_1). Note that, here, we cannot any more have arbitrary choices for α and β, as it is unclear what the point $\alpha_t x_0 + \beta_t x_1$, for arbitrary x_0, x_1, would correspond to on $\mathcal{M}$, especially when $\alpha_t \not\equiv (1 - \beta_t)$. We always set $\beta : s \mapsto s$.

Lagrangian self-distillation. To port Lagrangian self-distillation (LSD), we make the observation that $\partial_t \hat X_{s,t}(I_s)$ lives on the tangent space of $\hat X_{s,t}(I_s)$, and so does $\hat v_{t,t}(\hat X_{s,t}(I_s))$ by construction. Consequently, any linear combination of those vectors is also a vector in the tangent space. However,

Figure 1: Ramachandran plots on the General protein dataset. Red dots represent the test-set samples.



(a) ESD (b) LSD (c) PSD (d) RFM

we need to adapt the loss to incorporate the Riemannian metric in the norm of the resulting vector:

$$\mathcal{L}_{\text{G-LSD}}(\hat{v}) = \mathbb{E}_{t,s,x_0,x_1}\left[\left\|\partial_t \hat{X}_{s,t}(I_s) - \text{sg}\left(\hat{v}_{t,t}(\hat{X}_{s,t}(I_s))\right)\right\|_g^2\right]. \tag{1}$$

Eulerian self-distillation. In an analogous manner to LSD, we can instantiate an Eulerian loss for the Riemannian setting. Specifically, Eulerian self-distillation (ESD), as defined in (5), may seem challenging to port in its given formulation, but we can generalise it using the differential of the map:

$$\mathcal{L}_{\text{G-ESD}}(\hat{v}) = \mathbb{E}_{t,s,u,x_0,x_1}\left[\left\|\partial_s X^\theta_{s,t}(x_s) + \text{sg}\left(\text{d}(X^\theta_{s,t})_{I_s}[v^\theta_{s,s}(I_s)]\right)\right\|_g^2\right]. \tag{2}$$

Note that we also train naively the objective of Mean Flows [Geng et al., 2025], which we refer to as G-MF.

Progressive self-distillation. Progressive self-distillation (PSD) is arguably the method that is most natural to port to Riemannian manifolds. Given our redefinition of X using exponential maps, $X_{u,t} \circ X_{s,u} \in \mathcal{M}$ from (6) is well-defined for all $0 \leq s \leq u \leq t \leq 1$. The only major change is that in the choice of distance, which should now be d_g^2. The loss can therefore be rewritten as,

$$\mathcal{L}_{\text{G-PSD}}(\hat{v}) = \mathbb{E}_{t,s,u,x_0,x_1}\left[d_g^2\left(\hat{X}_{s,t}(I_s), \text{sg}\left(\hat{X}_{u,t}\left(\hat{X}_{s,u}(I_s)\right)\right)\right)\right]. \tag{3}$$

For training, we set $u = \frac{1}{2}s + \frac{1}{2}t$, and apply the stop-gradient operator to the second argument in d_g^2.

4 Experiments

We test our proposed methods on a few usual datasets: proteins ($\mathbb{S}^1 \times \mathbb{S}^1$), catastrophic events on Earth ($\mathbb{S}^2$), and a synthetic dataset on $\text{SO}(3)$. To evaluate these, alongside qualitative results, we use the negative log-likelihood (NLL) on the test set, and the MMD using an RBF kernel combined with the distance on the manifold in question, with a bandwith of $\gamma = 1$. To compute the test NLL, we use the instantaneous vector field, $\hat{v}_{t,t}(\cdot)$, and integrate it backwards in time for all test samples. For all tasks, we use Riemannian Flow Matching (RFM) [Chen and Lipman, 2024] as a baseline. We also re-train RFM as so to match our lower compute budget (in training time), and for us to be able to evaluate its MMDs.

