FIXED-POINT ACCELERATION ON NEURAL WASSERSTEIN MANIFOLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a fixed-point acceleration method for optimization over probability measures, applicable to a variety of machine learning problems. We represent probability measures as pushforwards of a reference distribution through a neural network map and equip the parameter space with the induced pullback Wasserstein metric. This geometric structure allows natural gradient descent to be interpreted as a fixed-point iteration on a Riemannian manifold. We accelerate this iteration using Riemannian Anderson mixing method, which exploits historical residuals to capture approximate second-order information without forming Hessians. In the present parametric setting, the manifold is represented by Euclidean coordinates endowed with a point-dependent metric, simplifying retraction and vector transport. We evaluate the method on multimodal sampling, PDE-based inverse problems, aggregation dynamics, Bayesian logistic regression, and generative modeling. Across the tested tasks, Anderson mixing reduces the number of natural-gradient iterations and achieves competitive performance relative to SGD and Adam in terms of both loss-gradient evaluations and wall-clock time. These results suggest that fixed-point acceleration is an efficient tool for improving the computational efficiency of training on parametric Wasserstein methods

1 INTRODUCTION

Many machine-learning problems can be formulated as optimization problems over probability distributions. Examples include variational inference, posterior approximation, density estimation, generative modeling, and the computation of equilibrium distributions in interacting-particle models. A mathematical framework for the minimization over probability measures is the Wasserstein gradient flow theory Ambrosio et al. (2005). Many relevant evolution equations — including Fokker-Planck Jordan et al. (1998), porous medium Otto (2001), and aggregation-diffusion equations Carrillo et al. (2006) — admit a gradient flow structure if the space is endowed with the Wasserstein distance Ambrosio et al. (2005); Villani (2021). In this formulation, the equation describes steepest descent of a free energy functional $\mathcal{E}$, and the long-time behavior of the flow converges to its minimizer — the stationary distribution. Moreover, the evolution of the gradient flow can be seen as a strictly contractive operator (Ambrosio et al., 2005, Theorem 11.2.1), i.e.

$$W_2(S[\mu_1](h), S[\mu_2](h)) \leq e^{-h\lambda} W_2(\mu_1, \mu_2), \tag{1.0.1}$$

with $S[\mu](h)$ being the semigroup of the gradient flow, μ initial distribution, h time step and the functional is $0 < \lambda$-convex along generalized geodesics.

From the optimization point of view, the primary object of interest is the minimizer, in other words, the stationary distribution or fixed-point of the evolution operator, and not the trajectory of the dynamical system. Examples are the minimization of the Kullback-Leibler Kullback & Leibler (1951) divergence for sampling in Bayesian Inverse problems and MMD Dziugaite et al. (2015); Arbel et al. (2019) or cross-entropy functionals in generative modeling Goodfellow et al. (2014).

A common strategy is to represent candidate distributions as pushforwards of a tractable reference measure, e.g., a Gaussian distribution, through a parametric map T_θ, e.g. a normalizing flow Papamakarios et al. (2021) or a neural ODE Chen et al. (2018). This reduces an infinite-dimensional optimization problem over probability measures to a finite-dimensional optimization problem over parameters. The Euclidean geometry of the parameter space is generally not aligned with the geometry

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

of the represented probability measures. Parametric Wasserstein methods address this mismatch by equipping the parameter space with the pullback Wasserstein metric induced by the pushforward map. The resulting metric defines a natural-gradient direction for optimizing distributional objectives and can be interpreted as the projection of the Wasserstein gradient onto the tangent directions generated by the model Liu et al. (2022); Jin et al. (2025); Wu et al. (2025).

The idea of using a pullback geometry, natural for the geometry of the infinite-dimensional space that is being modeled, has been applied in a broad array of papers, starting with Amari (1998), and has recently been actively applied in Scientific ML. The applications include methods based on the Dirac-Frenkel variational principle Du & Zaki (2021); Bruna et al. (2024); Wu & Zhou (2025), natural gradients induced by Hessian, Sobolev, or energy-based metrics Nurbekyan et al. (2023); Müller & Zeinhofer (2023); Chen et al. (2024), primal-dual natural gradient formulations Liu et al. (2024b), and accelerated natural gradient frameworks accommodating multiple metric structures Li et al. (2025). In the context of image generation, in Shen et al. the authors derived a Sinkhorn natural gradient. Substantially improvement of the accuracy and convergence speed of neural network-based PDE solvers were reported in these works.

In the literature, acceleration techniques in continuous time beyond steepest descent have been studied, with Hessian-damped Riemannian flows Li et al. (2025), accelerated Stein gradient flow Stein & Li (2025) and Hamiltonian flows Chen et al. (2025), to name but a few. The common drawback of the dynamic approaches is that the theoretical rates of convergence in the continuous time do not directly translate into improved iteration complexity due to time integration complexity. In **??** we provide a comparison with the two closest methods Aksenov et al. (2026); Li et al. (2025)

Contribution: We propose to treat parametric Wasserstein natural-gradient descent as a nonlinear fixed-point iteration and accelerate convergence using Riemannian Anderson Mixing (RAM) method. The fixed-points of Φ_h are exactly the Riemannian stationary points, and the natural-gradient method is the Picard iteration $\theta_{k+1} = \Phi_h(\theta_k)$. Since the geometry in the space of the parameters is $\mathbb{R}^D$ with a point-dependent scalar product, under local nondegeneracy assumptions, we show that it suffices to use Euclidean retraction and vector transport for the RAM .

We evaluate our approach numerically on an array of machine-learning problems. To put our method in context, we compare its performance with standard optimizers such as SGD and Adam, and demonstrate that they can still be outperformed for certain problems.

Limitations: Just as in NGD, we rely on solving a linear system in the space of the parameters to acquire the natural gradient. Additional memory storage of $2m$ times the number of the parameters is required. Though small a small number of historical vectors ($m = 2$-8) is enough to provide acceleration, for very large models this can still be prohibitive . Moreover, while we outperform NGD in most cases, the speedup can be minor for a bad choice of method's hyperparameters. While we give some advice on their selection, one has to note these are only heuristic. From the theoretical perspective, only a local residual decrease estimate has been shown for RAM, and not an explicit convergence rate. However, extensive numerical evaluation on a diverse variety of problems that we perform is in favor of RAM.

Outline: The paper is organized as follows. section 2 summarizes the pullback Wasserstein geometry. section 3 introduces Riemannian Anderson mixing, stating its local convergence estimate and relating it to the minimization error in the original problem. In section 4 we apply RAM to minimize KL divergence to a posterior density, including both synthetic targets and a PDE-based Bayesian inverse problem; Bayesian Logistic regression and MMD in context of generative modeling.

2 Riemannian geometry of probabilistic problems

The Wasserstein metric on probability measures stems from the optimal transport problems of Monge Monge (1781) and Kantorovich Kantorovich (1942). Unlike Hellinger or Total Variation Gibbs & Su (2002), it accounts for the geometry of the underlying space, leading to applications in supervised learning and domain adaptation Courty et al. (2016), dictionary learning Schmitz et al. (2018), NLP Xu et al. (2018), graphs Vayer et al. (2018), and generative modeling Arjovsky et al. (2017). The Benamou–Brenier formula allows to interpret it as a geodesic metric on a Riemannian

manifold:

$$W_2^2(\rho_0,\rho_1)=\min_{\rho_t,\,v_t}\int\frac{1}{2}\|v_t\|^2_{L_2(\rho_t)}\mathrm{d}t,\quad \text{s.t.}\ \begin{array}{c}\partial_t\rho_t+\nabla(\rho_t v_t)=0,\\ \rho_t|_{t=0}=\rho_0,\quad \rho_t|_{t=1}=\rho_1.\end{array}$$

Here, admissible vector fields $v_t \in L_2(\rho_t)$ satisfying the continuity equation are interpreted as tangent vectors to the curve ρ_t, while the $L_2(\rho_t)$ inner product plays the role of a metric tensor. The steepest descent direction of a functional $\mathcal{E}$, called the Wasserstein gradient, is given by

$$\partial_W\mathcal{E}(\rho)=\nabla_x\frac{\delta\mathcal{E}}{\delta\rho}\in L_2(\rho),$$

where $\frac{\delta}{\delta\rho}$ denotes the first variation. Other Riemannian-like constructions, including exponential maps Gigli (2008) and parallel transport Ambrosio & Gigli (2008), can also be defined. This geometric structure motivates adapting Riemannian optimization methods to minimization problems on Wasserstein space.

2.1 Natural gradient on the parametric space

Let ρ_{ref} be a fixed reference measure on $\mathbb{R}^d$ and let $T_\theta:\mathbb{R}^d\to\mathbb{R}^d$ be a differentiable map parameterized by $\theta\in\Theta\subset\mathbb{R}^D$. We consider the parametric family

$$\mathcal{P}_\Theta\coloneqq\{(T_\theta)_\sharp\rho_{\text{ref}}:\theta\in\Theta\}\subset\mathcal{P}_2(\mathbb{R}^d). \tag{2.1.1}$$

Given an energy functional $\mathcal{E}:\mathcal{P}_2(\mathbb{R}^d)\to\mathbb{R}$, we optimize its restriction to this family,

$$\theta^*\in\arg\min_{\theta\in\Theta}E(\theta),\qquad E(\theta)\coloneqq\mathcal{E}((T_\theta)_\sharp\rho_{\text{ref}}). \tag{2.1.2}$$

The geometry in the parameter space is defined by the point-dependent inner product

$$\langle u,v\rangle_\theta\coloneqq u^\top G(\theta)v,\qquad \|u\|^2_\theta\coloneqq u^\top G(\theta)u, \tag{2.1.3}$$

with the entries of the metric given by

$$G(\theta)_{ij}=\sum_{\ell=1}^{d}\mathbb{E}_{z\sim\rho_{\text{ref}}}\left[\partial_{\theta_i}T^\ell_\theta(z)\,\partial_{\theta_j}T^\ell_\theta(z)\right]. \tag{2.1.4}$$

The corresponding parametric Wasserstein natural gradient is

$$\operatorname{grad}_G E(\theta)=G(\theta)^\dagger\nabla_\theta E(\theta), \tag{2.1.5}$$

where $G(\theta)^\dagger$ denotes the Moore–Penrose pseudoinverse.

2.2 Fixed-point iteration on $\mathcal{P}_\Theta$

Every minimizer of the functional $\mathcal{E}$ restricted to the parametric submanifold $\mathcal{P}_\Theta$, assuming nondegeneracy of $G(\theta)$, satisfies the first-order condition

$$\operatorname{grad}_G E(\theta)=G(\theta)^\dagger\nabla_\theta E(\theta)=0.$$

Equivalently, for any step size $h>0$, a stationary point is a fixed point of the operator

$$\Phi_h(\theta)\coloneqq\theta-hG(\theta)^\dagger\nabla_\theta E(\theta). \tag{2.2.1}$$

The corresponding Picard iteration is $\theta_{k+1}=\Phi_h(\theta_k)$ and the zeros of the residual $r_k\coloneqq\Phi_h(\theta_k)-\theta_k$ are precisely Riemannian stationary points of the parametric objective.

Under the following standard assumptions on the functional and on the approximation of the Wasserstein gradient by the parametric tangent space, decay of the residual can be related to stationarity of the original Wasserstein problem.

Assumption 2.1 (Gradient-projection error (Jin et al., 2025, Definition 3.6)). *For every θ assume that*

$$\delta_0\coloneqq\sup_{\theta\in\Theta}\inf_{\xi\in\mathbb{R}^D}\left(\int|\partial_W\mathcal{E}(\rho_\theta)(T_\theta(z))-\partial_\theta T_\theta(z)\xi|^2\,\mathrm{d}\rho_{ref}(z)\right)^{1/2}<\infty, \tag{2.2.2}$$

where $\rho_\theta=(T_\theta)_\sharp\rho_{ref}$.

Assumption 2.2 (Polyak–Łojasiewicz condition)**.** *There exists $\mu > 0$ such that*

$$\mathcal{E}(\rho) - \mathcal{E}(\rho_*) \leq \frac{1}{\mu} \|\partial_W \mathcal{E}(\rho)\|^2_{L^2(\rho)}, \qquad \forall \rho \in \mathcal{P}_2(\mathbb{R}^d). \tag{2.2.3}$$

PŁis a standard assumption in the analysis of gradient-based optimization methods Karimi et al. (2016); Zhu & Chen (2025); Boufadène & Vialard. In optimal transport, when $\mathcal{E} = \mathrm{KL}(\cdot|\rho_\infty)$, it corresponds to a log-Sobolev inequality (Villani, 2021, Chapter 9.2). For example, it holds if $\rho_\infty \propto \exp(-V - \phi)$, with $V \in C^2(\mathbb{R}^d)$ satisfying $\nabla^2 V \succeq \tilde{\lambda} I$ and ϕ bounded, by the Holley–Stroock perturbation principle Holley & Stroock (1986).

