Improving Flow Matching for Simulation-Based Inference

Janis Fluri
Department of Computer Science
ETH Zurich
janis.fluri@inf.ethz.ch

Thomas Hofmann
Department of Computer Science
ETH Zurich
thomas.hofmann@inf.ethz.ch

Abstract

Flow matching is an incredibly powerful technique that can be used to sample arbitrary distributions. Recently, flow matching posterior estimation (FMPE) has been introduced in simulation-based inference (SBI) as a scalable alternative to standard neural posterior estimation (NPE) using normalizing flows. However, FMPE suffers from a lower sampling efficiency than NPE because it requires multiple network evaluations per sample. In this work, we propose extensions of FMPE based on mini-batch optimal transport that reduce the required number of network evaluations for high-quality samples. We investigate the extensions theoretically and show that they can lead to straight probability flows in the appropriate limit. Finally, we demonstrate the performance of the method on simple toy models and high-dimensional gravitational wave source parameter inference, showing that it is possible to increase the sample efficiency of FMPE while achieving approximately the same performance over different tasks.

1 Introduction

Bayesian inference connects model parameters to observed data via the likelihood function [1], but in many cases, this function is intractable or difficult to evaluate. Simulation-based inference (SBI), or likelihood-free inference, addresses this by using numerical simulators without needing to explicitly calculate the likelihood [2]. Traditional SBI methods like approximate Bayesian computation (ABC) [3, 4] and synthetic likelihood approaches [5, 6] can model complex distributions but struggle with high-dimensional data and require many simulations. More efficient approaches use neural networks [7–13], including normalizing flows [14], to approximate distributions. Theoretically, normalizing flows are universal density approximators [14], however, they face challenges in scaling and efficiency.

Recently, advanced generative models such as score-matching diffusion [15–17], consistency models [18, 19] and flow matching [20, 21] have been applied to SBI [22–26]. These models, unlike normalizing flows, use a vector field with a continuous time parameter, requiring the solution of a differential equation for sampling [27]. This allows for more flexible network architectures but increases the computational cost. This work explores methods to reduce the number of function evaluations in flow matching for SBI, proposing new generalizations of mini-batch optimal transport [28, 29] to improve sample efficiency.

Related Work Flow matching was first used for SBI in [26]. Mini-batch optimal couplings have been investigated by [28] and [29] for unconditional or simple class-conditional generation. We will mostly build upon the theory and notation introduced in [28]. Other works that involve continuous normalizing flows and optimal transport are for example [30–34]. However, these works use a significantly different approach and try to improve the flows with regularization or different training objectives. Neural optimal transport techniques [35–38] create generative models that try to learn

Machine Learning and the Physical Sciences Workshop, NeurIPS 2024.

the optimal transport map directly with neural networks. These usually do not involve continuous normalizing flows.

Contributions We propose three generalizations of mini-batch optimal transport for SBI, introducing additional terms into the cost matrix. We theoretically and empirically evaluate these methods, showing they can straighten flows while maintaining performance.

2 Flow Matching & SBI

Preliminaries The goal of simulation-based inference (SBI) is to sample or estimate the posterior distribution $p(\theta|x)$ of the underlying model parameters θ given empirically observed data x. In SBI we only have indirect access to the likelihood $p(x|\theta)$ via a simulator that can sample $x \sim p(x|\theta)$ for any parameter vector θ inside the support of our prior $p(\theta)$. There are SBI techniques based on modelling the posterior distribution, likelihood function and likelihood-to-evidence ratio [7–13]. Additionally, they can be divided into amortized and sequential methods, where amortized methods learn a general model of the posterior and sequential models usually work iteratively and require new simulations to perform inference on new observations. In this work, we will only consider amortized methods that estimate (and sample) the posterior distribution directly.

Continous Normalizing Flows Normalizing flows [14] transform a simple base distribution q_0 into an arbitrary distribution q_1 via an invertible function ψ parameterized by a neural network. Learning invertible functions greatly restricts the possible architectures of the neural networks and fundamentally limits the scalability of the method. Continuous normalizing flows [27] overcome this limitation by instead learning a vector field v_t with a continuous time component $t \in [0, 1]$ that smoothly transforms the initial distribution into the target distribution. For SBI, we additionally condition the vector field on the observed data x. The vector field defines the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi_{t,x}(\theta) = v_{t,x}(\psi_{t,x}(\theta))), \qquad \psi_{0,x}(\theta) = \theta \sim q_0. \tag{1}$$

This differential equation, commonly referred to as neural ODE, has a unique solution if the vector field is continuous in t and Lipschitz continuous in θ_t, which is true for most neural network architectures used in practice. The density of the learned posterior distribution can be evaluated by using the instantaneous change of variables [27] and integrating over time

$$q(\theta|x) = q_0(\theta_0) \exp\left(-\int_0^1 \nabla \cdot v_{t,x}(\theta_t)\mathrm{d}t\right), \tag{2}$$

where $\theta_t \equiv \psi_{t,x}(\theta)$ follows the trajectory of a solution of the neural ODE (1) and the divergence is calculated with respect to θ_t. Continuous normalizing flows can be trained like normalizing flow by maximizing the (log-)likelihood of the target distribution. However, the evaluation of density via the integral in equation (2) requires multiple network evaluations for a single mini-batch, making training overall more costly than normalizing flows.