Protein datasets. We train our model on a protein dataset on the flat 2D and 7D (RNA) tori ($\mathbb{S}^1 \times \mathbb{S}^1$ and $(\mathbb{S}^1)^7$). The 2D data is from Lovell et al. [2003] and the 7D data from Murray et al. [2003], compiled together by Huang et al. [2022]. We report our results in figs. 2 and 5 and figs. 1 and 3. We observe that the NLLs produced by all our methods are on par or better, e.g. G-PSD, than those of RFM, except on the RNA dataset. Most importantly, figs. 2 and 3 show that, for low NFEs, our methods offer considerable gains, especially G-LSD, which offers an improvement of up to $22\times$ on the MMD (on "General") for a single function evaluation. Furthermore, fig. 1 shows that our methods achieve log-likelihood landscapes comparable to those of RFM. Finally, we can also see in fig. 3 that not only our methods consistently improve on the MMD for low NFEs, but that their quality does not degrade or become worse than that of RFM with a high number of NFEs.

Synthetic data on $\text{SO}(3)$. We also instantiate our methods on $\text{SO}(3)$, using the synthetic dataset from Huang et al. [2022]. Our results are displayed in table 1. Once more, we can see that all our methods outperform RFM, while equally precisely capturing the log-likelihoods of the true dataset. This demonstrates the effectiveness of all our methods on non-trivial manifolds.

Table 1: Results on the $\mathrm{SO}(3)$ test set with standard deviation estimated over 5 seeds.

	MMD (vs. NFE)			**NLL**
	1	2	100	
RFM	0.147 ± 0.007	0.083 ± 0.003	0.042 ± 0.002	-7.15 ± 0.03
G-ESD (ours)	0.146 ± 0.01	0.087 ± 0.01	$\mathbf{0.039 \pm 0.002}$	-7.15 ± 0.03
G-LSD (ours)	$\mathbf{0.064 \pm 0.007}$	$\mathbf{0.059 \pm 0.005}$	0.044 ± 0.008	-7.11 ± 0.03
G-PSD (ours)	0.121 ± 0.01	0.073 ± 0.005	$\mathbf{0.039 \pm 0.004}$	-7.15 ± 0.03
G-MF (ours)	0.291 ± 0.007	0.280 ± 0.015	0.283 ± 0.014	-6.85 ± 0.03

Figure 2: Test MMD on protein datasets for 1 NFE. Standard deviation estimated over 5 runs.

	General (2D)	**Glycine (2D)**	**Proline (2D)**	**Pre-Pro (2D)**	**RNA (7D)**
RFM	0.45 ± 0.006	0.27 ± 0.008	0.52 ± 0.057	0.47 ± 0.022	0.68 ± 0.011
G-LSD (ours)	$\mathbf{0.02 \pm 0.003}$	$\mathbf{0.03 \pm 0.004}$	$\mathbf{0.04 \pm 0.012}$	$\mathbf{0.05 \pm 0.004}$	$\mathbf{0.08 \pm 0.007}$
G-PSD (ours)	0.11 ± 0.016	0.05 ± 0.019	0.07 ± 0.011	0.08 ± 0.015	0.14 ± 0.027
G-ESD (ours)	0.29 ± 0.006	0.12 ± 0.009	0.44 ± 0.024	0.24 ± 0.023	0.50 ± 0.004
G-MF (ours)	0.11 ± 0.029	0.04 ± 0.019	0.09 ± 0.046	0.09 ± 0.017	0.20 ± 0.019

Figure 3: MMD as a function of NFEs on the protein datasets.

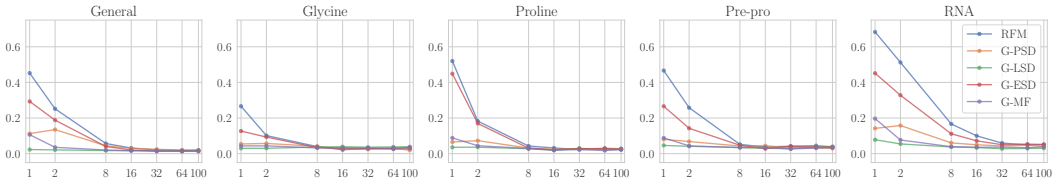


Figure 4: MMD as a function of NFEs on the Earth datasets.