Proposition 2.1 (Residual control of Wasserstein stationarity)**.** *Under Assumption 2.1, the Wasserstein gradient is controlled by the parametric fixed-point residual:*

$$\|\partial_W \mathcal{E}(\rho_\theta)\|^2_{L^2(\rho_\theta)} \leq \frac{1}{h^2} \|r_h(\theta)\|^2_\theta + \delta_0^2. \tag{2.2.4}$$

If, in addition, Assumption 2.2 holds, then

$$\mathcal{E}(\rho_{\theta_k}) - \mathcal{E}(\rho_*) \leq \frac{1}{\lambda} \left(\frac{\|r_h(\theta_k)\|^2_{\theta_k}}{h^2} + \delta_0^2 \right). \tag{2.2.5}$$

Remark 2.1 (Role of the parametrization)**.** *The projection error δ_0 depends on the expressivity of the parametric tangent space generated by $\partial_\theta T_\theta$. Richer pushforward families are expected to reduce this error by representing a larger class of Wasserstein gradient directions. In our experiments, T_θ is the terminal map of a neural ODE. The flexible flow parameterization is well suited for modeling continuous transport maps. The full architecture and solver details are given in Section 4.*

3 Riemannian Anderson Mixing

3.1 Anderson Mixing in Euclidean spaces

In fixed-point problems on $\mathbb{R}^D$, Anderson Mixing (AM) Anderson (1965) is a well-established acceleration method Walker & Ni (2011); Tang et al. (2023). AM stores up to m previous iterates and residuals, and modifies the Picard update using a linear combination of these historical vectors, with coefficients obtained from a small least-squares problem. The history incorporates second-order information: as shown in Fang & Saad (2009), AM is linked to quasi-Newton methods. Toth & Kelley (2015); Evans et al. (2020); Rebholz & Xiao (2023) show that AM performs at least as well as Picard iteration, while convergence rates for related methods were established in Scieur (2024).

Algorithmically, AM only requires scalar products, linear combinations, and a least-squares solve of dimension m, typically 1–20. Its overhead is therefore negligible when residual evaluations are expensive. In practice, AM often reduces iteration complexity Aksenov et al. (2021); Lott et al. (2012); Lipnikov et al. (2013); Brezinski et al. (2022); Qu et al. (2026) and improves robustness Willert et al. (2014). In machine learning, AM has been studied in context of nonsmooth stochastic optimization Wei et al. (2021) and reinforcement learning Sun et al. (2021).

3.2 Extension to Riemannian manifolds

Riemannian Anderson Mixing (RAM) Li & Bao (2023) applies to fixed-point problems of the form

$$\text{Find } x^* \in \mathcal{M} : x^* = \mathrm{Exp}_{x^*}(-F(x^*)) \tag{3.2.1}$$

where $\mathcal{M}$ is a Riemannian manifold, $F : \mathcal{M} \to \mathrm{Tan}\,\mathcal{M}$ is a vector field and Exp is the Riemannian exponential. If $F = \mathrm{grad}\,\mathcal{E}$ for $f : \mathcal{M} \to \mathbb{R}$, the method can be used to minimize $\mathcal{E}$. The authors establish the local convergence and introduce RAM with *adaptive regularization*, converging globally.

To adapt the AM method, following adjustments have to be made. Firstly, the update direction Δx_k "lives" is in $\mathrm{Tan}_x\,\mathcal{M}$, and has to be mapped back to the manifold. Secondly, the historical residuals $r_i \coloneqq -F(x_i)$ and update directions Δx_k have to be mapped to the tangent space of the current iterate. This is done using *retraction* and *vector transport* maps, respectively Absil et al. (2009). They can be viewed as approximations of the Riemannian exponential and parallel transport Absil et al.

Algorithm 1 Riemannian Anderson Mixing on Θ

Parameters: $\theta_0 \in \boldsymbol{M}$, learning rate h, relaxation $\beta_k > 0$, regularization factor λ, $m \in \mathbb{N}$.

1: $r_0 = -hG^\dagger(x\theta_0)\nabla\mathcal{E}(\theta_0)$, $\theta_1 = \theta_0 - hG^\dagger(\theta_0)\nabla\mathcal{E}(\theta_0)$.
2: **for** $k = 1,\ 2, \dots$ until convergence **do**
3: $m_k = \min\{m, k\}$.
4: Compute residual $r_k = -hG^\dagger(\theta_k)\nabla\mathcal{E}(\theta_k)$
5: Assemble matrices of historical vectors

$$X_k = [\theta_i - \theta_{i-1}]\,,\ R_k = [r_i - r_{i-1}]\,,\ i = k, \dots, k - m_k + 1.$$

6: Find mixing coefficients Γ_k

$$\Gamma_k = \arg\min_{\Gamma\in\mathbb{R}^{m_k}} \|r_k - R_k\Gamma\|^2_{\theta_k} + \lambda \begin{cases} \|\Gamma\|_2^2, & (\ell_2 \text{ regularization}) \\ \frac{\|r_k\|^2_{\theta_k}}{\|\theta_k - \theta_{k-1}\|^2_{\theta_k} + \epsilon}\|X_k\Gamma\|^2_{\theta_k}, & (\text{adaptive regularization}) \end{cases}$$

7: $\Delta\theta_k = \beta_k r_k - (X_k + \beta_k R_k)\,\Gamma_k$.
8: $\theta_{k+1} = \theta_k + \Delta\theta_k$ **if** $\langle\Delta\theta_k, r_k\rangle_{\theta_k} > 0$ **else** $\theta_k - hG^\dagger(\theta_k)\nabla\mathcal{E}(\theta_k)$
9: **end for**

(2009). In our parametric setting, the manifold Θ can be interpreted as $\mathbb{R}^D$ with point-dependent scalar product. Our simplified form of RAM is presented in Algorithm 1.

To avoid instability, instead of computing the pseudo-inverse of $G(\theta_k)$, we perform Tikhonov regularization i.e. $r_k = -h\left(G(\theta_k) + \Lambda\,\mathrm{Id}\right)^{-1}\nabla\mathcal{E}(\theta_k)$ for a predefined regularization coefficient Λ. The system is solved iteratively with conjugate gradient (CG) method and matrix-vector products (MVPs) with $G(\theta_k)$, implemented using Hessian-vector products Pearlmutter (1994). As in Wei et al. (2021) we check if $\Delta\theta_k$ is a descent direction, and perform a Picard step otherwise. For adaptive regularization, $\epsilon = 10^{-8}$.

The least-squares minimization subproblem requires computation of scalar products, adding computational overhead. However, it reduces to computing of up to $2m$ MVPs $G(\theta_k)(r_i - r_{i-1})$, $G(\theta_k)(\theta_i - \theta_{i-1})$ and then computing their Euclidean scalar products with corresponding vectors. These MVPs can be computed in parallel, unlike those in conjugate gradient iteration, thus it is reasonable to neglect the overhead. We further investigate this numerically in subsubsection 4.1.3.

Our simplification of RAM is justified by the following statement:

Proposition 3.1. *Let $\theta, \theta' \in \Theta$ and $\Delta\theta \in \mathrm{Tan}_\theta\,\Theta \simeq \mathbb{R}^D$. The mappings*

$$\mathcal{R}_\theta(\Delta\theta) \coloneqq \theta + \Delta\theta, \quad \mathcal{R}_\theta : \mathrm{Tan}_\theta \to \Theta \tag{3.2.2}$$

$$\mathcal{T}_\theta^{\theta'}(\Delta\theta) \coloneqq \Delta\theta, \quad \mathcal{T}_\theta^{\theta'} : \mathrm{Tan}_\theta \to \mathrm{Tan}_{\theta'} \tag{3.2.3}$$

define valid retraction and vector transport, respectively.

Under a local nondegeneracy assumption, we can quantify the convergence of RAM

Assumption 3.1. *Assume that for a fixed-point θ^*, there is some $R > 0$ and $0 < \lambda_G \le \Lambda_G$ such that*

$$\lambda_G\,\mathrm{Id} \preceq G(\theta) \preceq \Lambda_G\,\mathrm{Id}, \quad \forall\theta : \|\theta - \theta_*\|_2 < R \tag{3.2.4}$$

Proposition 3.2. *Under Assumption 3.1, there is a geodesic ball, in which retraction equation 3.2.2 and vector transport equation 3.2.3 satisfy requirements for convergence of RAM. If the operator is locally contractive with coefficient $\kappa < 1$, and under additional smoothness assumptions on the operator, the sequence generated by Algorithm 1, satisfies*

$$\|r_{k+1}\|_{\theta_{k+1}} \le \gamma_k\left(1 - \beta_k + \kappa\beta_k\right)\|r_k\|_{\theta_k} + \sum_{i=k-m_k+1}^{k}\mathcal{O}(\|r_i\|^2_{\theta_i}),\ \gamma_k = \frac{\|r_k - R_k\Gamma_k\|}{\|r_k\|} \le 1.$$

In Figure 1, Assumption 3.1 is verified numerically.

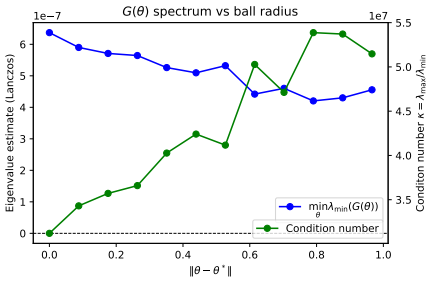


Figure 1: Empirical assessment of Assumption 3.1. Estimated minimal eigenvalue and condition number stay bounded away from 0 and ∞ in the unit ball around the fixed-point. See Appendix C.3 for detailed description.

Remark 3.1. *The estimate equation 3.2 is not an explicit convergence rate, but rather a comparison of local residual decrease against relaxed Picard step (in that case it would be exactly $(1 - \beta_k + \kappa\beta_k)\,\|r_k\|$). Estimates of this type are state-of-the-art for AM both on Riemannian manifolds Li & Bao (2023) and in the Euclidean space Toth & Kelley (2015); Evans et al. (2020); Rebholz & Xiao (2023). The gain factor γ_k quantifying the speed-up depends on how well the new residual is represented in the affine hull of the previous ones.*

RAM with an arbitrary β is equivalent to RAM with $\beta = 1$ and step size βh, thus, acceleration of RAM with $\beta > 1$ might be related to larger step size rather than to the usage of historical information. On the other hand, if γ_k is small, the leading term of the residual could still be decreasing even if $\beta_k > 1$ or $\kappa > 1$, thus RAM can still converge when NGD does not, as noted empirically in Lott et al. (2012); Willert et al. (2014), allowing for faster convergence. For the fairness of the evaluation, in the parameter sweeps below, we report the best acceleration for $\beta = 1$ as well for the best observed β.

The presence of higher-order terms in equation 3.2 reinforces the common practice of keeping the history short. In our experiments $m \leq 12$.

We analyze the assumptions for convergence of RAM in subsubsection A.3.1. Proofs of the propositions can be found in subsection B.2.

4 Numerical Experiments

In all experiments, $\rho_{\text{ref}} = \mathcal{N}(0, I)$ and $T_{\theta_0} \approx \text{Id}$. We parameterize T_θ as the time-1 flow of a neural ODE Chen et al. (2018),

$$\dot{z}(t) = v_\theta(z(t), t), \qquad z(0) = z, \qquad T_\theta(z) = z(1),$$

where v_θ is a dense feedforward network with n_{hidden} layers of width D_{hidden}, with time concatenated to the input. Such models provide expressive transport maps Holderrieth & Erives (2025); increasing capacity is expected to reduce the projection error δ_0.

RAM depends on (m, λ, β) and the NGD step size h. We first tune h for NGD and then sweep (m, λ, β) at the selected h.

4.1 Sampling Bayesian posteriors

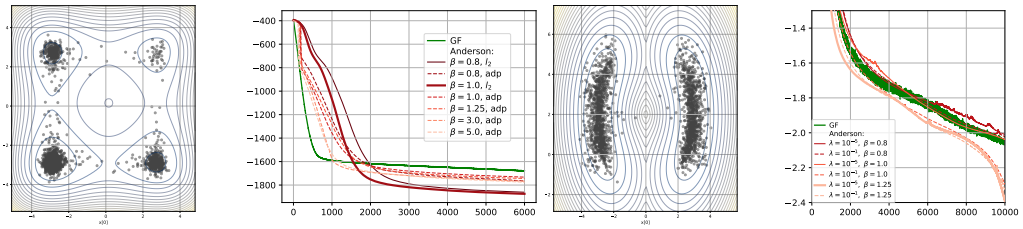
To sample from a target density ρ_∞ we minimize the KL divergence

$$\theta_* = \arg\min_{\rho_\theta \in \mathcal{P}_\Theta} \text{KL}(\rho_\theta | \rho_\infty). \tag{4.1.1}$$

The main challenges for standard samplers such as MCMC Hastings (1970) or SVGD Liu & Wang (2016) are concentration and multimodality of the target. Posteriors, appearing in applications Dinkel et al. (2024); Dashti & Stuart (2017), typically include computationally heavy components, thus, the computation of the loss gradient is the bottleneck of the algorithm, and it is natural to use the number its evaluation to compare different methods. We evaluate RAM on two synthetic problems with multimodal potentials (*Styblinski–Tang* (ST) Styblinski & Tang (1990) and *double-banana* Detommaso et al. (2018) (DB)), a Bayesian inverse problem for the heat equation Roberts et al. (2026) and benchmark Bayesian logistic regression datasets. The detailed setup is provided in Appendix C.1.

4.1.1 Synthetic posteriors

Figure 8 shows that RAM reduces the number of gradient evaluations needed to reach the low-energy regime. The NGD step size h was tuned to give the fastest convergence while preserving



(a) Styblinski–Tang (b) Energy decay (c) Double banana (d) Energy decay

Figure 2: Multimodal sampling. (a,c): samples from best trained models and contour lines of target potentials. (b,d): energy decay vs. iteration. RAM improves the convergence of NGD.

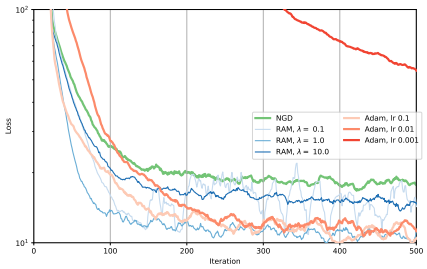


Figure 3: Loss vs. iteration for the Heat inverse problem (moving average).