Flow Matching Flow matching [20] provides an alternative objective function to train continuous normalizing flows. It does not require the evaluation of the density via equation (2), instead, it directly optimizes the vector field $v_{t,x}$. We refer the interested reader to [20] for a detailed description of the formalism and focus on flow matching posterior estimation introduced in [26]. The FMPE loss is given by

$$\mathcal{L}_{\text{FMPE}} = \mathbb{E}_{t\sim\lambda(t),x\sim p(x|\theta_1),\theta_1\sim p(\theta),\theta_0\sim\mathcal{N}(0,1)}\|v_{t,x}(\theta_t) - (\theta_1 - \theta_0)\|^2, \tag{3}$$

where $p(\theta)$ is the prior of the model parameters, $p(x|\theta_1)$ is sampled via the simulator, the initial parameters θ_0 are sampled from a standard normal distribution and the time component is usually sampled uniformly $t \sim \lambda(t) = \mathcal{U}[0, 1]$. This loss is obtained from the original flow matching loss proposed in [20] by applying Bayes' theorem and using the observation x as a condition of the flow. Therefore, it converges to the true probability flow that transforms samples from the initial distribution to posterior samples. It has been shown [26] that FMPE achieves competitive results on the SBI benchmark suite [39] and high-dimensional gravitational wave data where the observed data vector has $\sim 10,000$ dimensions. However, sampling the trained posterior takes significantly longer than for standard normalizing flows. A possible way to increase the sample efficiency is to

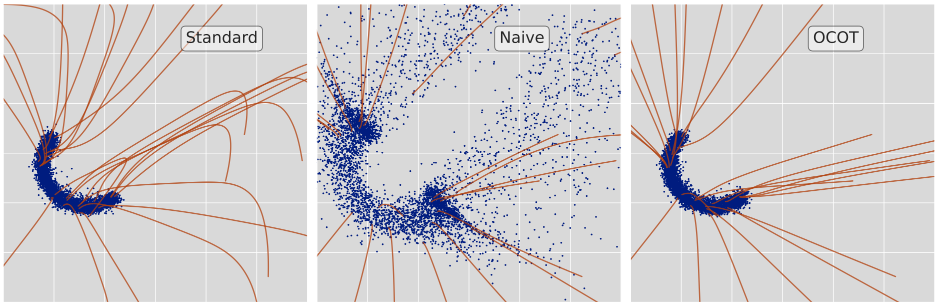

Figure 1: The sampled posterior distributions and example trajectories of an FMPE model trained on the Two Moons task. The left panel shows a model trained with the standard FMPE objective (curved trajectories), the middle panel shows the naive application ($m = \beta = 0$) of mini-batch optimal transport (leads to failure) and the right panel shows a model trained using OCOT and a match rate of $m = 0.3$ (straight trajectories, correct posterior).

"straighten" the flows of the trained vector field. Straightness increases the sample efficiency because it simplifies the integration of the trajectory from Equation (1). For straight, constant velocity flows, the neural ODE can be solved in a single step. It is easy to show [21] that the FMPE objective of equation (3) does not lead to straight flows, which is also visualized in Figure 1. This can be related to the fact that θ_0 and θ_1 are sampled independently [28].

Mini-batch Optimal Couplings For unconditional data generation tasks, e.g. images, with $\theta_0 \sim q_0$ and $\theta_1 \sim q_1$, it can be shown that sampling any coupling $\gamma(\theta_0, \theta_1)$ with marginal distributions q_0 and q_1 in the flow matching loss will lead to a vector field that maps q_0 to q_1 and only affects the trajectories [28, 29]. Further, [28, 29] show that choosing a coupling that is optimal with respect to a Wasserstein-p distance

$$W_p(p_0, p_1) = \min_{\gamma \in \Gamma(q_0, q_1)} \left[\mathbb{E}_\gamma d(\theta_0, \theta_1)^p\right]^{1/p}, \tag{4}$$

where d is a distance, leads to straight, constant velocity flows. We provide more details regarding the connection of optimal transport and straight flows in Appendix A. Finding such a coupling is generally intractable. A more feasible approach suggested by [28] is to solve the optimal transport problem of Equation (4) for discretely sampled mini-batches. Let $\theta_0^{(i)} \sim q_0$ and $\theta_1^{(i)} \sim q_1$ ($i \in \{1, \ldots, n\}$, the optimal transport problem of equation (4) is then fully defined by the cost matrix

$$C_{ij} = d\left(\theta_0^{(i)}, \theta_1^{(j)}\right). \tag{5}$$

The exact solution can be found with standard solvers, e.g. as implemented in POT [40], with an overall runtime complexity of $\mathcal{O}(n^3)$ [41] or an approximate solution can be found via entropic regularization and Sinkhorn's algorithm with a runtime complexity of $\mathcal{O}(n^2)$ [42]. The solution is given in the form of a permutation matrix, which is a type of doubly-stochastic matrix [28]. For most practical applications, the solution is not degenerate, i.e. there is only a single unique coupling that minimizes the total cost and one can simply rearrange the pairings $(\theta_0^{(i)}, \theta_1^{(j)})$ such that they attain minimal cost. For unconditional generation tasks, i.e. ignoring the x dependency in the FMPE loss (3), this is easy to do during the training and significantly improves the straightness of the trajectories of the vector field [28, 29]. However, adding the condition on the observed data x is non-trivial. In SBI we usually have only a single parameter vector θ per condition (observation x), making it impossible to apply mini-batch optimal transport per observation, i.e. reordering parameter pairs $(\theta_0^{(i)}, \theta_1^{(j)})$ that share the same condition x. Therefore, we propose to add an additional term to the cost matrix of Equation (5) weighted by the conditional weight β

