

---

# Asymptotically exact variational flows via involutive MCMC kernels

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1 Most expressive variational families—such as normalizing flows—lack practical  
2 convergence guarantees, as their theoretical assurances typically hold only at the  
3 intractable global optimum. In this work, we present a general recipe for construct-  
4 ing tuning-free, asymptotically exact variational flows from involutive MCMC  
5 kernels. The core methodological component is a novel representation of gen-  
6 eral involutive MCMC kernels as invertible, measure-preserving *iterated random*  
7 *function* systems, which act as the flow maps of our variational flows. This leads  
8 to three new variational families with provable total variation convergence. Our  
9 framework resolves key practical limitations of existing variational families with  
10 similar guarantees (e.g., MixFlows), while requiring substantially weaker theo-  
11 retical assumptions. Finally, we demonstrate the competitive performance of  
12 our flows across tasks including posterior approximation, Monte Carlo estimates,  
13 and normalization constant estimation, outperforming or matching state-of-the-art  
14 MCMC and black-box normalizing flows.

## 15 1 Introduction

16 Variational inference (VI) [1–3] is a general methodology for approximate probabilistic inference,  
17 where the goal is to approximate a target distribution (e.g., a Bayesian posterior) within a specified  
18 variational family. This variational family is typically chosen to be a parametric family that enables  
19 *tractable inference*—allowing for i.i.d. sampling and density evaluation [2–7]. This tractability  
20 offers key benefits: it enables the evaluation and optimization of approximation quality via unbiased  
21 estimates of the evidence lower bound (ELBO) [3], which corresponds to the Kullback-Leibler (KL)  
22 divergence [8] to the target distribution up to a constant. Moreover, it facilitates downstream tasks  
23 such as importance sampling [9, 10] and normalization constant estimation.

24 The quality of a variational approximation is fundamentally determined by the expressiveness of  
25 the variational family. Significant progress has been made in developing flexible families, including  
26 boosted mixtures [11–16] and normalizing flows [4, 6, 17–20]. These families often exhibit *universal*  
27 *approximation guarantees* [16, 21, 22]: as the number of mixture components or flow layers  
28 grows, the family becomes capable of approximating any distribution arbitrarily well under mild  
29 assumptions. However, a major limitation remains—theoretical guarantees pertain only to the *opti-*  
30 *mal* variational approximation, which is rarely obtained in practice due to the typical non-convexity  
31 of the optimization. In contrast, Markov chain Monte Carlo (MCMC) [23, 24; 25, Ch. 11, 12]  
32 is *asymptotically exact*, meaning it is guaranteed to produce arbitrarily accurate results given suffi-  
33 cient computation for any valid choice of tuning parameters (though some values may yield higher  
34 efficiency than others).

35 Xu et al. [26] introduced the first asymptotically exact variational family—MixFlow—that does  
36 not require optimal tuning. A MixFlow is constructed by averaging pushforwards of a reference

distribution under repeated application of an invertible map. When this map is both *ergodic* and *measure-preserving* (e.m.p) with respect to the target distribution  $\pi$ , MixFlows converge to  $\pi$  in total variation as the number of steps increases, while retaining the tractability of standard variational inference. However, its practical applicability is limited by the challenge of designing an invertible  $\pi$ -e.m.p. map for general continuous targets (several solutions exist for discrete spaces [27, 28]). The main obstacles are: (1) continuous e.m.p. maps often involve simulation of ODEs, which requires discretized numerical methods that destroy the e.m.p. property; (2) exactness often requires discrete Metropolis–Hastings (MH) corrections that are not invertible; and (3) proving ergodicity of such maps is very challenging. For example, Xu et al. [26] proposed a map based on the *uncorrected* Hamiltonian Monte Carlo (HMC), which is neither exactly measure-preserving nor provably ergodic. Other existing Hamiltonian-based methods [29] also suffer from discretization error and are non-ergodic [30]. Attempts via deterministic Gibbs samplers based on measure-preserving ODEs [31] or CDF/inverse-CDF transformations [27] are also limited by the intractability of computing the exact transformations. MH corrections used to restore exactness [27, 32] result in non-invertible transformations due to the accept-reject mechanism; recall that invertibility is required by variational flows to enable tractable density evaluation. To date, there is no framework for constructing variational families whose *practical implementation* achieves an MCMC-like asymptotic exactness.