Method	**No. iter.**	**iter. / s**	**T (s)**
NGD	4037	1.83	2226
RAM, $\lambda = 10.0$	2915	1.79	1628
RAM, $\lambda = 1.0$	**242**	1.79	**135**
RAM, $\lambda = 0.1$	571	1.79	319
Adam, $\eta = 0.1$	1128	1.96	576
Adam, $\eta = 0.01$	**968**	1.96	**494**
Adam, $\eta = 0.001$	3243	1.96	1655
SGD	N/A	2.09	N/A

Table 1: Iterations, iterations per second and wall-clock time to convergence for the Heat inverse problem. Best time for each optimizer family in bold.

the multimodal structure. For both targets, over-relaxed RAM updates with $\beta > 1$ performed best. On ST, we tested both ℓ_2 and adaptive regularization; adaptive regularization was less sensitive to hyperparameters, though slower than the best tuned ℓ_2 variant. On DB, the best setting used ℓ_2 regularization with $\beta = 1.25$ and $\lambda = 10^{-1}$.

Since these targets model Bayesian posteriors, where potential evaluations often dominate the cost, we also compare against SGD Amari (1993) and Adam Kingma & Ba (2014) in terms of loss-gradient evaluations. On ST, our hyperparameter search did not find SGD or Adam settings that preserved all modes. On DB, Figure 7 shows the objective gap in log scale. After 10^4 iterations, Adam and SGD remain approximately one and two orders of magnitude above RAM, respectively.

4.1.2 PDE-BASED INVERSE PROBLEM

Next, a coefficient inverse problem for an elliptic PDE in one dimension, was considered Roberts et al. (2026). The goal is to reconstruct the conductivity (in form of the the first d terms in the Karhunen–Loève expansion of the prior Gaussian process) from the measurements of the temperature field. See subsubsection C.1.1 for details.

For SGD and Adam, we tested a broad range of learning rates, For RAM, we fixed $\beta = 1$ and the history length to $m = 6$, used adaptive regularization, and varied only the regularization factor The results are presented in Figure 1. All methods have comparable per-iteration execution time, with the slowest method, RAM, and the fastest method, SGD, differing by only about 18%, due to the loss gradient being the bottleneck of the iteration. In particular, the overhead of RAM relative to NGD is insignificant, since their iteration rates differ by approximately 2% due to efficient parallel computation of the scalar products. Given the comparable per-iteration cost, RAM reduces the convergence time of NGD by at least 40%. The best-performing RAM configuration is also 3.6 times faster than the best Adam configuration in wall-clock convergence time.

4.1.3 BAYESIAN LOGISTIC REGRESSION

We next consider Bayesian logistic regression, following the setup of Fan et al. (2022); Mokrov et al. (2021). Further dataset and implementation details are provided in Appendix C.2.

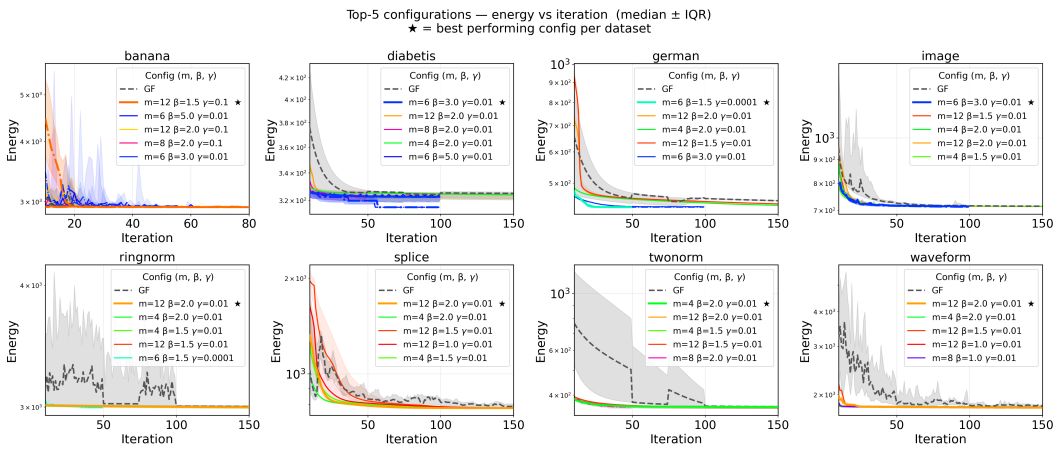


Figure 4: Bayesian logistic regression convergence across eight benchmark datasets. Each panel compares NGD with the top five RAM configurations from the hyperparameter sweep for 3 seeds. RAM reaches the low-energy regime in fewer iterations than NGD.

Figure 4 compares NGD with the top five RAM configurations selected from our hyperparameter sweep. The complete list of tested configurations is reported in Appendix C.2. Across all datasets, RAM reaches the low-energy regime in fewer iterations than NGD for at least one configuration, and the configuration $(m, \beta, \lambda) = (12, 2.0, 0.01)$ appears among the top-performing configurations for every dataset. We therefore use this configuration to report the downstream predictive accuracy and test log-likelihood.

4.2 Aggregation models

We consider the interaction energy functional

$$\mathcal{E}(\rho) = \frac{1}{2} \int \int \left(\frac{|x-y|^4}{4} - \frac{|x-y|^2}{2} \right) \mathrm{d}\rho(x) \mathrm{d}\rho(y).$$

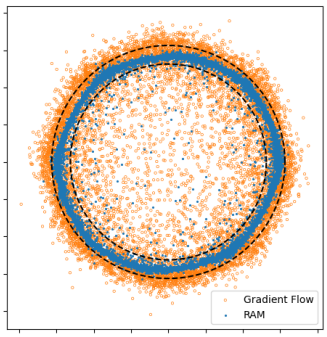
The stationary state is supported on a ring of radius 0.5 centered at the origin. Figure 5a shows the pushforward samples at iteration 300. At this stage, both methods have moved toward the ring-shaped equilibrium, but RAM-adaptive already produces a sharper ring, whereas NGD remains more diffuse and retains more mass in the interior.

To quantify this difference, we report $\mathrm{RingMass}_\epsilon$ i.e. the fraction of particles within a ϵ-tube around a ring of radius 0.5 on which the stationary distribution is concentrated. At iteration 300, $\mathrm{RingMass}_{0.05}$ is 46% for NGD and 96.9% for RAM-adaptive. The convergence diagnostics in Figure 5b show the same effect quantitatively. RAM-adaptive decreases the energy faster both as a function of iteration count and as a function of metric-vector-product count. In particular, RAM-adaptive reaches the low-energy plateau around iteration 10, whereas NGD begins to plateau only around iteration 200. By the final iteration 1500, both methods produce comparable approximations of the ring-shaped steady state with $\mathrm{RingMass}_{0.05}$ being 98.4% for NGD and 99.3% for RAM-adaptive.

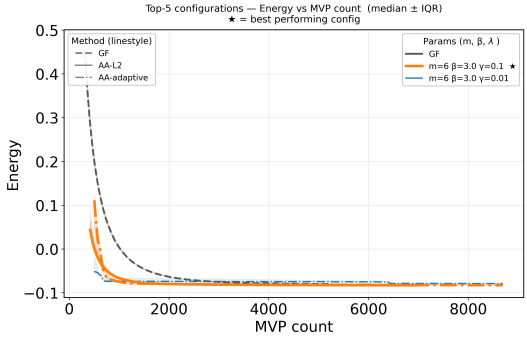
4.3 Generative modeling

To evaluate our method in the generative setting, we consider minimizing the Maximum Mean Discrepancy (MMD) functional with a mixture of Gaussian RBF kernels Gretton et al. (2012). We evaluate the performance of our method on three common model distributions: *two spirals*, *checkerboard* and *eight Gaussians* Grathwohl et al. (2018).

The results are presented on Figure 6, with training data and generated samples on Figure 6a, and decay of the MMD_k vs. iteration on Figure 6b-6d. The Table 2 demonstrates that acceleration can be achieved without over-relaxation ($\beta = 1$). However, the best result in all three cases, with acceleration ranging from $\times 4$ for 8 Gaussians to $\times 17$, was for the over-relaxed $\beta = 3$.

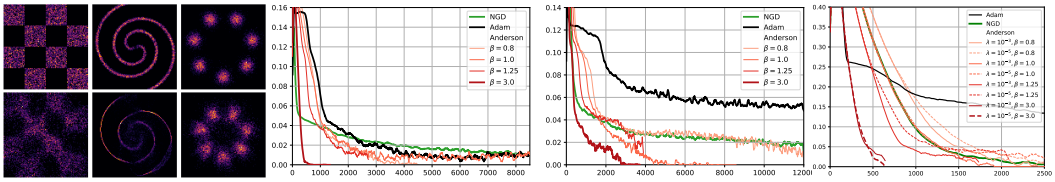


(a) Aggregation comparison. Pushforward samples for the aggregation problem at iteration 300. RAM (blue) produces a sharper ring structure and concentrates more mass near the target ring than NGD (orange).



(b) Energy vs. MVP. RAM reduces the energy faster than NGD both per iteration and per metric-vector product (MVP). , indicating that the acceleration is not only an iteration-count effect.

Figure 5: Convergence diagnostics for the aggregation-energy problem.



(a) Data and generated (b) Checkerboard (c) Two spirals (d) 8 Gaussians

Figure 6: Performance in generative problems.

5 Conclusion and Outlook

We demonstrate that Riemannian methods, in particular, Riemannian Anderson mixing, can accelerate the Natural Gradient Descent iteration and serve as an efficient training method in a broad array of machine-learning problems. An obvious next step would be the adaptation of other Riemannian quasi-Newton methods Huang et al. (2015); Hu et al. (2018); Huang et al. (2018); Kasai et al. (2018); Narushima et al. (2023). As for the convergence theory, the existing estimates can be adapted, if the statements on metric nondegeneracy (Assumption 3.1), sectional curvature bounds and Wasserstein gradient approximation (Assumption 2.1) are established for certain parametric architectures. These broad and interesting questions are left to future work.

Computational resources: The experiments were performed on several workstations including ones with ProLiant XL675d Gen10 Plus 2xAMD EPYC 7402, 24 Core 2800 MHz, 1024 GB RAM, 8 GPUs NVIDIA A100-SXM4 40GB; ProLiant Compute DL380a Gen12 2xXeon, 24 Core 2700 MHz, 1024 GB RAM, 4 GPUs NVIDIA H200 NVL4 141GB; and AMD 7543 CPU + NVIDIA RTX A6000 GPU

Problem	$\mathrm{MMD}_{\mathrm{k}}^2$	**NGD**	**RAM**		β
			$\beta = 1$	**best**	
Checkerboard	0.01	8500	4000	500	3
Two spirals	0.01	12000	3800	2200	3
8 Gaussians	0.001	2500	1750	600	3

Table 2: Number of iterations to convergence in the generative problems

AI USE STATEMENT

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

In this work, we used generative AI tools to assist in the writing and refinement of mathematical proofs, refine research hypotheses, provide feedback on research methodology and experimental design, assist with the implementation of the proposed methods, and support the interpretation of numerical results. We additionally used generative AI tools to summarize and analyze existing literature, identify potentially relevant literature, and edit portions of the manuscript to improve readability.

All AI-assisted mathematical arguments were independently reviewed and verified by the authors, and the final statements, assumptions, and proofs were determined by the authors. AI-assisted implementations were inspected, tested, and validated through the numerical experiments reported in the paper. Literature identified or summarized with the assistance of generative AI was checked against the original sources before being cited or used in the manuscript. All AI-assisted edits to the manuscript were reviewed by the authors. We take responsibility for the final content of this work, including all text, mathematical claims, proofs, code, experimental results, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

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

We provide the information needed to reproduce both the theoretical and experimental results of this work. The proposed Riemannian Anderson Mixing method is specified in Algorithm 1, including the regularization and descent safeguard used in our implementation. Section 4 describes the common neural-ODE parameterization and experimental protocol, while Appendix C reports the network architectures, ODE discretizations, optimizer parameters, metric regularization, and conjugate-gradient parameters used in the experiments. Problem-specific setups, data-processing and evaluation details, and hyperparameter choices are provided in the corresponding subsections of Appendix C, including the Bayesian inverse problem (Appendix C.1.1) and Bayesian logistic-regression experiments (Appendix C.2). Computational resources are reported at the end of the main text. For the theoretical results, the assumptions are stated alongside the corresponding results in Sections 2 and 3, with supporting derivations and proofs provided in the appendices on Riemannian Anderson mixing and the parametrized Wasserstein space

AUTHOR CONTRIBUTIONS

If you'd like to, you may include a section for author contributions as is done in many journals. This is optional and at the discretion of the authors.

Upon acceptance, we will release the code accompanying this work, including the implementation of the proposed method and scripts needed to reproduce the reported experiments.

ACKNOWLEDGMENTS

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper.

REFERENCES

P-A Absil, Robert Mahony, and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. In *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009.

Nabila Aghanim, Yashar Akrami, Mark Ashdown, J Aumont, Carlo Baccigalupi, M Ballardini, Anthony J Banday, RB Barreiro, N Bartolo, S Basak, et al. Planck 2018 results-V. CMB power spectra and likelihoods. *Astronomy & Astrophysics*, 641:A5, 2020.

Vitalii Aksenov, Maxim Chertov, and Konstantin Sinkov. Application of accelerated fixed-point algorithms to hydrodynamic well-fracture coupling. *Computers and Geotechnics*, 129:103783, 2021.