$$C_{ij} = d\left(\theta_0^{(i)}, \theta_1^{(j)}\right) + \beta d\left(s^{(i)}, s^{(j)}\right), \tag{6}$$

where s can be the observation x, a summary statistics of x or the parameter θ_1 itself. For $s = x$, we call it observation conditional optional transport (OCOT), for $s = f(x)$ summary conditional optimal transport (SCOT) and for $s = \theta_1$ parameter conditional optimal transport (PCOT). The idea

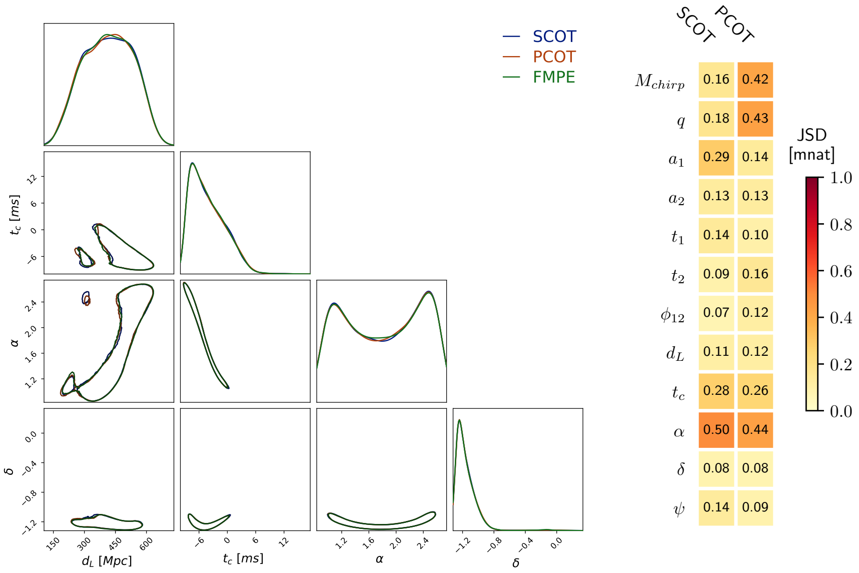

Figure 2: Example 2D contours of a gravitational wave parameter inference. The Jensen Shannon divergence (JSD) is calculated with the 1D marginals with respect to the standard FMPE results.

is to reorder pairs of initial and final parameters $(\theta_0^{(i)}, \theta_1^{(j)})$ only if the corresponding $d(s^{(i)}, s^{(j)})$ is small. We analyse the theoretical properties of this extension in Appendix B and show that SCOT and OCOT can lead to straight, constant velocity flows in the limit of $\beta \to \infty$ and batch size $n \to \infty$.

The distance d can also affect the efficiency of the proposed methods similarly to the distances used in approximate Bayesian computation (ABC) [43]. However, we leave the examination of these effects to future work and focus only on the standard Euclidean distance here.

Choosing the correct conditional weight β can be difficult as it is theoretically unbounded. Therefore, we also introduce the match rate $0 \leq m \leq 1$, which is the fraction of samples in a batch that is not reordered after solving the optimal transport problem. Setting m is more intuitive and translates better between tasks. How the match rate is enforced during training is explained in Appendix C.1.

3 Experiments

Toy Models We use the Two Moons and SLCP Distractors tasks from the simulation-based inference benchmarks [39] to evaluate our methods and perform ablation studies. The detailed results are presented in Appendix C. We show examples of trajectories and posterior samples for the Two Moons task in Figure 1. It can be seen that OCOT straightens the trajectories and recovers the correct posterior while the naive application of mini-batch optimal transport fails. We find similar results for the SLCP Distractors task (see Appendix C).

Graviational Wave Inference To demonstrate the performance of our methods in real-world scenarios, we reproduce the gravitational wave source parameter inference ($\dim(x) = 15,744$) from [26] using their publicly available code and compared the standard reference results to runs with SCOT and PCOT using a matching rate of $m = 0.5$. Because of the high dimensionality of the observation, we decided to not run OCOT. For SCOT we use the 2048-dimensional embedding of the embedding network that was used in their approach. The only other change was that we reduced the batch size from 4096 to 2048. It should be noted that reducing the batch size by a factor of two increases the training time as well. We solve the optimal transport problem of Equation (6) using entropic regularization and Sinkhorn's algorithm that can be run on GPUs and does not significantly affect the training time. We show example 2D posteriors and the Jensen Shanon divergence between the reference run and our methods in Figure 2. We find that the contours agree extremely well between all methods with the JSD being below 10^{-3} nat for all 1D distributions. Additionally, using the adaptive solver `dopri5` from the `torchdiffeq` package [44], both PCOT and SCOT reduce the number of required network evaluations by 20% when sampling. If the probability of the posterior samples is also computed (using Equation (2) and requires the divergence of the vector field) the number of function evaluations is reduced by 16%. This shows that PCOT and SCOT can be applied to high-dimensional problems and effectively reduce the number of network evaluations without degrading the performance of FMPE. Additional details are provided in Appendix D.