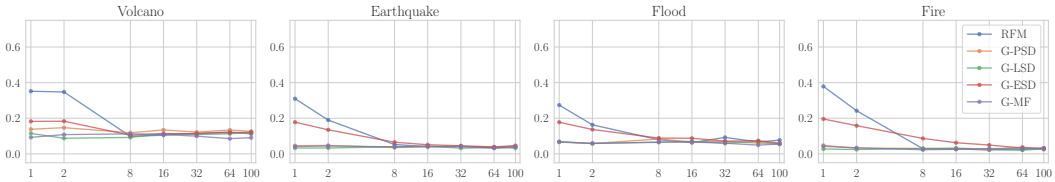


Figure 5: Test NLL on protein datasets. Standard deviation estimated over 5 runs.

	General (2D)	**Glycine (2D)**	**Proline (2D)**	**Pre-Pro (2D)**	**RNA (7D)**
Dataset size	138,208	13,283	7,634	6,910	9,478
RFM	1.01 ± 0.025	1.90 ± 0.055	0.15 ± 0.027	1.18 ± 0.055	-5.20 ± 0.067
G-MF (ours)	0.97 ± 0.01	1.97 ± 0.01	0.21 ± 0.04	$\mathbf{1.02 \pm 0.04}$	-3.79 ± 0.09
G-LSD (ours)	0.99 ± 0.05	1.99 ± 0.02	0.24 ± 0.07	1.11 ± 0.02	-4.15 ± 0.09
G-PSD (ours)	$\mathbf{0.95 \pm 0.02}$	$\mathbf{1.94 \pm 0.03}$	$\mathbf{0.08 \pm 0.04}$	1.10 ± 0.04	-4.40 ± 0.13
G-ESD (ours)	0.99 ± 0.01	1.97 ± 0.04	0.11 ± 0.07	1.10 ± 0.04	-4.01 ± 0.11

Earth datasets. Finally, we evaluate our model on a collection of Earth datasets, living on the 2-sphere, $\mathbb{S}^2$; the data was collected by Mathieu and Nickel [2020] from various sources [NOAA, 2020a,b, Brakenridge, 2017, EOSDIS, 2020]. We report our results for our best performing method in fig. 4 and table 2. Once more, while our method offers log-likelihoods that are on par with RFM for a similar compute budget, fig. 4 clearly demonstrates the great improvement on MMD our method offers for low NFEs of 1 or 2 steps, while preserving the quality for higher NFEs.

5 Conclusion

We have shown and generalised, in this work, three self-distillation methods to train few-step generative models on Riemannian manifolds, which recover their known counterparts when instantiated on Euclidean spaces. These three methods show state-of-the-art performance on a host of datasets for single- and few-step generation, while also being on par with the state-of-the-art negative log-likelihoods. We believe that this work may enable faster and more accurate generative modelling on Riemannian manifolds. Current limitations include the fact that these manifolds have closed-form expressions for their usual operations, and we have not considered approximations thereof, such as meshes. Finally, guidance is also another technique of interest for generative modelling that should be explored in this context.

Acknowledgements

OD is funded by both Project CETI and Intel. AJB is partially supported by an NSERC Post-doc fellowship. This research is partially supported by the EPSRC Turing AI World-Leading Research Fellowship No. EP/X040062/1 and EPSRC AI Hub No. EP/Y028872/1.

References

M. S. Albergo, N. M. Boffi, and E. Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions, 2023. URL https://arxiv.org/abs/2303.08797.