Vitalii Aksenov, Martin Eigel, and Mathias Oster. Anderson mixing in bures wasserstein space of gaussian measures. *arXiv preprint arXiv:2601.22038*, 2026.

Shun-ichi Amari. Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4-5): 185–196, 1993.

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998. doi: 10.1162/089976698300017746.

Luigi Ambrosio and Nicola Gigli. Construction of the parallel transport in the Wasserstein space. 2008.

Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2005.

Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965.

Michael Arbel, Anna Korba, Adil Salim, and Arthur Gretton. Maximum mean discrepancy gradient flow. *Advances in neural information processing systems*, 32, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. Pmlr, 2017.

Siwan Boufadène and François-Xavier Vialard. On the global convergence of wasserstein gradient flow of the coulomb discrepancy (2024). *arXiv preprint arXiv:2312.00800*.

Claude Brezinski, Stefano Cipolla, Michela Redivo-Zaglia, and Yousef Saad. Shanks and anderson-type acceleration techniques for systems of nonlinear equations. *IMA Journal of Numerical Analysis*, 42(4):3058–3093, 2022.

Joan Bruna, Benjamin Peherstorfer, and Eric Vanden-Eijnden. Neural Galerkin schemes with active learning for high-dimensional evolution equations. *Journal of Computational Physics*, 496:112588, January 2024. ISSN 00219991. doi: 10.1016/j.jcp.2023.112588. URL `https://linkinghub.elsevier.com/retrieve/pii/S0021999123006836`.

Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg Stadler. A computational framework for infinite-dimensional bayesian inverse problems part i: The linearized case, with application to global seismic inversion. *SIAM Journal on Scientific Computing*, 35(6):A2494–A2523, 2013.

José A Carrillo, Robert J McCann, and Cédric Villani. Contractions in the 2-Wasserstein length space and thermalization of granular media. *Archive for Rational Mechanics and Analysis*, 179: 217–263, 2006.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Shi Chen, Qin Li, Oliver Tse, and Stephen J Wright. Accelerating optimization over the space of probability measures. *Journal of Machine Learning Research*, 26(31):1–40, 2025.

Zhuo Chen, Jacob McCarran, Esteban Vizcaino, Marin Soljačić, and Di Luo. TENG: Time-Evolving Natural Gradient for Solving PDEs With Deep Neural Nets Toward Machine Precision, June 2024. URL `http://arxiv.org/abs/2404.10771`. arXiv:2404.10771 [cs].

Jon Cockayne, Chris Oates, Tim Sullivan, and Mark Girolami. Probabilistic numerical methods for pde-constrained bayesian inverse problems. In *AIP Conference Proceedings*, volume 1853, pp. 060001. AIP Publishing LLC, 2017.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

Masoumeh Dashti and Andrew M Stuart. The Bayesian approach to inverse problems. In *Handbook of uncertainty quantification*, pp. 311–428. Springer, 2017.

Gianluca Detommaso, Tiangang Cui, Youssef Marzouk, Alessio Spantini, and Robert Scheichl. A stein variational newton method. *Advances in Neural Information Processing Systems*, 31, 2018.

Maximilian Dinkel, Carolin M Geitner, Gil Robalo Rei, Jonas Nitzler, and Wolfgang A Wall. Solving bayesian inverse problems with expensive likelihoods using constrained gaussian processes and active learning. *Inverse Problems*, 40(9):095008, 2024.

Yifan Du and Tamer A. Zaki. Evolutional Deep Neural Network. *Physical Review E*, 104(4): 045303, October 2021. ISSN 2470-0045, 2470-0053. doi: 10.1103/PhysRevE.104.045303. URL http://arxiv.org/abs/2103.09959. arXiv:2103.09959 [physics].

Gintare Karolina Dziugaite, Daniel M Roy, and Zoubin Ghahramani. Training generative neural networks via maximum mean discrepancy optimization. *arXiv preprint arXiv:1505.03906*, 2015.

Claire Evans, Sara Pollock, Leo G Rebholz, and Mengying Xiao. A proof that anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically). *SIAM Journal on Numerical Analysis*, 58(1):788–810, 2020.

Jiaojiao Fan, Qinsheng Zhang, Amirhossein Taghvaei, and Yongxin Chen. Variational Wasserstein gradient flow. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 6185–6215. PMLR, June 2022. URL https://proceedings.mlr.press/v162/fan22d.html.

Haw-ren Fang and Yousef Saad. Two classes of multisecant methods for nonlinear acceleration. *Numerical linear algebra with applications*, 16(3):197–221, 2009.

Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

Nicola Gigli. On the geometry of the space of probability measures in $\mathbb{R}^n$ endowed with the quadratic optimal transport distance. 2008.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Will Grathwohl, Ricky TQ Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *arXiv preprint arXiv:1810.01367*, 2018.

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(1):723–773, 2012.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970. ISSN 0006-3444. doi: 10.1093/biomet/57.1.97. URL https://doi.org/10.1093/biomet/57.1.97.

Peter Holderrieth and Ezra Erives. An Introduction to Flow Matching and Diffusion Models, July 2025. URL http://arxiv.org/abs/2506.02070. arXiv:2506.02070 [cs].

Richard Holley and Daniel W Stroock. Logarithmic sobolev inequalities and stochastic ising models. 1986.

Jiang Hu, Andre Milzarek, Zaiwen Wen, and Yaxiang Yuan. Adaptive quadratically regularized newton method for riemannian optimization. *SIAM Journal on Matrix Analysis and Applications*, 39(3):1181–1207, 2018.

Wen Huang, Kyle A Gallivan, and P-A Absil. A broyden class of quasi-newton methods for riemannian optimization. *SIAM Journal on Optimization*, 25(3):1660–1685, 2015.

Wen Huang, P-A Absil, and Kyle A Gallivan. A riemannian bfgs method without differentiated retraction for nonconvex optimization problems. *SIAM Journal on Optimization*, 28(1):470–495, 2018.

Yijie Jin, Shu Liu, Hao Wu, Xiaojing Ye, and Haomin Zhou. Parameterized Wasserstein gradient flow. *Journal of Computational Physics*, 524:113660, March 2025. ISSN 0021-9991. doi: 10.1016/j.jcp.2024.113660. URL https://www.sciencedirect.com/science/article/pii/S0021999124009082.

Richard Jordan, David Kinderlehrer, and Felix Otto. The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17, 1998.

Leonid V Kantorovich. On the translocation of masses. In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pp. 199–201, 1942.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 795–811. Springer, 2016.

Hiroyuki Kasai, Hiroyuki Sato, and Bamdev Mishra. Riemannian stochastic quasi-newton algorithm with variance reduction and its convergence analysis. In Amos Storkey and Fernando Perez-Cruz (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 269–278. PMLR, 09–11 Apr 2018. URL https://proceedings.mlr.press/v84/kasai18a.html.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Chenyi Li, Shuchen Zhu, Zhonglin Xie, and Zaiwen Wen. Accelerated Natural Gradient Method for Parametric Manifold Optimization. *arXiv.org*, April 2025. URL https://arxiv.org/abs/2504.05753v1.

Zanyu Li and Chenglong Bao. Riemannian Anderson mixing methods for minimizing C^2-functions on Riemannian manifolds. *arXiv preprint arXiv:2309.04091*, 2023.

Konstantin Lipnikov, Daniil Svyatskiy, and Y Vassilevski. Anderson acceleration for nonlinear finite volume scheme for advection-diffusion problems. *SIAM Journal on Scientific Computing*, 35(2): A1120–A1136, 2013.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose Bayesian inference algorithm. *Advances in neural information processing systems*, 29, 2016.

Qiong Liu, Yu-Jung Tsai, Jean-Dominique Gallezot, Xueqi Guo, Ming-Kai Chen, Darko Pucar, Colin Young, Vladimir Panin, Michael Casey, Tianshun Miao, Huidong Xie, Xiongchao Chen, Bo Zhou, Richard Carson, and Chi Liu. Population-based deep image prior for dynamic PET denoising: A data-driven approach to improve parametric quantification. *Medical Image Analysis*, 95:103180, 2024a. ISSN 1361-8415. doi: https://doi.org/10.1016/j.media.2024.103180. URL https://www.sciencedirect.com/science/article/pii/S1361841524001051.

Shu Liu, Wuchen Li, Hongyuan Zha, and Haomin Zhou. Neural Parametric Fokker–Planck Equation. *SIAM Journal on Numerical Analysis*, 60(3):1385–1449, June 2022. ISSN 0036-1429, 1095-7170. doi: 10.1137/20M1344986. URL https://epubs.siam.org/doi/10.1137/20M1344986.

Shu Liu, Stanley Osher, and Wuchen Li. A Natural Primal-Dual Hybrid Gradient Method for Adversarial Neural Network Training on Solving Partial Differential Equations, December 2024b. URL http://arxiv.org/abs/2411.06278. arXiv:2411.06278 [math].

PA Lott, HF Walker, CS Woodward, and UM Yang. An accelerated picard method for nonlinear systems related to variably saturated flow. *Advances in Water Resources*, 38:92–101, 2012.

S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.R. Mullers. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No.98TH8468)*, pp. 41–48, August 1999. doi: 10.1109/NNSP.1999.788121. URL https://ieeexplore.ieee.org/document/788121/metrics.

Petr Mokrov, Alexander Korotin, Lingxiao Li, Aude Genevay, Justin Solomon, and Evgeny Burnaev. Large-Scale Wasserstein Gradient Flows, October 2021. URL http://arxiv.org/abs/2106.00736. arXiv:2106.00736 [cs].

Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pp. 666–704, 1781.

Johannes Müller and Marius Zeinhofer. Achieving high accuracy with pinns via energy natural gradient descent. In *International Conference on Machine Learning*, pp. 25471–25485. PMLR, 2023.

Yasushi Narushima, Shummin Nakayama, Masashi Takemura, and Hiroshi Yabe. Memoryless quasi-newton methods based on the spectral-scaling broyden family for riemannian optimization. *Journal of Optimization Theory and Applications*, 197(2):639–664, 2023.

Levon Nurbekyan, Wanzhou Lei, and Yunan Yang. Efficient natural gradient descent methods for large-scale pde-based optimization problems. *SIAM Journal on Scientific Computing*, 45(4): A1621–A1655, 2023.

Felix Otto. The geometry of dissipative evolution equations: the porous medium equation. 2001.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Barak A Pearlmutter. Fast exact multiplication by the hessian. *Neural computation*, 6(1):147–160, 1994.

Noemi Petra, James Martin, Georg Stadler, and Omar Ghattas. A computational framework for infinite-dimensional bayesian inverse problems, part ii: Stochastic newton mcmc with application to ice sheet flow inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1525–A1555, 2014.

Xin Qu, Wei Bian, and Xiaojun Chen. An extra gradient anderson-accelerated algorithm for pseudomonotone variational inequalities. *Mathematics of Computation*, 95(359):1361–1387, 2026.

Leo G Rebholz and Mengying Xiao. The effect of anderson acceleration on superlinear and sublinear convergence. *Journal of Scientific Computing*, 96(2):34, 2023.

Andrew Gerard Roberts, Michael Dietze, and Jonathan H Huggins. Propagating surrogate uncertainty in bayesian inverse problems. *arXiv preprint arXiv:2601.03532*, 2026.

Morgan A Schmitz, Matthieu Heitz, Nicolas Bonneel, Fred Ngole, David Coeurjolly, Marco Cuturi, Gabriel Peyré, and Jean-Luc Starck. Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM Journal on Imaging Sciences*, 11(1):643–678, 2018.

Damien Scieur. Adaptive quasi-newton and anderson acceleration framework with explicit global (accelerated) convergence rates. In *International Conference on Artificial Intelligence and Statistics*, pp. 883–891. PMLR, 2024.

Zebang Shen, Zhenfu Wang, Alejandro Ribeiro, and Hamed Hassani. Sinkhorn Natural Gradient for Generative Models.

Viktor Stein and Wuchen Li. Accelerated stein variational gradient flow. In *International Conference on Geometric Science of Information*, pp. 80–90. Springer, 2025.

MA Styblinski and T-S Tang. Experiments in nonconvex optimization: stochastic approximation with function smoothing and simulated annealing. *Neural Networks*, 3(4):467–483, 1990.

Ke Sun, Yafei Wang, Yi Liu, Bo Pan, Shangling Jui, Bei Jiang, Linglong Kong, et al. Damped anderson mixing for deep reinforcement learning: Acceleration, convergence, and stabilization. *Advances in Neural Information Processing Systems*, 34:3732–3743, 2021.

Bohao Tang, Nicholas C Henderson, and Ravi Varadhan. Accelerating fixed-point algorithms in statistics and data science: A state-of-art review. *Journal of Data Science*, 21(1), 2023.

Ziyuan Tang, Tianshi Xu, Huan He, Yousef Saad, and Yuanzhe Xi. Anderson acceleration with truncated Gram–Schmidt. *SIAM Journal on Matrix Analysis and Applications*, 45(4):1850–1872, 2024.

Alex Toth and Carl T Kelley. Convergence analysis for Anderson acceleration. *SIAM Journal on Numerical Analysis*, 53(2):805–819, 2015.

Titouan Vayer, Laetitia Chapel, Rémi Flamary, Romain Tavenard, and Nicolas Courty. Optimal transport for structured data with application on graphs. *arXiv preprint arXiv:1805.09114*, 2018.

Cédric Villani. *Topics in optimal transportation*, volume 58. American Mathematical Soc., 2021.