In this work, we address the challenges mentioned above and propose a new framework for developing practical, asymptotically exact variational flows. Rather than relying on e.m.p dynamics as in MixFlow [26], our framework leverages *iterated random functions* (IRF)<sup>1</sup> [34]—a type of *random* dynamical system. The main contributions of this work are as follows:

1. We develop a method for deriving exact measure-preserving transformations from general *involutional* MCMC kernels [35, 36], while preserving invertibility of the transformation.
2. We introduce a more general framework for constructing asymptotically exact flows, leading to three novel variational families beyond the original MixFlow.
3. We establish total variation convergence guarantees for these new families under significantly weaker assumptions than those required in MixFlow theory [26], notably relaxing the ergodicity conditions of the flow maps.

## 2 Background

Throughout, let  $\pi$  be the target distribution on a measurable space  $(\mathcal{X}, \mathcal{B})$ , where  $\mathcal{X} \subseteq \mathbb{R}^d$ . With some abuse of notation, we use the same symbol to denote both a distribution and its density.

### 2.1 Deterministic MixFlows

A mixed variational flow (MixFlow) [26, 28] is built from a *deterministic*,  $\pi$ -ergodic (Definition D.1) and measure-preserving (e.m.p.) diffeomorphism  $f$ . Given such a map  $f$  and a reference distribution  $q_0$  on  $\mathcal{X}$  that enables i.i.d. sampling and density evaluation, the MixFlow density is given by

$$\forall x \in \mathcal{X}, \quad \bar{q}_T(x) = \frac{1}{T} \sum_{t=0}^{T-1} f^t q_0(x) = \frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0(f^{-t}x)}{\prod_{i=1}^t J(f^{-i}x)}, \quad J(x) = |\det \nabla_x f(x)|, \quad (1)$$

where  $f^t x$  and  $f^t q_0$  denote mapping  $x$  or pushing  $q_0$  through  $t$  iterations of  $f$ . Eq. (1) is tractable if  $f^{-1}$  and the Jacobian  $J$  can be evaluated. To generate  $X \sim \bar{q}_T$ , we first draw  $X_0 \sim q_0$  and a flow length  $K \sim \text{Unif}\{0, 1, \dots, T-1\}$ , and then map  $X_0$  through  $K$  iterations of  $f$ , i.e.,  $X = f^K(X_0)$ . Theorem C.1 shows that  $\bar{q}_T$  converges in both density and total variation (TV) to  $\pi$  as  $T$  increases. Since  $\bar{q}_T$  is built from a deterministic e.m.p dynamical system, we label it a *deterministic MixFlow*, to distinguish it from our proposed *random* dynamical system flows (see Section 3).

In practice, the map  $f$  is typically designed to mimic familiar MCMC kernels [26, 28], so that its trajectories have similar statistical behavior to the corresponding Markov chain. However, general constructions of *exact* e.m.p. MixFlow maps for continuous target distributions remain unavailable. As discussed in the introduction, both the exact measure-preservation and the ergodicity are highly non-trivial to obtain in practice. As a result, practitioners often resort to approximate maps, creating a gap between theory and implementation. These approximations can introduce numerical instability and degraded performance as  $T$  grows [37].

<sup>1</sup>IRFs are also referred to as iterated function systems (IFS) in some literature, e.g., [33].