N. M. Boffi, M. S. Albergo, and E. Vanden-Eijnden. Flow map matching. *arXiv preprint arXiv:2406.07507*, 2, 2024.

N. M. Boffi, M. S. Albergo, and E. Vanden-Eijnden. How to build a consistency model: Learning flow maps via self-distillation, 2025. URL https://arxiv.org/abs/2505.18825.

V. D. Bortoli, E. Mathieu, M. Hutchinson, J. Thornton, Y. W. Teh, and A. Doucet. Riemannian score-based generative modelling, 2022. URL https://arxiv.org/abs/2202.02763.

A. J. Bose, T. Akhound-Sadegh, G. Huguet, K. Fatras, J. Rector-Brooks, C.-H. Liu, A. C. Nica, M. Korablyov, M. Bronstein, and A. Tong. Se(3)-stochastic flow matching for protein backbone generation, 2024. URL https://arxiv.org/abs/2310.02391.

G. R. Brakenridge. Global active archive of large flood events. http://floodobservatory.colorado.edu/Archives/index.html, 2017. Dartmouth Flood Observatory, University of Colorado.

R. T. Q. Chen and Y. Lipman. Flow matching on general geometries, 2024. URL https://arxiv.org/abs/2302.03660.

R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud. Neural ordinary differential equations, 2019. URL https://arxiv.org/abs/1806.07366.

C. Cheng, J. Li, J. Peng, and G. Liu. Categorical flow matching on statistical manifolds, 2025. URL https://arxiv.org/abs/2405.16441.

O. Davis, S. Kessler, M. Petrache, İsmail İlkan Ceylan, M. Bronstein, and A. J. Bose. Fisher flow matching for generative modeling over discrete data, 2024. URL https://arxiv.org/abs/2405.14664.

EOSDIS. Active fire data. https://earthdata.nasa.gov/earth-observation-data/near-real-time/firms/active-fire-data, 2020. Land, Atmosphere Near real-time Capability for EOS (LANCE) system operated by NASA's Earth Science Data and Information System (ESDIS).

K. Frans, D. Hafner, S. Levine, and P. Abbeel. One step diffusion via shortcut models, 2025. URL https://arxiv.org/abs/2410.12557.

Z. Geng, A. Pokle, W. Luo, J. Lin, and J. Z. Kolter. Consistency models made easy, 2024. URL https://arxiv.org/abs/2406.14548.

Z. Geng, M. Deng, X. Bai, J. Z. Kolter, and K. He. Mean flows for one-step generative modeling, 2025. URL https://arxiv.org/abs/2505.13447.

C.-W. Huang, M. Aghajohari, A. J. Bose, P. Panangaden, and A. Courville. Riemannian diffusion models, 2022. URL https://arxiv.org/abs/2208.07949.

A. Hyvärinen. Estimation of non-normalized statistical models by score matching. *J. Mach. Learn. Res.*, 6:695–709, Dec. 2005. ISSN 1532-4435.

D. Kim, C.-H. Lai, W.-H. Liao, N. Murata, Y. Takida, T. Uesaka, Y. He, Y. Mitsufuji, and S. Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion, 2024. URL https://arxiv.org/abs/2310.02279.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling, 2023. URL https://arxiv.org/abs/2210.02747.

I. Loshchilov and F. Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

S. C. Lovell, I. W. Davis, W. B. Arendall, 3rd, P. I. W. de Bakker, J. M. Word, M. G. Prisant, J. S. Richardson, and D. C. Richardson. Structure validation by calpha geometry: phi,psi and cbeta deviation. *Proteins*, 50(3):437–450, Feb. 2003.

C. Lu and Y. Song. Simplifying, stabilizing and scaling continuous-time consistency models, 2025. URL https://arxiv.org/abs/2410.11081.