Homer F Walker and Peng Ni. Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis*, 49(4):1715–1735, 2011.

Fuchao Wei, Chenglong Bao, and Yang Liu. Stochastic anderson mixing for nonconvex stochastic optimization. *Advances in Neural Information Processing Systems*, 34:22995–23008, 2021.

Fuchao Wei, Chenglong Bao, and Yang Liu. A class of short-term recurrence Anderson mixing methods and their applications. In *International Conference on Learning Representations*, 2022a.

Fuchao Wei, Chenglong Bao, Yang Liu, and Guangwen Yang. A variant of anderson mixing with minimal memory size. *Advances in Neural Information Processing Systems*, 35:16650–16663, 2022b.

Fuchao Wei, Chenglong Bao, Yang Liu, and Guangwen Yang. Convergence analysis for restarted Anderson mixing and beyond. *arXiv preprint arXiv:2307.02062*, 2023.

Jeffrey Willert, William T Taitano, and Dana Knoll. Leveraging anderson acceleration for improved convergence of iterative solutions to transport systems. *Journal of Computational Physics*, 273: 278–286, 2014.

Hao Wu and Haomin Zhou. Deep Tangent Bundle (DTB) method: a Deep Neural Network approach to compute solutions of PDES, August 2025. URL `http://arxiv.org/abs/2509.00957`. arXiv:2509.00957 [math].

Hao Wu, Shu Liu, Xiaojing Ye, and Haomin Zhou. Parameterized wasserstein hamiltonian flow. *SIAM Journal on Numerical Analysis*, 63(1):360–395, 2025.

Hongteng Xu, Wenlin Wang, Wei Liu, and Lawrence Carin. Distilled wasserstein learning for word embedding and topic modeling. *Advances in Neural Information Processing Systems*, 31, 2018.

Shuailong Zhu and Xiaohui Chen. Convergence analysis of the wasserstein proximal algorithm beyond geodesic convexity. *arXiv preprint arXiv:2501.14993*, 2025.

A ANDERSON MIXING

A.1 ANDERSON MIXING ON $\mathbb{R}^D$

Anderson mixing (see Fang & Saad (2009) and references therein) is an algorithm of accelerating fixed-point iterations, i.e. a problem of finding an x such that

$$x = G(x)$$

for some operator $G : \mathbb{R}^D \to \mathbb{R}^D$. The fixed-point formulation is quite general. A minimization problem for a differentiable function can be recast as a fixed-point

$$x^* = \arg\min_x f(x) \Rightarrow \nabla f(x^*) = 0 \Leftrightarrow x^* \text{ — fixed point of } \mathrm{Id} - h\nabla f \tag{A.1.1}$$

In Anderson mixing (Algorithm 2), at each iteration k the history of up to m previous iterates $\{x_{k-i}\}_{i=0}^{m-1}$ and values of the operator $g_i = G(x_i)$ is stored. The next iterate is chosen by a linear combination with coefficients acquired from solving a least-squares subproblem over the affine hull of the previous iterates. Note that the method is trivially reformulated using residuals $r_k = g_k - x_k$, such that finding the fixed-point is equivalent to setting the residual to zero.

Algorithm 2 Anderson mixing in $\mathbb{R}^d$

Parameters: $x_0 \in \mathbb{R}^d$, mixing parameters $0 \leq \beta_k \leq 1$, and memory parameter $m \geq 1$.
Returns: The algorithm computes a sequence $x_0, x_1, \ldots$, intended to converge to a fixed-point of $G : \mathbb{R}^d \to \mathbb{R}^d$.

1: **for** $k = 0, 1, \ldots$ until convergence **do**
2: $m_k = \min(m, k)$.
3: Compute $R_k = (r_{k-m_k}, \ldots, r_k)$, where $r_i = G(x_i) - x_i$.
4: Solve

$$\alpha^{(k)} = \arg\min_{v \in \mathbb{R}^{m_k+1}} \begin{cases} \|R_k v\|_2 \\ \text{s.t. } \sum_{i=0}^{m_k} v_i = 1 \end{cases} \tag{A.1.2}$$

5: Set

$$x_{k+1} = (1 - \beta_k) \sum_{i=0}^{m_k} \alpha_i^{(k)} x_{k-m_k+i} + \beta_k \sum_{i=0}^{m_k} \alpha_i^{(k)} G(x_{k-m_k+i}). \tag{A.1.3}$$

6: **end for**

The update step equation A.1.3 can be rewritten as

$$x_{k+1} = \sum_{i=0}^{m_k} \alpha_i^{(k)} x_{k-m_k+i} + \beta_k \sum_{i=0}^{m_k} \alpha_i^{(k)} r_{k-m_k+i}.$$

The minimization subproblem equation A.1.2 is constrained, but can be reformulated Anderson (1965) as an unconstrained one. Introducing new variables

$$\gamma_i^{(k)} = \sum_{j=k-m_k}^{i} \alpha_j^{(k)},\ i = \overline{k - m_k,\ k-1}$$

$$\alpha_k^{(k)} = 1 - \sum_{i=k-m_k}^{k-1} \alpha_i^{(k)}$$

one can see that

$$\sum_{i=k-m_k}^{k} \alpha_i^{(k)} r_i = \left(1 - \sum_{i=k-m_k}^{k-1} \alpha_i^{(k)}\right) r_k + \sum_{i=k-m_k}^{k-1} \alpha_i^{(k)} r_i =$$

$$= r_k + \sum_{i=k-m_k}^{k-1} \alpha_i^{(k)} (r_i - r_k) = r_k + \sum_{i=k-m_k+1}^{k-1} (\gamma_i^{(k)} - \gamma_{i-1}^{(k)})(r_i - r_k) + \gamma_{k-m_k}^{(k)} r_{k-m_k} =$$

$$r_k + \sum_{i=k-m_k+1}^{k-1} \gamma_i^{(k)} (r_i - r_k) - \sum_{i=k-m_k+1}^{k-1} \gamma_i^{(k)} (r_{i+1} - r_k) + \gamma_{k-m_k}^{(k)} r_{k-m_k} =$$

$$= r_k - \sum_{i=k-m_k}^{k-1} \gamma_i^{(k)} (r_{i+1} - r_i)$$

Introducing shorthand notation for the forward finite difference

$$\Delta r_k = r_{k+1} - r_k,\ \Delta x_k = x_{k+1} - x_k, \tag{A.1.4}$$

and the matrices

$$X_k = [\Delta x_{k-m}, \Delta x_{k-m+1}, \ldots, \Delta x_{k-1}] \tag{A.1.5}$$

$$R_k = [\Delta r_{k-m}, \Delta r_{k-m+1}, \ldots, \Delta r_{k-1}] \tag{A.1.6}$$

$$\Gamma = \left(\gamma_{k-m}^{(k)}, \ldots, \gamma_{k-1}^{(k)}\right)^T, \tag{A.1.7}$$

then Anderson mixing in $\mathbb{R}^D$ can be rewritten as:

$$\begin{cases} \Gamma_k = \arg\min_{\Gamma} \|r_k - R_k\Gamma\|_2, \\ \bar{r}_k = r_k - R_k\Gamma_k, \\ x_{k+1} = x_k - X_k\Gamma_k + \beta_k\bar{r}_k, \end{cases} \tag{A.1.8}$$

A typical assumption for convergence of AM, as in Toth & Kelley (2015) if that the coefficients Γ_k, generated by the method, are uniformly bounded over all iterations k in some norm. In practice, to satisfy these assumptions, a regularization term is added to the least-squares subproblem, simplest variant being l_2 regularization $\lambda\|\Gamma_k\|_2^2$. Adapritive regularization was considered in Wei et al. (2021); Li & Bao (2023).

To facilitate the solution of the least-squares problem equation A.1.2, QR- or SVD-decompositions can be used Fang & Saad (2009); Walker & Ni (2011), with decomposition matrices updated iteratively. Instead of fixing the length of history m, one can consider an adaptive restart strategy Wei et al. (2023). In Wei et al. (2022b;a), modifications with minimal memory requirements are considered. Truncated decomposition of the historical vectors is considered in Tang et al. (2024). The extensions of these methods to Riemannian manifolds is not obvious, because the orthogonality of the vectors in general is not preserved under vector transport, and thus is left to future work.

A.2 AM AS QUASI-NEWTON METHOD

The rationale of Anderson mixing in $\mathbb{R}^d$ is that Δr_k are used to incorporate second-order information Fang & Saad (2009). In particular, let J denote the Jacobian of the residual. Then the Newton update takes the form

$$x_{k+1} = x_k - J^{-1}(x_k)r_k$$

Instead of computing $J(x_k)$, it (or its inverse) can be approximated by a matrix, that is updated iteratively. It is common to impose the following conditions on the update

$$J_{k+1}\Delta x_k = \Delta r_k \qquad \textit{(secant condition)}$$
$$J_{k+1}q = J_k q,\ \forall q \perp \Delta x_k \qquad \textit{(no-change condition)}$$

The update formula for J_k, satisfying these two conditions, was found by Broyden:

$$J_{k+1} = J_k + (\Delta r_k - J_k\Delta x_k)\frac{\Delta x_k^T}{\Delta x_k^T\Delta x_k}$$

Alternatively, similar conditions can be imposed on the approximate inverse of the Jacobian, i.e. $G_k \approx J^{-1}(x_k)$.

$$G_{k+1}\Delta r_k = \Delta x_k \qquad \textit{(secant condition)}$$
$$G_{k+1}q = G_k q,\ \forall q \perp \Delta r_k \qquad \textit{(no-change condition)}$$

with the update formula

$$G_{k+1} = G_k + (\Delta x_k - G_k\Delta r_k)\frac{\Delta r_k^T}{\Delta r_k^T\Delta r_k}$$

One can also consider satisfying m secant conditions at the time, i.e.

$$G_{k+1}R_k = X_k \qquad \textit{(secant condition)}$$
$$G_{k+1}q = G_k q,\ \forall q \perp \mathrm{Span}\{\Delta r_{k-m}, \dots \Delta r_{k-1}\} \qquad \textit{(no-change condition)}$$

where $R_k = [\Delta r_{k-i}]_{i=1}^m,\ X_k = [\Delta x_{k-i}]_{i=1}^m$. This leads to the update formula

$$G_k = G_{k-m} + (X_k - G_{k-m}R_k)(R_k^TR_k)^{-1}R_k^T \tag{A.2.1}$$

$$x_{k+1} = x_k - G_{k-m}r_k - (X_k - G_{k-m}R_k)(R_k^TR_k)^{-1}R_k^Tr_k \tag{A.2.2}$$

On the other hand, the Anderson update takes the form

$$x_{k+1} = x_k + \beta r_k - (X_k + \beta R_k)\Gamma_k.$$

Since (without regularization) we can get Γ_k by solving the normal equations

$$\Gamma_k = \arg\min_{\Gamma} \|r_k - R_k\Gamma\|_2^2 = (R_k^TR_k)^{-1}R_k^Tr_k$$

the Anderson update becomes

$$x_{k+1} = x_k + \beta r_k - (X_k + \beta R_k)(R_k^TR_k)^{-1}R_k^Tr_k$$

which is equation A.2.2 with $G_{k-m} := -\beta I$.

A.3 ANDERSON MIXING ON A RIEMANNIAN MANIFOLD

The Riemannian version of Anderson mixing is described in Li & Bao (2023). The authors consider some Riemannian manifold $\boldsymbol{M}$, a vector field $F : \boldsymbol{M} \to \operatorname{Tan} \boldsymbol{M}$ and the operator $g : \boldsymbol{M} \to \boldsymbol{M},\ g(x) := \operatorname{Exp}_x(-F(x))$, so finding a fixed-point of g is equivalent to finding $x : F(x) = 0$. Direct adoption of equation A.1.5 is not possible due to the lack of linear structure on the manifold. Firstly, the residuals are not well defined, but it is reasonable to set the values of F as their natural counterpart. Secondly, the forward difference is not defined, because, in general, for different x_i the residuals $r_k = -F(x_k)$ lie in different spaces $\operatorname{Tan}_{x_k} \boldsymbol{M}$. The authors overcome this by using parallel transports to the tangent space of the current iterate.

Definition A.1 (Retraction). *A retraction on a manifold $\mathcal{M}$ is a smooth mapping $\mathcal{R}$ from the tangent bundle $\operatorname{Tan} \mathcal{M}$ onto $\mathcal{M}$ with the following properties. Let R_x denote the restriction of $\mathcal{R}$ to $\operatorname{Tan}_x \mathcal{M}$.*

1. *$\mathcal{R}_x(0_x) = x$, where 0_x denotes the zero element of $\operatorname{Tan}_x \mathcal{M}$.*
2. *With the canonical identification $\operatorname{Tan}_{0_x} \operatorname{Tan}_x \mathcal{M} \simeq \operatorname{Tan}_x \mathcal{M}$, $\mathcal{R}_x$ satisfies*

$$D\mathcal{R}_x(0_x) = \mathrm{id}_{\operatorname{Tan}_x \mathcal{M}},$$

where $\mathrm{id}_{\operatorname{Tan}_x \mathcal{M}}$ denotes the identity mapping on $\operatorname{Tan}_x \mathcal{M}$.