4 Discussion & Conclusion

Flow matching is a powerful tool in SBI, offering strong results in posterior estimation but with higher inference times. We introduced three mini-batch optimal coupling methods to address this, adding a term to the cost matrix weighted by the hyperparameter β. These methods generally enhance vector field straightness without much performance loss, though their effectiveness varies by task and hyperparameter choice. Future work could explore these methods in the sequential version of FMPE or apply them to other generative tasks like image generation.

Acknowledgments and Disclosure of Funding

This work was supported by the Swiss National Science Foundation (project number CRSII5_213497 - PI: Fragkos). We thank Jonathan Gair for helpful discussions.

References

[1] R. van de Schoot, S. Depaoli, R. King, B. Kramer, K. Märtens, M. G. Tadesse, M. Vannucci, A. Gelman, D. Veen, J. Willemsen, and C. Yau, "Bayesian statistics and modelling," *Nature Reviews Methods Primers*, vol. 1, p. 1, Jan 2021.

[2] K. Cranmer, J. Brehmer, and G. Louppe, "The frontier of simulation-based inference," *Proceedings of the National Academy of Science*, vol. 117, pp. 30055–30062, Dec. 2020.

[3] M. A. Beaumont, W. Zhang, and D. J. Balding, "Approximate Bayesian Computation in Population Genetics," *Genetics*, vol. 162, pp. 2025–2035, 12 2002.

[4] J.-M. Marin, P. Pudlo, C. P. Robert, and R. Ryder, "Approximate Bayesian Computational methods," *arXiv e-prints*, p. arXiv:1101.0955, Jan. 2011.

[5] S. N. Wood, "Statistical inference for noisy nonlinear ecological dynamic systems," *Nature*, vol. 466, no. 7310, pp. 1102–1104, 2010.

[6] A. L. L. F. Price, C. C. Drovandi and D. J. Nott, "Bayesian synthetic likelihood," *Journal of Computational and Graphical Statistics*, vol. 27, no. 1, pp. 1–11, 2018.

[7] G. Papamakarios, "Neural Density Estimation and Likelihood-free Inference," *arXiv e-prints*, p. arXiv:1910.13233, Oct. 2019.

[8] J.-M. Lueckmann, G. Bassetto, T. Karaletsos, and J. H. Macke, "Likelihood-free inference with emulator networks," *arXiv e-prints*, p. arXiv:1805.09294, May 2018.

[9] O. Thomas, R. Dutta, J. Corander, S. Kaski, and M. U. Gutmann, "Likelihood-free inference by ratio estimation," *arXiv e-prints*, p. arXiv:1611.10242, Nov. 2016.

[10] J. Zeghal, F. Lanusse, A. Boucaud, B. Remy, and E. Aubourg, "Neural posterior estimation with differentiable simulators," *arXiv preprint arXiv:2207.05636*, 2022.

[11] S. Wiqvist, J. Frellsen, and U. Picchini, "Sequential neural posterior and likelihood approximation," *arXiv preprint arXiv:2102.06522*, 2021.

[12] D. Greenberg, M. Nonnenmacher, and J. Macke, "Automatic posterior transformation for likelihood-free inference," in *Proceedings of the 36th International Conference on Machine Learning* (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of *Proceedings of Machine Learning Research*, pp. 2404–2414, PMLR, 09–15 Jun 2019.

[13] Y. Xiong, X. Yang, S. Zhang, and Z. He, "An efficient likelihood-free bayesian inference method based on sequential neural posterior estimation," *arXiv preprint arXiv:2311.12530*, 2023.

[14] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference," *J. Mach. Learn. Res.*, vol. 22, jan 2021.

[15] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 2256–2265, PMLR, 07–09 Jul 2015.

[16] Y. Song and S. Ermon, "Generative Modeling by Estimating Gradients of the Data Distribution," *arXiv e-prints*, p. arXiv:1907.05600, July 2019.

[17] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 6840–6851, Curran Associates, Inc., 2020.

[18] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever, "Consistency models," *arXiv preprint arXiv:2303.01469*, 2023.

[19] J. Kohler, A. Pumarola, E. Schönfeld, A. Sanakoyeu, R. Sumbaly, P. Vajda, and A. Thabet, "Imagine flash: Accelerating emu diffusion models with backward distillation," *arXiv preprint arXiv:2405.05224*, 2024.

[20] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, “Flow matching for generative modeling,” in *The Eleventh International Conference on Learning Representations*, 2023.

[21] X. Liu, C. Gong, and qiang liu, “Flow straight and fast: Learning to generate and transfer data with rectified flow,” in *The Eleventh International Conference on Learning Representations*, 2023.

[22] J. Simons, L. Sharrock, S. Liu, and M. Beaumont, “Neural score estimation: Likelihood-free inference with conditional score based diffusion models,” in *Fifth Symposium on Advances in Approximate Bayesian Inference*, 2023.

[23] T. Geffner, G. Papamakarios, and A. Mnih, “Compositional score modeling for simulation-based inference,” in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 11098–11116, PMLR, 23–29 Jul 2023.

[24] M. Gloeckler, M. Deistler, C. Weilbach, F. Wood, and J. H. Macke, “All-in-one simulation-based inference,” *arXiv e-prints*, p. arXiv:2404.09636, Apr. 2024.