## 85 2.2 Involutive MCMC

86 An involutive MCMC kernel [35, 36] is a Metropolis-type Markov kernel with a deterministic pro-  
 87 posal given by an *involution*  $g$ , i.e., a self-inverse function satisfying  $g = g^{-1}$  (Algorithm 1 in Ap-  
 88 pendix A). Consider an auxiliary variable  $v$  defined on a space  $\mathcal{V}$ , with conditional density  $\rho(v | x)$   
 89 given  $x \in \mathcal{X}$  with respect to a fixed dominating measure on  $\mathcal{V}$ , and the augmented target density  
 90  $\bar{\pi}(x, v) := \pi(x)\rho(v|x)$ . For a differentiable involution  $g : \mathcal{X} \times \mathcal{V} \rightarrow \mathcal{X} \times \mathcal{V}$ , each transition from a  
 91 current state  $x$  proceeds in three steps:

- 92 1. Sample an auxiliary variable  $v \sim \rho(\cdot | x)$ ;
- 93 2. Propose a new state  $(x', v') = g(x, v)$ ;
- 94 3. Accept  $x'$  with probability  $\min \left( 1, \frac{\bar{\pi}(x', v')}{\bar{\pi}(x, v)} \left| \frac{\partial g(x, v)}{\partial (x, v)} \right| \right)$ .

95 This framework encompasses a broad class of MCMC algorithms, with many popular algorithms  
 96 appearing as special cases of involutive MCMC [36, 38–41] (see Appendix B).

## 97 2.3 Iterated random functions

98 An *iterated random function* (IRF) system [34] on  $\mathcal{X}$  consists of a sequence of *random* maps:

$$\forall t \in \mathbb{N}, X_{t+1} = f_{\theta_{t+1}}(X_t), \quad X_0 \in \mathcal{X}, \quad (\theta_t)_{t \in \mathbb{N}} \stackrel{\text{iid}}{\sim} \mu, \quad (2)$$

99 where  $\{f_\theta : \mathcal{X} \rightarrow \mathcal{X} : \theta \in \Theta\}$  is a set of parametrized functions, with each  $\theta$  drawn *randomly* from  
 100 a distribution  $\mu$  on the parameter space  $\Theta$ . The above IRF induces a Markov kernel given by:

$$\forall x \in \mathcal{X}, \quad \forall B \in \mathcal{B}, \quad P(x, B) := \int_{\Theta} \mathbb{1}_B(f_\theta(x)) \mu(d\theta). \quad (3)$$

101 Throughout this work, we focus on IRFs where the family  $\{f_\theta : \theta \in \Theta\}$  satisfies Assumption 2.1.

102 **Assumption 2.1.** *For  $\mu$ -a.s. all  $\theta \in \Theta$ ,  $f_\theta(\cdot)$  is bijective and  $\pi$ -measure-preserving. Furthermore,*  
 103  *$\pi$  is the unique invariant probability of the Markov kernel  $P$  induced by the IRF.*

104 Assumption 2.1 implies that the sequence of iterates  $X_t$  produced by the IRF behave like a  $\pi$ -  
 105 invariant, irreducible Markov chain. Therefore, long-run averages of IRF iterates converge to expec-  
 106 tations under  $\pi$ , following the standard law of large numbers (LLN) for MCMC [42], also known as  
 107 the *random Birkhoff ergodic theorem* in the IRF literature [43, 44].

108 **Theorem 2.2** (Random Birkhoff ergodic theorem [44, Cor. 2.2]). *Suppose that IRF  $f_\theta$  satisfies*  
 109 *Assumption 2.1. Then, for  $\pi$ -a.e.  $x \in \mathcal{X}$  and  $\mu$ -almost all  $(\theta_t)_{t \in \mathbb{N}}$ :*

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \phi(f_{\theta_t} \circ \dots \circ f_{\theta_1}(x)) \xrightarrow{L^1(\pi)} \mathbb{E}[\phi(X)], \quad X \sim \pi,$$

110 *and the same result holds for the above equation with  $f_{\theta_t}^{-1}$  instead of  $f_{\theta_t}$ .*

## 111 3 Variational flows based on IRFs

112 In this section, we provide a methodology that turns any IRF system that satisfies Assumption 2.1  
 113 into *asymptotically exact* variational families. We introduce three variational families—each built  
 114 from the same IRF but combined in a different way—and prove that all three converge to the target in  
 115 total variation. All proofs are deferred to Appendix D. Throughout, we assume access to a reference  
 116 distribution  $q_0$  that permits i.i.d. sampling and tractable density evaluation.