Definition A.2 (Vector transport). *A vector transport on a manifold $\mathcal{M}$, associated to a retraction R is a smooth mapping $\mathcal{T} : \operatorname{Tan} \mathcal{M} \oplus \operatorname{Tan} \mathcal{M} \to \operatorname{Tan} \mathcal{M}$ such that for every $x, \eta_x \in \operatorname{Tan}_x \mathcal{M}$ it is a linear map between $\operatorname{Tan}_x \mathcal{M}$ and $\operatorname{Tan}_{R_x(\eta_x)} \mathcal{M}$ and*

$$\mathcal{T}_{0_x} \xi_x = \xi_x \quad \forall \xi_x \in \operatorname{Tan}_x \mathcal{M}. \tag{A.3.1}$$

If the associated retraction is invertible for each $x,\ y \in \mathcal{M}$, we denote the vector transport as $\mathcal{T}_x^y := \mathcal{T}_{R_x^{-1}(y)}$.

The method is summarized in Algorithm 3, with $\mathcal{T}_y^x : \operatorname{Tan}_y \boldsymbol{M} \to \operatorname{Tan}_x \boldsymbol{M}$ denoting the vector transport, and $\mathcal{R}_x : \operatorname{Tan}_x \boldsymbol{M} \to \boldsymbol{M}$ is the retraction mapping.

Algorithm 3 Riemannian Anderson Mixing Method

Parameters: $x_0 \in \boldsymbol{M}, \epsilon, \beta_k > 0, m \in \mathbb{N}^*, k = 1.$

$r_0 = -F(x_0)$.
$x_1 = \mathcal{R}_{x_0}(r_0)$.
while $\|F(x_k)\| \geq \epsilon$ **do**
  $m_k = \min\{m, k\}$.
  $\Delta x_{k-i}^{(k)} = \mathcal{T}_{x_{k-1}}^{x_k} \Delta x_{k-i}^{(k-1)} \in T_{x_k}\mathcal{M}, i = 1, \ldots, m_k$.
  $\Delta r_{k-1}^{(k)} = r_k - \mathcal{T}_{x_{k-1}}^{x_k}\, r_{k-1} \in T_{x_k}\mathcal{M}$.
  if $k \geq 2$ **then**
    $\Delta r_{k-i}^{(k)} = \mathcal{T}_{x_{k-1}}^{x_k} \Delta r_{k-i}^{(k-1)} \in T_{x_k}\mathcal{M}, i = 2, \ldots, m_k$.
  end if
  $X_k = [\Delta x_{k-m_k}^{(k)}, \ldots, \Delta x_{k-1}^{(k)}], R_k = [\Delta r_{k-m_k}^{(k)}, \ldots, \Delta r_{k-1}^{(k)}]$.
  $r_k = -F(x_k)$.

$$\Gamma_k = \arg\min_{\Gamma \in \mathbb{R}^{m_k}} \|r_k - R_k \Gamma\|_{x_k}^2 \tag{A.3.2}$$

  $\bar{r}_k = r_k - R_k\Gamma_k \in T_{x_k}\mathcal{M}, \Delta x_k^{(k)} = -X_k\Gamma_k + \beta_k \bar{r}_k \in T_{x_k}\mathcal{M}$.
  $x_{k+1} = \mathcal{R}_{x_k}(\Delta x_k^{(k)})$.
  $k = k + 1$.
end while

A.3.1 CONVERGENCE OF RAM

In Li & Bao (2023), the following assumptions are made in order to establish the local convergence of RAM:

Assumption A.1. *$\mathcal{M}$ is a complete and connected manifold with bounded sectional curvature and $\mathcal{U}$ is a compact subset of $\mathcal{M}$. Moreover, there exist some $\tilde{r} > 0$ and a neighborhood $\mathcal{U}$ on $\mathcal{M}$ such that the open ball $\mathcal{B}_{\mathcal{M}}(x^*, 2\tilde{r}) \subset \mathcal{U}$ and $F(x^*) = 0$ for some $x^* \in \mathcal{U}$.*

The following assumption provides a uniformly lower bound of the distance and the uniform boundedness of the vector transport in the compact subset $\mathcal{U}$.

Assumption A.2. *There exist a retraction R_x and a constant $\tilde{C} > 0$ such that for all $x \in \mathcal{U}$ and $\|v_x\| < \tilde{r}$, we have*

$$\mathrm{d}\left(\mathcal{R}_x(v_x), x\right) \geq \tilde{C}\|v_x\|.$$

Assumption A.3. *The vector transport $\mathcal{T}$ is a C^1-mapping and there exists a constant $M_{\mathcal{U}} > 0$ such that for any $x, y \in \mathcal{U}$ (whenever $\mathcal{T}_x^y$ is well defined), and $v_x \in T_x\mathcal{M}$, we have $\|\mathcal{T}_x^y v_x\|_y \leq M_{\mathcal{U}}\|v_x\|_x$.*

Similar to the Euclidean case in Toth & Kelley (2015); Wei et al. (2022a), we impose the ¡¡coercive¿¿ and ¡¡locally Lipschitz continuous¿¿ property on the vector field F and the contractive property on the fixed point map $g(x) := \mathrm{Exp}_x(-F(x))$.

Assumption A.4. *There exist $0 < L_1 < L_2$ such that for all $x, y \in \mathcal{B}_{\mathcal{M}}(x^*, 2\tilde{r})$ such that $\mathrm{d}(x, y) < 2\tilde{r}$, we have*

$$L_1\mathrm{d}(x, y) \leq \|F(x) - \mathcal{P}_y^x F(y)\| \leq L_2\mathrm{d}(x, y).$$

Assumption A.5. *There exists $0 < \kappa < 1$ such that for all $x, y \in \mathcal{U}$, we have*

$$\mathrm{d}(g(x), g(y)) \leq \kappa\mathrm{d}(x, y).$$

Define H to be the Jacobian of the vector field F, and we further impose Lipschitz continuous assumption on H.

Assumption A.6. *There exits $L > 0$, such that for all $x, y \in \mathcal{U}$, we have*

$$\left\|\mathcal{P}_y^x H(y)\mathcal{P}_x^y - H(x)\right\| \leq L\mathrm{d}(x, y).$$

The final assumption concerns the uniform boundedness of the extrapolation coefficients Γ_k in the RAM method. This assumption is commonly used for analyzing convergence in the Euclidean setting (Toth & Kelley, 2015, Assumption 2.1).

Assumption A.7. *There exists a positive constant M_Γ such that $\|\Gamma_k\|_\infty \leq M_\Gamma$ for all $k \in \mathbb{N}$.*

The assumptions can be split into assumptions on the manifold's geometry in the vicinity of the fixed-point (Assumption A.1), on the properties of chosen retraction and vector transport (Assumption A.2 and A.3), and on the properties of the operator (Assumption A.4-A.6). In order to satisfy the Assumption A.7, regularization is introduced. In the parametric Wasserstein setting, we show below that under Assumption A.1 and additional nondegeneracy assumptions on the metric G, trivial mappings $(\theta, \Delta) \mapsto \theta + \Delta\theta$ and $\Delta\theta \mapsto \Delta\theta$ serve as a valid retraction and vector transport respectively. Verification of the rest of the assumptions appears to be quite involved, since properties of the particular functional, the parametrization architecture, local neighborhood of the minimizer and the step size, have to be taken into the account. First steps in a similar direction were made in Aksenov et al. (2026), where on the Bures-Wasserstein manifold, local contractivity of an operator $\mathrm{Exp}_\Sigma(-h\partial_W \mathrm{KL}(\Sigma|\Sigma_*))$ was shown, but the general treatment is left out to future work.

B Parametrized Wasserstein space

B.1 Comparison of tangent spaces and distances

Definition B.1 (Mapping to $L^2_{\rho_\theta}$ space)**.** *Consider the mapping*

$$u_\theta : \Delta\theta \mapsto (\partial_\theta T_\theta \Delta\theta) \circ T_\theta^{-1} \tag{B.1.1}$$

where $\partial_\theta T_\theta$ is the Jacobian of T_θ with respect to parameters. The mapping is between

$$u_\theta : \mathbb{R}^D \cong \mathrm{Tan}_\theta\, \Theta \to L^2_{\rho_\theta} \tag{B.1.2}$$

and has the property

$$\langle \Delta\theta_1, \Delta\theta_2 \rangle_\theta = \int_{\mathbb{R}^d} u_\theta(\Delta\theta_1) \cdot u_\theta(\Delta\theta_2)\mathrm{d}\rho_\theta$$

where the scalar product $\langle \cdot, \cdot \rangle_\theta$ is defined in equation 2.1.3. Thus u_θ allows us to isometrically identify the tangent space of θ to a subspace of L^2 of ρ_θ. This could be helpful if we need to project some vectors to the parametric tangent space.

It is important to note that the distance on the manifold Θ is induced by the metric tensor and has the form

$$W^2_{2,\Theta}(\theta_0,\ \theta_1) := \inf_{\theta_t} \int\limits_{t=0}^{t=1} \frac{1}{2} \langle \dot{\theta}_t, \dot{\theta}_t \rangle_\theta \mathrm{d}t \tag{B.1.3}$$

where the infimum is take over all smooth curves $\theta_t : [0,1] \mapsto \Theta$ with $\theta_0,\ \theta_1$ being the start and end points of the curve, respectively, and $\dot{\theta}_t := \frac{\mathrm{d}}{\mathrm{d}t}\theta_t$. One can show that the pair $(\rho_{\theta_t}, u_{\theta_t}(\dot{\theta}_t))$ satisfies the continuity equation. Indeed, taking any test function $\eta \in C_c^\infty\left([0;1] \times \mathbb{R}^d\right)$

$$\begin{aligned}
&\int\limits_0^1 \int\limits_{\mathbb{R}^d} \left\{ \partial_t \eta(t,x) + \left\langle \nabla_x \eta(t,x), \frac{\partial T_{\theta_t}}{\partial \dot{\theta}_t} \circ T^{-1}_{\theta_t} \right\rangle \right\} \mathrm{d}\rho_{\theta_t} \mathrm{d}t = \\
&\qquad = \int\limits_0^1 \int\limits_{\mathbb{R}^d} \left\{ \partial_t \eta(t, T_{\theta_t}(z)) + \left\langle \nabla_x \eta(t, T_{\theta_t}(z)), \frac{\partial T_{\theta_t}}{\partial \dot{\theta}_t}(z) \right\rangle \right\} \mathrm{d}\rho_{\text{ref}}(z) \mathrm{d}t = \\
&\qquad = \int\limits_0^1 \int\limits_{\mathbb{R}^d} \left\{ \partial_t \eta(t, T_{\theta_t}(z)) + \left\langle \nabla_x \eta(t, T_{\theta_t}(z)), \frac{\partial T_{\theta_t}}{\partial \theta_t} \dot{\theta}_t(z) \right\rangle \right\} \mathrm{d}\rho_{\text{ref}}(z) \mathrm{d}t = \\
&\qquad = \int\limits_0^1 \int\limits_{\mathbb{R}^d} \left\{ \partial_t \eta(t, T_{\theta_t}(z)) + \left\langle \nabla_x \eta(t, T_{\theta_t}(z)), \frac{\mathrm{d}}{\mathrm{d}t} T_{\theta_t}(z) \right\rangle \right\} \mathrm{d}\rho_{\text{ref}}(z) \mathrm{d}t = \\
&= \int\limits_{\mathbb{R}^d} \int\limits_0^1 \left\{ \frac{\mathrm{d}}{\mathrm{d}t} \eta(t, T_{\theta_t}(z)) \right\} \mathrm{d}t \mathrm{d}\rho_{\text{ref}}(z) = \int\limits_{\mathbb{R}^d} \left\{ \eta(1, T_{\theta_t}(z)) - \eta(0, T_{\theta_t}(z)) \right\} \mathrm{d}\rho_{\text{ref}}(z) = 0
\end{aligned}$$

since η has compact support in $[0;1] \times \mathbb{R}^d$ and thus $\eta(1, T_{\theta_t}(z)) \equiv \eta(0, T_{\theta_t}(z)) \equiv 0$. Due to this, one can formulate the following

Proposition B.1. *The geodesic distance on Θ gives an upper bound for the Wasserstein distance between parametrized meatures.*

Proof. Using the Benamou-Brenier theorem, the fact that $(\rho_{\theta_t}, \hat{v}_{\theta_t})$ satisfies the continuity equation and equation B.1.3, one gets

$$\begin{aligned}
W_2^2(\rho_{\theta_0},\ \rho_{\theta_1}) = \inf_{\partial_t \rho_t + \nabla(\rho_t v_t)} \int \frac{1}{2} \|v_t\|_2^2 \mathrm{d}\rho_t \mathrm{d}t \leq \\
\leq \inf_{\theta_t} \int \frac{1}{2} \|u_{\theta_t}(\dot{\theta}_t)\|_2^2 \mathrm{d}\rho_{\theta_t} \mathrm{d}t = \inf_{\theta_t} \int \frac{1}{2} \langle \dot{\theta}_t, \dot{\theta}_t \rangle_\theta \mathrm{d}t = W^2_{2,\Theta}(\theta_0,\ \theta_1)
\end{aligned}$$

Thus

$$W_2^2(\rho_{\theta_0},\ \rho_{\theta_1}) \leq W^2_{2,\Theta}(\rho_{\theta_0},\ \rho_{\theta_1}) \tag{B.1.4}$$

□

B.2 Constructing retraction and vector transport

Let us denote by $B_R^2 := \{\theta : \|\theta - \theta_*\|_2 < R\}$ a Euclidean and by $\mathcal{B}_r^2 := \{\theta : W_{2,\Theta}(\theta, \theta_*) < r\}$ a geodesic ball, with the center in θ_*. We make the following local nondegeneracy assumption

Assumption B.1. *Assume a that there exists a Euclidean ball B_R^2 and constants $\lambda_G \leq \Lambda_G$ such that*

$$\lambda_G \,\mathrm{Id} \preceq G(\theta) \preceq \Lambda_G \,\mathrm{Id} \quad \forall \theta \in B_R^2 \tag{B.2.1}$$