[25] M. Schmitt, V. Pratz, U. Köthe, P.-C. Bürkner, and S. T. Radev, “Consistency Models for Scalable and Fast Simulation-Based Inference,” *arXiv e-prints*, p. arXiv:2312.05440, Dec. 2023.

[26] J. B. Wildberger, M. Dax, S. Buchholz, S. R. Green, J. H. Macke, and B. Schölkopf, “Flow matching for scalable simulation-based inference,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[27] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, “Neural ordinary differential equations,” in *Advances in Neural Information Processing Systems* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, Curran Associates, Inc., 2018.

[28] A.-A. Pooladian, H. Ben-Hamu, C. Domingo-Enrich, B. Amos, Y. Lipman, and R. T. Q. Chen, “Multisample flow matching: Straightening flows with minibatch couplings,” in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 28100–28127, PMLR, 23–29 Jul 2023.

[29] 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,” *Transactions on Machine Learning Research*, 2024. Expert Certification.

[30] N. Kornilov, A. Gasnikov, and A. Korotin, “Optimal flow matching: Learning straight trajectories in just one step,” *arXiv preprint arXiv:2403.13117*, 2024.

[31] A. Tong, J. Huang, G. Wolf, D. Van Dijk, and S. Krishnaswamy, “TrajectoryNet: A dynamic optimal transport network for modeling cellular dynamics,” in *Proceedings of the 37th International Conference on Machine Learning* (H. D. III and A. Singh, eds.), vol. 119 of *Proceedings of Machine Learning Research*, pp. 9526–9536, PMLR, 13–18 Jul 2020.

[32] C. Finlay, J.-H. Jacobsen, L. Nurbekyan, and A. Oberman, “How to train your neural ode: the world of jacobian and kinetic regularization,” in *International conference on machine learning*, pp. 3154–3164, PMLR, 2020.

[33] D. Onken, S. W. Fung, X. Li, and L. Ruthotto, “Ot-flow: Fast and accurate continuous normalizing flows via optimal transport,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 9223–9232, 2021.

[34] A. Asadulaev, A. Korotin, V. Egiazarian, P. Mokrov, and E. Burnaev, “Neural optimal transport with general cost functionals,” *arXiv preprint arXiv:2205.15403*, 2022.

[35] A. Korotin, D. Selikhanovych, and E. Burnaev, “Neural optimal transport,” *arXiv preprint arXiv:2201.12220*, 2022.

[36] A. Makkuva, A. Taghvaei, S. Oh, and J. Lee, "Optimal transport mapping via input convex neural networks," in *International Conference on Machine Learning*, pp. 6672–6681, PMLR, 2020.

[37] C. Finlay, A. Gerolin, A. M. Oberman, and A.-A. Pooladian, "Learning normalizing flows from entropy-kantorovich potentials," *arXiv preprint arXiv:2006.06033*, 2020.

[38] J. Fan, S. Liu, S. Ma, Y. Chen, and H.-M. Zhou, "Scalable computation of monge maps with general costs," in *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022.

[39] J.-M. Lueckmann, J. Boelts, D. Greenberg, P. Goncalves, and J. Macke, "Benchmarking simulation-based inference," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics* (A. Banerjee and K. Fukumizu, eds.), vol. 130 of *Proceedings of Machine Learning Research*, pp. 343–351, PMLR, 13–15 Apr 2021.

[40] R. Flamary, N. Courty, A. Gramfort, M. Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, L. Gautheron, N. T. Gayraud, H. Janati, A. Rakotomamonjy, I. Redko, A. Rolet, A. Schutz, V. Seguy, D. J. Sutherland, R. Tavenard, A. Tong, and T. Vayer, "Pot: Python optimal transport," *Journal of Machine Learning Research*, vol. 22, no. 78, pp. 1–8, 2021.

[41] G. Peyré and M. Cuturi, "Computational optimal transport: With applications to data science," *Foundations and Trends® in Machine Learning*, vol. 11, no. 5-6, pp. 355–607, 2019.

[42] J. Altschuler, J. Weed, and P. Rigollet, "Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration," *arXiv e-prints*, p. arXiv:1705.09634, May 2017.

[43] Y. Fan and S. A. Sisson, "ABC Samplers," *arXiv e-prints*, p. arXiv:1802.09650, Feb. 2018.

[44] R. T. Q. Chen, "torchdiffeq," 2018.

[45] A. Figalli, F. Glaudo, and E. M. S. P. H. E.-Z. S. A27, *An Invitation to Optimal Transport, Wasserstein Distances, and Gradient Flows*. EMS textbooks in mathematics, EMS Press, 2021.

[46] J.-D. Benamou and Y. Brenier, "A computational fluid mechanics solution to the monge-kantorovich mass transfer problem," *Numerische Mathematik*, vol. 84, pp. 375–393, Jan 2000.

[47] B. O. Koopman, "On distributions admitting a sufficient statistic," *Transactions of the American Mathematical Society*, vol. 39, pp. 399–409, 1936.

[48] Y. Chen, D. Zhang, M. Gutmann, A. Courville, and Z. Zhu, "Neural Approximate Sufficient Statistics for Implicit Models," *arXiv e-prints*, p. arXiv:2010.10079, Oct. 2020.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv e-prints*, p. arXiv:1512.03385, Dec. 2015.

[50] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[51] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, (San Diega, CA, USA), 2015.