E. Mathieu and M. Nickel. Riemannian continuous normalizing flows, 2020. URL https://arxiv.org/abs/2006.10605.

L. J. W. Murray, W. B. Arendall, D. C. Richardson, and J. S. Richardson. Rna backbone is rotameric. *Proceedings of the National Academy of Sciences*, 100(24):13904–13909, 2003. doi: 10.1073/pnas.1835769100. URL https://www.pnas.org/doi/abs/10.1073/pnas.1835769100.

NOAA. Global significant earthquake database. https://data.nodc.noaa.gov/cgi-bin/iso?id=gov.noaa.ngdc.mgg.hazards:G012153, 2020a. National Geophysical Data Center / World Data Service (NGDC/WDS): NCEI/WDS Global Significant Earthquake Database. NOAA National Centers for Environmental Information.

NOAA. Global significant volcanic eruptions database. https://data.nodc.noaa.gov/cgi-bin/iso?id=gov.noaa.ngdc.mgg.hazards:G10147, 2020b. National Geophysical Data Center / World Data Service (NGDC/WDS): NCEI/WDS Global Significant Volcanic Eruptions Database. NOAA National Centers for Environmental Information.

A. Sabour, S. Fidler, and K. Kreis. Align your flow: Scaling continuous-time flow map distillation, 2025. URL https://arxiv.org/abs/2506.14603.

Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

Y. Song, P. Dhariwal, M. Chen, and I. Sutskever. Consistency models, 2023. URL https://arxiv.org/abs/2303.01469.

A. Tong, K. Fatras, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, G. Wolf, and Y. Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

L. Zhou, S. Ermon, and J. Song. Inductive moment matching, 2025. URL https://arxiv.org/abs/2503.07565.

A Appendix

A.1 Flow map matching losses

We state here the three self-distillation losses proposed in Boffi et al. [2024]. The first one, LSD, amounts to optimising the following loss:

$$\mathcal{L}_{\mathrm{LSD}}(\hat{v}) \coloneqq \mathbb{E}_{t,s,x_0,x_1}\left[\left\|\partial_t \hat{X}_{s,t}(I_s) - \hat{v}_{t,t}(\hat{X}_{s,t}(I_s))\right\|_2^2\right], \tag{4}$$

where $\hat{v}$ and $\hat{X}$ correspond to our approximations of v and X, respectively, and $I_s \equiv I_s(x_0, x_1)$; the second one, ESD, to

$$\mathcal{L}_{\mathrm{ESD}}(\hat{v}) \coloneqq \mathbb{E}_{t,s,x_0,x_1}\left[\left\|\partial_s \hat{X}_{s,t}(I_s) + \nabla \hat{X}_{s,t}(I_s)\hat{v}_{t,t}(I_s)\right\|_2^2\right]; \tag{5}$$

and the last one, PSD, based on the semi-group property of flow maps, is

$$\mathcal{L}_{\mathrm{PSD}}(\hat{v}) \coloneqq \mathbb{E}_{t,s,u,x_0,x_1}\left[\left\|\hat{X}_{s,t}(I_s) - \hat{X}_{u,t}\left(\hat{X}_{s,u}(I_s)\right)\right\|_2^2\right]. \tag{6}$$

A.2 Related works

Consistency models. We designate by consistency models, here, the broad class of generative models that output high quality samples with low NFE, and that are typically trained with an underlying stochastic (often Gaussian, in the Euclidean case) process, because of the seminal work of Song et al. [2023]. Those have been found to be unstable to train in their original formulation, and were therefore refined in Lu and Song [2025], where mostly many multiple training tricks were proposed to stabilise the training. Geng et al. [2024] have shown a simpler and more stable objective for consistency models while offering comparable quality. The aforementioned papers were mostly based around diffusion processes; however, many different classes of consistency models have emerged since then. For instance, Zhou et al. [2025] model a distribution of paths and minimises the MMD between these at different times, using old weights at lower times (closer to data) as a teacher model for the current weights at higher times (closer to noise). Flow maps are also extensively discussed in the main body of the paper [Boffi et al., 2025, 2024]. Mean Flows model the neural network as the integral/average of the instantaneous vector field for arbitrary time steps [Geng et al., 2025]. Shortcut models [Frans et al., 2025] propose a training procedure very similar to the PSD objective of Boffi et al. [2025], but in discrete time. Finally, consistency trajectory models [Kim et al., 2024] essentially estimate two-time flow maps. Remark that we are not interested, in this work, by *distillation* models, that take a separate teacher model and distil it into another (typically smaller) model.