We would later require the two following technical statements

Proposition B.2. *Let $\theta_1,\ \theta_2 \in \mathcal{B}^W_r$. Then the whole geodesic $\theta^g_t \in \mathcal{B}^W_{2r}$*

Proof. By application of triangle inequality, for every t

$$W_{2,\Theta}(\theta^*, \theta^g_t) \leq$$
$$\leq r + \min(W_{2,\Theta}(\theta_1, \theta^g_t), W_{2,\Theta}(\theta_2, \theta^g_t)) \leq r + \frac{1}{2} W_{2,\Theta}(\theta_1, \theta_2) \leq$$
$$\leq r + \frac{1}{2}\left(W_{2,\Theta}(\theta_1, \theta_*) + W_{2,\Theta}(\theta_*, \theta_2)\right) \leq 2r$$

□

Proposition B.3. *Under Assumption 3.1, for $r < \sqrt{\frac{\lambda_G}{2}} R$, it holds that $\mathcal{B}^W_r \subset B^2_R$*

Proof. Consider the point $\theta : \|\theta - \theta_*\|_2 > R$. For every continuous curve θ_t that connects θ_* and θ, there exists a time t' where the curve first leaves the ball B^2_R. We argue that

$$W^2_{2,\Theta}(\theta, \theta_*) = \inf_{\theta_t} \int_{t=0}^{t=1} \frac{1}{2}\|\dot\theta_t\|^2_{\theta_t} \mathrm{d}t \geq \inf_{\theta_t} \int_{t=0}^{t=t'} \frac{1}{2}\|\dot\theta_t\|^2_{\theta_t} \mathrm{d}t \geq \lambda_G \inf_{\theta_t} \int_{t=0}^{t=t'} \frac{1}{2}\|\dot\theta_t\|^2_2 \mathrm{d}t = \lambda_G \|\theta_* - \theta'\|^2_2 = \frac{1}{2}\lambda_G R^2$$

Thus, all points outside B^2_R are at least $\sqrt{\frac{\lambda_G}{2}} R$ away from θ_* □

We now can verify the Assumption A.2, needed for convergence of RAM, for the trivial retraction mapping $\Delta\theta \mapsto \theta + \Delta\theta$

Proposition B.4. *With the canonical identification $\mathrm{Tan}\,\Theta \simeq \mathbb{R}^D$, the mapping $\mathcal{R}$, defined as*

$$\mathcal{R}_\theta(\Delta\theta) = \theta + \Delta\theta \tag{B.2.2}$$

is a valid retraction on $\mathcal{P}_\Theta$. Additionally, under Assumption 3.1, for $\mathcal{U} = \mathcal{B}_{2\tilde r}$, where $\tilde r < \frac{1}{2\sqrt{2}} \left(1 + \frac{\sqrt{\Lambda_G}}{2\sqrt{\lambda_G}}\right)^{-1} \sqrt{\lambda_G} R$, it holds that

$$W^2_{2,\Theta}(\theta, \mathcal{R}_\theta(\Delta\theta)) > \frac{\lambda_G}{\Lambda_G}\|\Delta\theta\|^2_\theta, \quad \forall \theta \in \mathcal{B}_{2\tilde r},\ \forall \Delta\theta : \|\Delta\theta\|_\theta \leq \tilde r \tag{B.2.3}$$

Proof. The first statement leads from (Absil et al., 2009, Proposition 4.1.2) with $\mathcal{N} = \{0\}$, and the diffeomorphism $\phi : (\theta, 0) \mapsto \theta$.

As for the second one, since $\tilde r < \frac{1}{2\sqrt{2}}\sqrt{\lambda_G} R$, it holds that $\|\theta - \theta_*\|_2 \leq \frac{1}{2} R$ due to Proposition B.3. Using Assumption 3.1, we argue that

$$\|\Delta\theta\|_2 \leq \frac{1}{\sqrt{\lambda_G}}\|\Delta\theta\|_\theta \leq \frac{\tilde r}{\sqrt{\lambda_G}} \leq \frac{1}{2} R$$

Combining, one shows that

$$\|\theta + \Delta\theta - \theta_*\|_2 \leq \|\theta - \theta_*\|_2 + \|\Delta\theta\|_2 \leq R \tag{B.2.4}$$

Thus, both θ and $\mathcal{R}_\theta(\Delta\theta)$ are inside B^2_R. Using the curve

$$\theta^e_t = \theta + t\Delta\theta \in B^2_R\ \forall t \in [0; 1],$$

we bound the Wasserstein distance from above:

$$W^2_{2,\Theta}(\theta, \theta + \Delta\theta) \leq \int \frac{1}{2}\|\Delta\theta\|^2_{\theta^e_t} \mathrm{d}t \leq \frac{1}{2}\Lambda_G \|\Delta\theta\|^2_2 \leq \frac{\Lambda_G}{2\lambda_G}\|\Delta\theta\|^2_\theta \leq \frac{\Lambda_G}{4\lambda_G}\tilde r^2$$

Consequently, $W_{2,\Theta}(\theta_*, \theta + \Delta\theta) \leq (1 + \frac{\sqrt{\Lambda_G}}{2\sqrt{\lambda_G}})\tilde r \leq \frac{1}{2\sqrt{2}}\sqrt{\lambda_G} R$. Thus, both θ and $\theta + \Delta\theta$ belong to the geodesic ball $\mathcal{B}^W_{\frac{1}{\sqrt{2}}\sqrt{\lambda_G} R}$. According to Proposition B.2, the whole $W_{2,\Theta}$-geodesic between them,

denoted by θ_t^g, lies in $\mathcal{B}^W_{\sqrt{\frac{\lambda_G}{2}}R}$. Due to Proposition B.3, $\theta_t^g \in B^W_{\sqrt{\lambda_G}R} \subset B^2_R$, where Assumption 3.1 holds. We can thus lower bound the distance by

$$W^2_{2,\Theta}(\theta, \theta+\Delta\theta) = \int \|\dot\theta_t^g\|^2_{\theta_t^g}\mathrm{d}t \geq \lambda_G \int \|\dot\theta_t^g\|_2^2\mathrm{d}t \geq \lambda_G\|\Delta\theta\|_2^2 \geq \frac{\lambda_G}{\Lambda_G}\|\Delta\theta\|_\theta^2$$

Taking the square root, we establish

$$W_{2,\Theta}\left(\theta, \mathcal{R}_\theta(\Delta\theta)\right) \geq \sqrt{\frac{\lambda_G}{\Lambda_G}}\|\Delta\theta\|_\theta$$

□

We can check if the identity map will be a suitable vector transport in the sense of Assumption A.3.

Proposition B.5. *Under Assumption 3.1, $\mathcal{T}^{\theta_2}_{\theta_1} = \mathrm{Id}$ satisfies the condition in Assumption A.3 with constant $M = \sqrt{\frac{\Lambda_G}{\lambda_G}}$.*

Proof. Let $\Delta\theta_1 \in \mathrm{Tan}_{\theta_1}\Theta$. Then $\Delta\theta_2 := \mathcal{T}^{\theta_2}_{\theta_1}\Delta\theta_1 = \Delta\theta_1$. From equation B.2.1

$$\|\Delta\theta_2\|^2_{\theta_2} \leq \Lambda_G\|\Delta\theta_2\|_2^2 = \Lambda_G\|\Delta\theta_1\|_2^2 \leq \frac{\Lambda_G}{\lambda_G}\|\Delta\theta_1\|^2_{\theta_1}$$

Taking square root, one establishes

$$\left\|\mathcal{T}^{\theta_2}_{\theta_1}\Delta\theta_1\right\|_{\theta_2} \leq \sqrt{\frac{\Lambda_G}{\lambda_G}}\|\Delta\theta_1\|_{\theta_1}$$

□

B.3 Relation between fixed points on $\mathcal{P}_2(\mathbb{R}^d)$ and $\mathcal{P}_\Theta$

Following (Wu et al., 2025, Lemma 3.14), define $\mathcal{K}_\theta : L^2(\rho_{\text{ref}};\mathbb{R}^d) \to L^2(\rho_{\text{ref}};\mathbb{R}^d)$ by

$$\mathcal{K}_\theta[f](z') := \partial_\theta T_\theta(z')G^\dagger(\theta)\int \partial_\theta T_\theta(z)^T f(z)\mathrm{d}\rho_{\text{ref}}(z). \tag{B.3.1}$$

Then $\mathcal{K}_\theta$ is the $L^2(\rho_{\text{ref}})$-orthogonal projection onto the parametric tangent space

$$Q_\theta := \mathrm{span}\left\{\partial_{\theta_j}T_\theta(\cdot) : j = 1,\dots,D\right\}.$$

Proposition B.6 (Proposition 2.1)**.** *Under Assumption 2.1, the Wasserstein gradient is controlled by the parametric fixed-point residual:*

$$\|\partial_W\mathcal{E}(\rho_\theta)\|^2_{L^2(\rho_\theta)} \leq \frac{1}{h^2}\|r_h(\theta)\|^2_\theta + \delta_0^2. \tag{B.3.2}$$

Moreover, if Assumption 2.2 holds, then

$$\mathcal{E}(\rho_{\theta_k}) - \mathcal{E}(\rho_*) \leq \frac{1}{\mu}\left(\frac{\|r_h(\theta_k)\|^2_{\theta_k}}{h^2} + \delta_0^2\right). \tag{B.3.3}$$

Consequently, decay of the fixed-point residual controls the squared Wasserstein-gradient norm up to the parametric projection error δ_0^2.

Proposition 2.1. Since $\mathcal{K}_\theta$ is the orthogonal projection, by the Pythagorean theorem, for an arbitrary θ

$$\|\partial_W\mathcal{E}(\rho_\theta)\|^2_{L_2(\rho_\theta)} = \|\mathcal{K}_\theta\partial_W\mathcal{E}(\rho_\theta)\|^2_{L_2(\rho_\theta)} + \|(\mathrm{Id}-\mathcal{K}_\theta)\partial_W\mathcal{E}(\rho_\theta)\|^2_{L_2(\rho_\theta)}$$

By definition of the pullback metric, $\|\mathcal{K}_\theta\partial_W\mathcal{E}(\rho_\theta)\|_{L_2(\rho_\theta)} = \|G^\dagger(\theta)\nabla_\theta E(\theta)\| = \frac{1}{h}\|r_k\|_\theta$. The second term can be bound by the approximation error from Assumption 2.1, yielding

$$\|\partial_W\mathcal{E}(\rho_\theta)\|^2_{L^2(\rho_\theta)} \leq \frac{1}{h^2}\|r_h(\theta)\|^2_\theta + \delta_0^2.$$

Combining with the Polyak-Łojasiewicz condition from Assumption 2.2, one gets

$$\mathcal{E}(\rho_{\theta_k}) - \mathcal{E}(\rho_*) \leq \frac{1}{\mu}\|\partial_W\mathcal{E}(\rho_{\theta_k})\|^2_{L^2(\rho_{\theta_k})} \leq \frac{1}{\mu}\left(\frac{\|r_k\|^2_{\theta_k}}{h^2} + \delta_0^2\right)$$

□

C ADDITIONAL NUMERIC RESULTS

In Table 3 we report the neural network and RAM parameters to reproduce our experiments.

Table 3: Network architectures and optimizer parameters used in the experiments.

Problem		n_{hidden}	D_{hidden}	Activation	$\mathrm{d}t$	h	Λ	N_{CG}
Sampling	Styblinsky–Tang	2	128	tanh	0.1	0.001	10^{-1}	50
	Double banana	2	16	tanh	0.1	0.001	10^{-1}	50
Aggregation		2	32	tanh	0.25	5×10^{-3}	5×10^{-4}	20
BLR		2	128	tanh	0.25	1×10^{-3}	5×10^{-2}	20
Generative	Checkerboard	3	128	swish	0.1	0.001	10^{-3}	50
	2 spirals	3	128	swish	0.1	0.001	10^{-3}	50
	8 gaussians	3	128	swish	0.1	0.001	10^{-5}	50

Remark C.1 (Compatibility of comparative figures)**.** *For Figures figs. 5a and 6a, we use the same fixed set of reference samples $z_{\text{test}} \sim \rho_{ref}$ for all methods and evaluate the corresponding trained maps on this common input set. This makes the qualitative comparison between NGD and RAM attributable to the learned maps, not to sampling variability.*

C.1 MINIMIZATION OF KL FUNCTIONAL

We consider the problem of minimization of the Kullback-Leibler divergence to a given distribution, known only up to an unnormalized density function. In this setting the posterior density has the form

$$\rho_*(x) = \frac{1}{Z} e^{-V(x)}$$

with the normalization constant $Z = \int e^{-V(x)} \mathrm{d}x$ being unknown. In the parametric pushforward setting, one can see that

$$\begin{aligned} \mathrm{KL}(\rho_\theta \mid \rho_*) &:= \mathbb{E}_{x\sim\rho_\theta} \left[\log \rho_\theta(x) - \log \rho_*(x)\right] \\ &= \mathbb{E}_{x\sim\rho_\theta} \left[\log \rho_\theta(x) + V(x) + \log Z\right] \\ &= \mathbb{E}_{z\sim\rho_{\text{ref}}} \left[\log \rho_\theta(T_\theta(z)) + V(T_\theta(z)) + \log Z\right]. \end{aligned}$$

Since $\log Z$ does not depend on the parameters, it can be omitted from the loss function. As for the current density ρ_θ, it can be written out as

$$\log \rho_\theta(T_\theta(z)) = \log \rho_{\text{ref}}(z)) - \log |\det JT_\theta(z)|$$