[52] J. H. Friedman, "On multivariate goodness of fit and two sample testing," *eConf*, vol. C030908, p. THPD002, 2003.

[53] D. Lopez-Paz and M. Oquab, "Revisiting Classifier Two-Sample Tests," *arXiv e-prints*, p. arXiv:1610.06545, Oct. 2016.

[54] B. Farr, E. Ochsner, W. M. Farr, and R. O'Shaughnessy, "A more effective coordinate system for parameter estimation of precessing compact binaries from gravitational waves," *Phys. Rev. D*, vol. 90, p. 024018, Jul 2014.

A Optimal Transport and Straight Flows

In this Appendix, we provided a more detailed description of the connection between optimal transport and straight flows. We start with a formal definition of straightness. For a fixed observation x, the straightness of a flow is defined as [28]

$$S_x = \mathbb{E}_{t\sim\lambda(t),\theta_0\sim q_0}\left[\|v_{t,x}(\theta_t)\|^2 - \|\theta_1 - \theta_0\|^2\right]. \tag{7}$$

It can be shown [28] that $S \geq 0$ and $S = 0$ only if $v_{t,x}(\theta_t)$ is constant along t, indicating that the trajectory of θ_t with initial condition θ_0 is a straight line. Straightness increases the sample efficiency because it simplifies the integration of the trajectory from equation (1). Theoretically, for a perfectly straight flow ($S_x = 0$), the integration can be performed in a single step.

Intuitively, the FMPE objective function can not lead to straight flow because it is an expected value over independently drawn samples θ_0 and θ_1. In the objective function, these two independent samples are connected via a straight line, leading to many overlapping trajectories. However, the learned vector field $v_{t,x}$ is deterministic and can only go into a single direction from any given point, meaning that intersection points are impossible. Instead, the resulting vector field takes the expected direction of all trajectories in an intersection point of the training data [21] which causes the curvature.

Optimal Transport Flows The definition of the Wasserstein-p distance from Equation (4) is a special case of the general optimal transport problem, where the distance d can be replaced with an arbitrary cost function c. Using the Euclidean distance, it can be shown that the optimal coupling γ^* is unique for $p > 1$ if p_0 and p_1 satisfy some weak regularity assumptions [45]. An optimal coupling induces a constant speed geodesic between the two distributions [45]. Such a geodesic can be thought of as a linear interpolation between the two distributions that connects samples via straight lines. We will focus on the Wasserstein-2 distance, which is one of the most studied and well-understood Wasserstein distances, but other cost functions are possible [28]. For the Wasserstein-2 distance there exists an equivalent formulation based on probability flows that illustrates the concept of constant-speed geodesics and straight flows. As shown in [46], the Wasserstein-2 distance can be written as

$$W_2(p_0, p_1) = \min_{q_t, u_t} \int_0^1 \|u_t(x)\|^2 q_t(x)\mathrm{d}x\mathrm{d}t, \tag{8}$$

where u_t is a flow generating q_t and q_t has boundary conditions $q_{t=0} = p_0$ and $q_{t=1} = p_1$. The optimal q_t of equation (8) is a constant speed geodesic of p_0 and p_1 and the corresponding optimal vector field u_t is constant along the trajectories. Sampling from Wasserstein-2-optimal couplings would lead to pairs (θ_0, θ_1) that almost surely do not intersect. Using such a coupling in our loss function instead of the classic independent coupling will therefore create straight, constant-velocity flows [28, 29].

B Theoretical Analysis

This Appendix contains a theoretical analysis of the proposed extensions to mini-batch optimal transport. The additional term in the cost matrix of Equation (6) depends either on the observation x, a summary of the observation $f(x)$ or the underlying model parameter θ. Before we go into the theoretical properties of the extension, we will first explain the notion of the summary used in the extension in more detail. Summaries, sometimes just called summary statistics, are commonly used in SBI. The goal of summary statistics is to reduce the dimensionality of the observations x while preserving the relevant information about the model parameters θ. The summary statistics can be expressed in terms of a function acting on the observation $f(x)$, where f can by any function, e.g. a neural network. A summary statistics is called sufficient if it preserves all the information about model parameters. A sufficient summary leads to the same posterior distribution as the original observation $p(\theta|f(x)) = p(\theta|x)$. Note that such a sufficient summary statistics does not always exist [47]. Good summary statistics can be obtained for example via domain experts, training a compression network along with the flow matching objective or pre-training a network with a specialized loss [48]. In this work, we will only consider training a compression network alongside the FMPE objective, e.g. as in the gravitational wave source parameter inference of the main paper.

For unconditional generation, it can be shown that the optimal solution of the flow matching objective is a straight flow as the batch size goes towards infinity [28]. Using the notion of a sufficient summary statistics, we can generalize the informal theorem 4.2 of [28] for conditional generation.