Riemannian generative models. Generative modelling on Riemannian manifolds remains a challenging task with quite a few different proposals that were offered. One of the first formulations is that of Riemannian Diffusion Models [Huang et al., 2022], where the authors attempted to port directly the evidence lower bound (ELBO) of Euclidean diffusion models to Riemannian geometry; moreover, the authors are then faced with stochastic calculus on Riemannian manifolds, which may namely fail to be simulation-free, naively. Perhaps the most well-known method is that of RFM [Chen and Lipman, 2024], where the authors have ported usual flow matching [Lipman et al., 2023] to Riemannian geometries. Flow matching's formulation using an ODE allows for the same properties on the manifold as on the Euclidean space, simulation-free. Bortoli et al. [2022] proposed a method porting denoising score matching [Song and Ermon, 2020, Hyvärinen, 2005] to Riemannian manifolds. In a similar vain, Mathieu and Nickel [2020] ported continuous normalising flows to Riemannian manifolds, with the drawbacks of CNFs [Chen et al., 2019] (namely their cost to train and their lower performance, compared to that of diffusion/flow matching).

A.3 Experimental details

For the earth and protein datasets, we train all models using the same architectures as in RFM, with the same parameter count, adapted to include two time parameters, with the exact same hyperparameters, except for the optimiser, where we use AdamW [Loshchilov and Hutter, 2019] instead of

Table 2: Test NLL on Earth datasets. Standard deviation estimated over 5 runs.

	Volcano	**Earthquake**	**Flood**	**Fire**
Dataset size	827	6,120	4,875	12,809
RFM	-7.93 ± 1.67	-0.28 ± 0.08	0.42 ± 0.05	-1.86 ± 0.11
RFM (re-run)	-3.88 ± 0.31	-0.76 ± 0.032	-0.82 ± 0.04	-2.80 ± 0.25
R-MF (ours)	-3.44 ± 0.33	-0.75 ± 0.31	-0.72 ± 0.09	-2.51 ± 0.13

Adam [Kingma and Ba, 2017]. As for the inclusion of the two time parameters, we naively sum the linear embeddings of both time steps—we believe more architectural work could further improve the performance of our models. For the $\mathrm{SO}(3)$ dataset, we use the same architecture but with less parameters, still maintained equal across all methods.

In RFM, according to its publicly available repository, and its configuration files, are trained for $1{,}000{,}000$ training steps. We re-trained all models for about $10\times$ less steps, although the exact number varies, as we employed early stopping based on the validation NLL.

We ran all our numbers with all the same 5 randomly chosen seeds of 42, 123, 456, 789 and 1337.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, the claims in the abstract, the introduction and in the content in general reflect the contributions of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We briefly outline in the conclusion the limitations of the current work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The assumptions are made clear where required, and some details are further added in the appendix for rigour.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The code is publicly available and the experiments fully reproducible.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

Justification: Yes, all instructions are provided in the public repository, and the data is all publicly available, with instructions available for download and extraction.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Again, the publicly available code contains the exact configurations used for the training and test settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper does report error bars gathered over 5 random seeds.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Sufficient details are provided in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This paper conforms to the Code of Ethics in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: This work's scope is more theoretical and its direct applications to nefarious intents are limited as such.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The authors of each dataset used in this paper have been properly cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: NA

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The usage of LLMs has been sparse, and has not affected the main body of text, the methodology or any other "core" aspect of the paper in any significant, direct way.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.