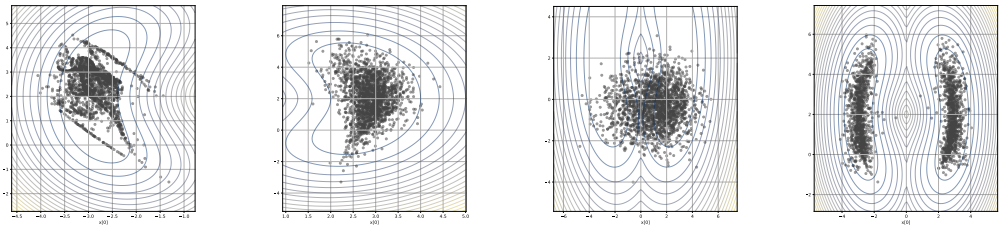
where T_θ is the parametric map and JT_θ stands for its Jacobian w.r.t. x. When using Neural ODE models, the latter term can be efficiently estimated by solving an augmented ODE and using Hutchinson trace estimator, as shown in Grathwohl et al. (2018). Thus, we optimize the loss function

$$\mathcal{E}(\rho_\theta) = \mathop{\mathbb{E}}_{z\sim\rho_{\text{ref}}} (\log \rho_{\text{ref}}(z)) - \log |\det JT_\theta(z)| + V(T_\theta(z))) \tag{C.1.1}$$

This problem can be viewed as a model problem for Bayesian inversion, the important setup in uncertainty quantification. Consider a system described by parameters x, with prior distribution $\rho_0(x)$. One is given noisy measurements of the system of the form $y = F(x) + \varepsilon$, where F is a known model and ε is the noise, whose distribution $l(\varepsilon) = l(y|F(x))$ is known. Then the Bayes theorem gives the posterior density ρ_∞ up to normalization constant.

$$\log \rho_\infty(x) = \log l(y|F(x)) + \log \rho_0(x) + const.$$

Normally, the forward model F, or likelihood l and prior ρ_0, contain computationally challenging Bui-Thanh et al. (2013); Petra et al. (2014); Dinkel et al. (2024); Cockayne et al. (2017); Aghanim et al. (2020) or data-driven components Liu et al. (2024a). For example, in PDE-based inverse problems, computing F requires solving a PDE for each set of parameters x. Thus, the posterior calls are the most computationally heavy parts in computing equation C.1.1, and direct comparison of iteration



(a) $h = 10^{-3}$ (b) $h = 10^{-4}$ (c) $h = 10^{-5}$ (d) RAM (reference)

Figure 8: (a-c): Scatter plots for Adam after 10^4 iterations with varied learning rate h. Mode collapse and numerical artifacts are clearly present in the samples. (d) Solution with RAM exhibits correct structure and relative weight of the modes.

complexity for NGD, RAM with SGD and Adam in our findings is justified. Moreover, the pointwise knowledge of the unnormalized posterior does not directly allow to estimate statistical quantites of interest such as moments or quantiles of the parameters. One has to resort to sampling from ρ_∞, which can require millions of posterior calls.

The characteristic difficulties, encountered in Bayesian inverse problems are anisotropy, concentration, non-convexity and multimodality of the posterior distribution. We choose the test distributions to reflect these properties, in accord with a multitude of previous works.

Styblinski-Tang potential The potential is defined as

$$V(x) = \frac{3}{50}\left(\sum_{i=1}^{d} x_i^4 - 16x_i^2 + 5x_i\right).$$

Since the potential is separable, its local minima in dimension d are obtained by choosing, independently for each coordinate, one of the two one-dimensional minima $s_- \approx -2.90$ and $s_+ \approx 2.75$. Hence the potential has 2^d local minima. These minima fall into $d+1$ energy levels: if k coordinates are equal to s_+ and $d-k$ are equal to s_-, then the energy is

$$V_k = \frac{3}{50}\left[(d-k)v(s_-) + kv(s_+)\right], \qquad v(s) = s^4 - 16s^2 + 5s.$$

There are $\binom{d}{k}$ minima at level V_k. The global minimum is $(s_-, \dots, s_-)$, while $(s_+, \dots, s_+)$ is the shallowest local minimum. Thus, the target distribution becomes increasingly multimodal as d grows, with exponentially many modes organized into finitely many energy levels. Figure 2a shows a cross-section of the potential and generated samples along the first and second dimensions.

Double banana The distribution is defined as

$$\rho_*(x) \propto e^{-2(\|(x-s)\|-3)^2} \cdot \left\{e^{-2((x_1-s_1)-3)^2} + e^{-2((x_1-s_1)+3)^2}\right\}$$

This is a bimodal distribution with a nonconvex nonsmooth potential. $s = [s_1,\ s_2]$ is a shift vector that we set to $[2, 0]$ in order to evaluate the methods' ability to approximate non-centered distributions.

The performance of NGD, RAM in comparison to standard methods (SGD and Adam) is presented on Figure 7. In our preliminary experiments we have noted that it is hard to select a learning rate such that the reference methods produce a distribution with correct number of modes. In particular, for Adam on the double-banana target, values larger than $h = 10^{-5}$ suffered from mode collapse

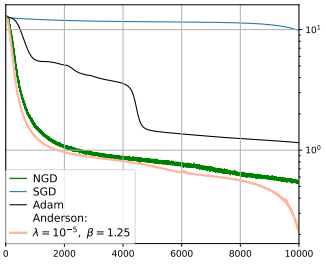


Figure 7: RAM and NGD compared with SGD and Adam on DB.

C.1.1 Bayesian inverse problem for the heat equation

We use the setup and the numerical implementation from Roberts et al. (2026). A problem for a heat equation on the interval $[0, 1]$ with mixed boundary conditions

$$\frac{\partial}{\partial x}\left[\kappa \frac{\partial v}{\partial x} v\right] = -s, \quad \begin{cases} v(1) = 1 \\ \kappa(0)\frac{\partial v}{\partial x}(0) = 0 \end{cases}$$

Table 4: Dataset statistics for the Bayesian logistic regression benchmark.

Dataset	Number of features	Dataset size
banana	2	5300
diabetes	8	768
german	20	1000
image	18	2086
ringnorm	20	7400
splice	60	2991
twonorm	20	7400
waveform	21	5000

is considered. The prior distribution of $\log \kappa$ is approximated by a Gaussian process, and only d first components of the Karhunen-oève expansion are considered

$$\log \kappa(X) \coloneqq 1 + \sum_{r=1}^{d} \sqrt{\lambda_r}\, u_j \psi_j, \qquad u_j \overset{\text{iid}}{\sim} \mathcal{N}(0,1),$$

This gives a distribution ρ_∞ in $\mathbb{R}^d$. For each set of expansion coefficients, the PDE is solved using a finite-difference approximation. Compared to the original implementation, we increased the distribution dimension d from 6 to 10, decreased the standard deviation of the likelihood from 0.01 to 0.001, and increased the number of grid nodes in the finite-difference solver from 100 to $1,250$. This has made the problem more challenging both in terms of time per iteration and the number of iterations required for convergence.

C.2 Bayesian logistic regression

We use the Bayesian logistic-regression benchmark datasets from Mika et al. (1999) Table 4, following the experimental protocol of Fan et al. (2022); Mokrov et al. (2021). Given a dataset $\mathcal{S} = \{s_i\}_{i=1}^M$ and a prior distribution $p_0(x)$ over the model parameters $x \in \mathbb{R}^d$, the posterior satisfies

$$\rho_\infty(x) \coloneqq p(x \mid \mathcal{S}) = \frac{p_0(x)p(\mathcal{S} \mid x)}{p(\mathcal{S})} \propto p_0(x)p(\mathcal{S} \mid x) = p_0(x) \prod_{i=1}^{M} p(s_i \mid x).$$

The parameter vector is $x = (w, \alpha)$, where w denotes the regression weights with conditional prior $p_0(w \mid \alpha) = \mathcal{N}(w, \alpha^{-1} I)$, and α is assigned a Gamma prior, $\alpha \sim \mathrm{Gamma}(1, 0.01)$. Each dataset is split into training and test sets using a $4{:}1$ ratio. Inference is performed using the training set, and posterior predictive performance is evaluated on the held-out test set.

For RAM, we sweep over the memory parameter m, relaxation parameter β, and regularization parameter λ. Figure 4 reports the top five RAM configurations for each dataset over three random seeds. For the predictive metrics in Table 5, we report the representative configuration $(m, \beta, \lambda) = (12, 2.0, 0.01)$, which appears among the top-performing configurations across all datasets.

Given posterior samples $x_j = (w_j, \alpha_j) \sim \rho_\theta$, we approximate the Bayesian predictive probability on a test input a by posterior averaging,

$$\widehat{p}(b = 1 \mid a) = \frac{1}{N} \sum_{j=1}^{N} \sigma(a^\top w_j).$$

We report classification accuracy and average test log-likelihood,

$$\frac{1}{M_{\text{test}}} \sum_{i=1}^{M_{\text{test}}} \left[b_i \log \widehat{p}(b = 1 \mid a_i) + (1 - b_i) \log\left(1 - \widehat{p}(b = 1 \mid a_i)\right) \right].$$

The predictive metrics in Table 5 show that RAM maintains test accuracy and log-likelihood comparable to NGD on this benchmark. The main effect of RAM is therefore not a different terminal predictive model, but faster optimization of the parametric posterior approximation, as shown by the convergence curves in Figure 4.

Table 5: Bayesian logistic-regression test performance. We report test accuracy and average test log-likelihood. RAM uses the representative configuration $(m, \beta, \lambda) = (12, 2.0, 0.01)$. VWGF and JKO-ICNN results are from Fan et al. (2022); SVGD results are from Mokrov et al. (2021).

	Accuracy ↑					Log-likelihood ↑				
Dataset	NGD	RAM	VWGF	JKO-ICNN	SVGD	NGD	RAM	VWGF	JKO-ICNN	SVGD
banana	0.565	0.540	0.550	0.550	0.540	-0.688	-0.692	-0.690	-0.690	-0.690
diabetes	0.753	0.753	0.786	0.775	0.780	-0.558	-0.557	-0.450	-0.450	-0.460
german	0.742	0.750	0.670	0.670	0.650	-0.552	-0.561	-0.590	-0.600	-0.600
image	0.821	0.820	0.866	0.820	0.815	-0.435	-0.435	-0.394	-0.430	-0.440
ringnorm	0.758	0.764	0.730	0.740	0.740	-0.520	-0.509	-0.500	-0.500	-0.500
splice	0.836	0.841	0.840	0.845	0.850	-0.373	-0.362	-0.380	-0.360	-0.355
twonorm	0.985	0.985	0.982	0.980	0.980	-0.048	-0.048	-0.056	-0.059	-0.062
waveform	0.779	0.779	0.785	0.780	0.765	-0.474	-0.471	-0.455	-0.485	-0.465

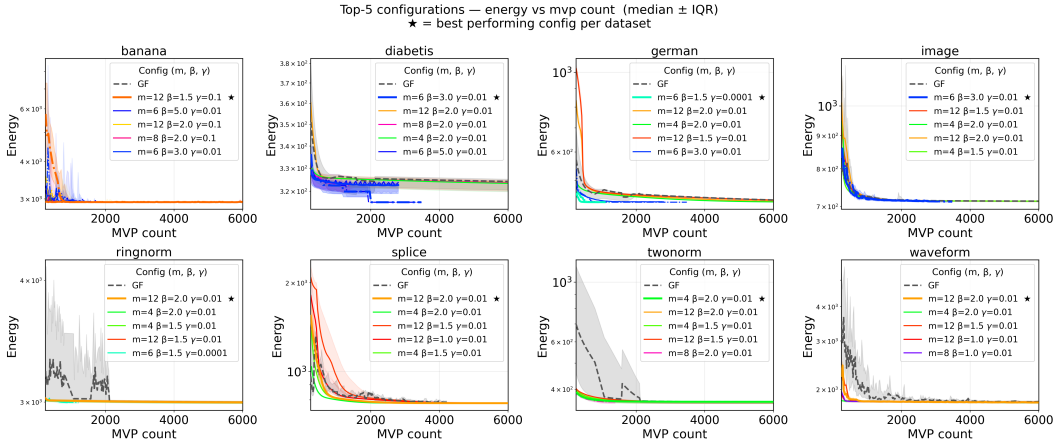


Figure 9: Bayesian logistic regression convergence as a function of MVP. RAM reaches a low-energy regime with fewer MVPs.

C.3 Analysis of Assumption 3.1

To numerically probe Assumption B.2.1, we estimate the extremal eigenvalues of the empirical pullback metric in a neighborhood of a computed critical point $\theta_\star$. We first flatten the parameter into a vector representation and sample perturbations of the form $\theta = \theta_\star + ru$, where u is a random unit vector and r ranges over a prescribed set of radii. For each perturbed parameter value, we evaluate matrix-vector products with $G(\theta)$ without explicitly forming the metric matrix: given a direction v, the product $G(\theta)v$ is computed through the metric MVP routine, equivalently using the Jacobian-vector and vector-Jacobian structure of the pullback metric. We then apply Lanczos iterations to this matrix-free operator to estimate $\lambda_{\min}(G(\theta))$ and $\lambda_{\max}(G(\theta))$. Repeating this procedure over 150 random perturbations at each radius provides empirical lower and upper spectral bounds for $G(\theta)$ on the sampled neighborhood around $\theta_\star$. These estimates suggest that the assumption is numerically consistent with the observed local geometry of the parameterization.