Theorem 1 (Informal). *Let s be a sufficient summary statistics of x with model parameters θ. Suppose FMPE is run with SCOP with batch size n and conditional weight β in the cost matrix. Then as $n, \beta \to \infty$,*

(i) The value of the objective function (equation (3)*for the optimal vector field $v_{t,x}$ converges to zero.*

(ii) The straightness S_x (equation (7)*) of the optimal vector field converges to zero.*

Proof Sketch: The proof relies on the ABC approximation of the posterior distribution [43]

$$p_{\mathrm{ABC}}(\theta|s_o) \propto \int K_h\left(\|s - s_0\|\right) p(s|\theta) p(\theta) \mathrm{d}s,$$

where $K_h(x) = K(x/h)/h$ is a kernel density function with scale parameter h. The ABC posterior converges to the true posterior for $h \to 0$ if s is sufficient. Using the constant base kernel ($K(x) = 1/2$ if $|x| \leq 1$ and 0 otherwise), we see that we can obtain the true posterior by integrating over summaries that are close to our summary of interest s_0 with respect to the norm $\|\cdot\|$. It follows that as $\beta \to \infty$, SCOP will only reorder triplets $\theta_0^{(i)}, \theta_1^{(i)}, s^{(i)}$ that have the same posterior distribution. The rest then follows from the proof for unconditional generation provided in [28].

This shows the potential of SCOT (and OCOT) to generate straight flows for SBI applications. It is also possible to apply the same arguments to the other benefits of mini-batch optimal couplings presented in [28], such as a reduced variance of the gradients and a faster convergence. Of course, in practical applications with finite batch sizes, setting large values for β will result in classical FMPE and all these benefits are lost.

It should be noted that the heuristic of PCOT does not lead to straight flows or even the correct posterior for $n, \beta \to \infty$. The reason is that observation with the same underlying parameter will generally not create the same posterior distribution. Therefore, as $n, \beta \to \infty$, PCOT will not act on triplets with the same posterior. We nevertheless investigate PCOT as a heuristic for practical applications with finite batch sizes.

C Experiments on the SBI Benchmarks

This appendix contains our experiments on the SBI benchmark tasks Two Moons and SLCP Distractors. We provide several results that show how the proposed methods improve the straightness of the trajectories. We chose the Two Moons task with $\dim(\theta) = \dim(x) = 2$ because it is easy to visualize and fast to train. The slightly more difficult SLCP Distractors task with $\dim(\theta) = 5$ and $\dim(x) = 100$ was chosen because the dimensionality of the observation is higher than the dimensionality of the model parameters such that it is possible to compress the observation for SCOP.

C.1 Training and Evaluation

We created 100,000 simulations for both tasks and used 5% for validation during the training. The models parametrizing the vector fields in this work are fully connected residual networks [49] with three residual blocks and a hidden dimension of 256. Our default batch size is 1024 and we train for 100 epochs using the cosine annealing schedule [50] and the ADAM optimizer [51] with an initial learning rate of 10^{-3} and default moments. All models were trained on a single GPU. The dominant part of the training was the evaluation of the C2ST score as provided in the SBI benchmark package which does not have GPU support as of the writing of this paper. We find that this training setup reproduces the results from [26] for standard FMPE, without the need to use GLU conditioning or a re-scaled time prior.

For SCOT we train a fully connected residual embedding network with three residual blocks which reduces the dimensionality of the observation. Motivated by [48] we set the dimensionality of the summary to $\dim(s) = 2\dim(\theta)$. The cost matrix used for SCOT is calculated during each training step after the observations pass through the embedding network. The optimal transport problem is

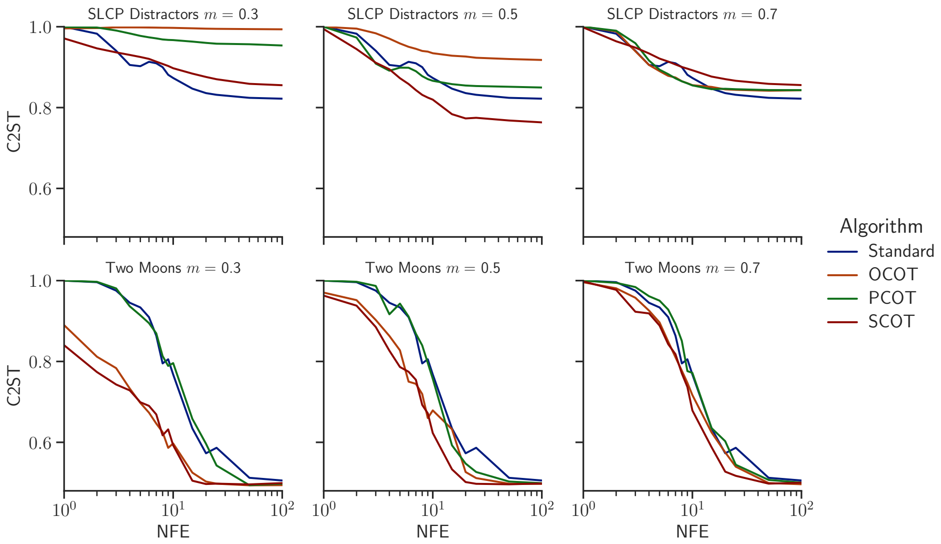

Figure 3: C2ST scores vs evaluation budget for the two tasks and different match rates for a single observation.

solved exactly for all tasks and methods. This is possible since we only consider batch sizes of 1024 or lower for the toy models.

To achieve the correct match rate m we adapt β during each iteration of the training. Starting with $\beta = 1.0$, we calculate the empirical match rate after each batch and increase or decrease β by 10% towards the desired match rate. This way, the match rate stabilizes around the correct value after ~ 20 iterations.

We evaluate the performance of the models in terms of the C2ST score [52, 53]. The C2ST score is the accuracy of a classifier trained to distinguish true (reference) posterior samples and posterior samples generated with the trained model. The score ranges from 0.5 (best) to 1.0 (worst). We sample the posterior of our models by integrating equation (2) via the explicit Euler method. This allows us fix the number of function evaluations (NFE) used to sample the posterior. We also used other evaluation metrics from the SBI benchmarks but found very similar results and trends, such that we decided to omit them here.

The straightness of the trajectories is evaluated by calculating the C2ST score for different NFE used to integrate the neural ODE (Equation (1)) via explicit Euler. Obtaining a better C2ST score with fewer NFE indicates straighter trajectories.

C.2 Impact of the Match Rate

The match rate m (or conditional weight β) is the only hyperparameter of the proposed methods. The match rate is always between 0 and 1, where $m = 0$ corresponds to the naive application of mini-batch optimal couplings that are unaware of the conditional observation and $m = 1$ corresponds to standard FMPE. Choosing a match rate that is too low will diminish the performance (see e.g. Figure 1), while a match rate that is too high will have curved trajectories like standard FMPE. We present the results for the two tasks and three different match rates ($[0.3, 0.5, 0.7]$) in Figure 3. For the simple Two Moons task, OCOT and SCOT perform best for the lowest match rate. PCOT performs like standard FMPE for all match rates. For the SLCP Distractor task, one can see that the lowest match rate diminishes the performance of all extensions. SCOT performs best for $m = 0.5$. For both tasks, all results approach the performance of standard FMPE for $m = 0.7$, as expected for higher match rates.

This shows that the performance of the methods depends strongly on the chosen match rate and that the optimal match rate might differ between tasks. However, SCOT leads to straighter trajectories for both tasks if the match rate is chosen carefully.

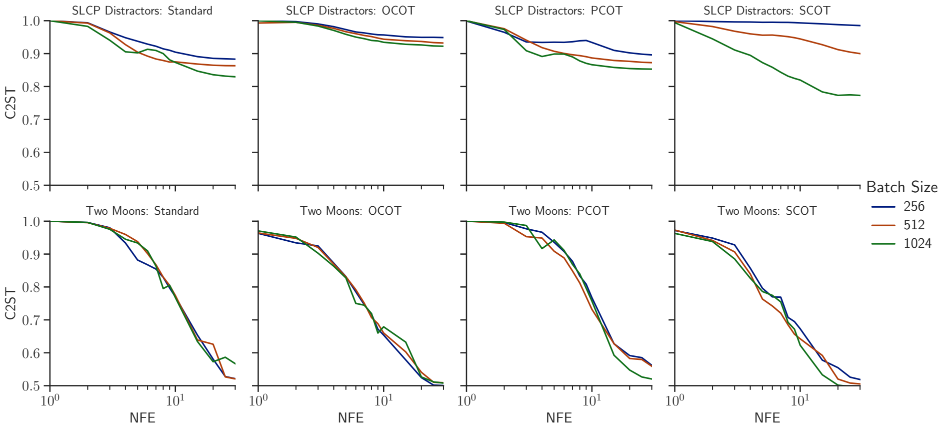

Figure 4: C2ST scores vs evaluation budget for the two tasks and different batch sizes for a single observation and a match rate of $m = 0.5$.

C.3 Impact of the Batch Size

Another important hyperparameter for the proposed methods is the batch size that is used during the training. Theoretically, we would expect the performance of OCOT and SCOT to be better for larger batch sizes. The results are presented in Figure 4. It can be seen that the performance of the simple Two Moons task is very similar for all three examined batch sizes of 256, 512 and 1024. For the SLCP Distractors task, there is a tendency for larger batch sizes to lead to better performance. We expect this tendency to become clearer for even larger batch sizes.

D Graviational Wave Source Parameter Inference

The gravitational source parameter inference was performed in the same way as presented in [26]. We provide a list of inferred parameters and their priors in Table 1. Note that not all of the parameters are constrained, some are derived. For example, component masses of the binary system are not inferred directly but can be calculated from the chirp mass M_{chrip} and the mass ratio q. The example contours shown in Figure 2 are the same as in [26] for the sake of consistency.

Description	Parameter	Prior
component masses	m_1, m_2	$[10, 120]M_{\odot}$, $m_1 > m_2$
chirp mass	$M_{chrip} = (m_1 m_2)^{3/5}/(m_1 + m_2)^{1/5}$	$[20, 120]M_{\odot}$
mass ratio	$q = m_2/m_1$	$0.125, 1.0]$
spin magnitudes	a_1, a_2	$[0, 0.99]$
spin angles	$\theta_1, \theta_2, \phi_{12}, \phi_{JL}$	standard as in [54]
time of coalescence	t_c	$[-0.03, 0.03]$ s
luminosity distance	d_L	$[100, 1000]$ Mpc
reference phase	ϕ_c	$[0, 2\pi]$
inclination	θ_{JN}	$[0, \pi]$ uniform in sine
polarization	ψ	$[0, \pi]$
sky position	α, β	uniform over sky

Table 1: The parameters used in the gravitational wave source parameter inference.

Finally, it is interesting to see that PCOT improves the straightness for the high-dimensional gravitational source parameter inference but not for the two toy models. We believe that this is partly caused by the larger batch size of 2048 and partly task-dependent because PCOT is a simple heuristic that is not well-suited for all inference tasks.