### 117 3.1 IRF MixFlows

118 An IRF MixFlow is a mixture of pushforwards of a reference  $q_0$  through an IRF sequence:

$$\vec{q}_T := \frac{1}{T} \sum_{t=0}^{T-1} f_{\theta_t} \circ \dots \circ f_{\theta_1} q_0,$$

where  $\theta_1, \dots, \theta_T$  is a *cached* i.i.d. sequence drawn from  $\mu$ . When constructing the flow, we first sample and freeze the random stream  $\theta_1, \dots, \theta_T$ , yielding an *inhomogeneous* sequence of  $T$  parameterized bijections. Then to draw  $X \sim \vec{q}_T$ , we treat  $\vec{q}_T$  as a mixture of  $T$  distributions:

$$K \sim \text{Unif}\{0, 1, \dots, T-1\} \quad X_0 \sim q_0 \quad X = f_{\theta_K} \circ \dots \circ f_{\theta_1}(X_0)$$

Note crucially that each sample  $X$  is generated using the *same frozen* sequence  $\theta_1, \dots, \theta_T$ . For density evaluation, we compute the inverse IRF  $f_{\theta_T}^{-1}, \dots, f_{\theta_1}^{-1}$ . Because each  $f_{\theta}$  is  $\pi$ -measure-preserving, by Lemma C.2, the density takes a similar form as in a deterministic MixFlow (Eq. (6)):

$$\vec{q}_T(x) = \pi(x) \cdot \frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi} (f_{\theta_1}^{-1} \circ \dots \circ f_{\theta_t}^{-1}(x)), \quad \forall x \in \mathcal{X}.$$

However, note that this density requires simulating the *backward process* of the inverse IRF ([34])

$$\overleftarrow{X}_t(x) := f_{\theta_1}^{-1} \circ \dots \circ f_{\theta_t}^{-1}(x) \quad \text{for } t \in [T],$$

which cannot be computed sequentially. As a result, IRF MixFlows incur a quadratic density evaluation cost  $O(T^2)$ . Fortunately, this backward process can be computed in a parallel fashion, as the computation of each  $\overleftarrow{X}_t(x)$ ,  $t \in [T]$  is independent. We recommend deploying IRF MixFlows on modern parallel hardware (e.g., GPUs) for efficient density evaluation.

IRF MixFlows share the total variation convergence guarantee (Theorem 3.1) of deterministic MixFlows. The proof (Appendix D.2.1) is similar to the original MixFlow argument [26, Theorem 4.2], interpreting the IRF (Eq. (2)) as a time-homogeneous, e.m.p. dynamical system over the joint space  $\Theta^{\mathbb{N}} \times \mathcal{X}$ . However, we emphasize that Assumption 2.1 is significantly weaker than the ergodicity assumption of Theorem C.1. See Section 3.4 for a detailed discussion.

**Theorem 3.1.** *Let  $\mathbb{P}$  denote the joint distribution over the i.i.d. sequence  $(\theta_t)_{t \in \mathbb{N}} \stackrel{iid}{\sim} \mu$ . If Assumption 2.1 holds and  $q_0 \ll \pi$ , then*

$$\text{TV}(\vec{q}_T, \pi) \xrightarrow{\mathbb{P}} 0 \quad \text{as } T \rightarrow \infty. \quad (4)$$

### 3.2 Backward IRF MixFlows

To address the  $O(T^2)$  density cost of IRF MixFlows, we propose a simple modification: constructing the flow from the *backward process*. Specifically, we define the *backward IRF MixFlow* as:

$$\overleftarrow{q}_T := \frac{1}{T} \sum_{t=0}^{T-1} f_{\theta_1} \circ \dots \circ f_{\theta_t} q_0.$$

This construction retains  $O(T)$  complexity of sampling  $X \sim \overleftarrow{q}_T$  via:

$$K \sim \text{Unif}\{0, 1, \dots, T-1\} \quad X_0 \sim q_0 \quad X = f_{\theta_1} \circ \dots \circ f_{\theta_K}(X_0),$$

while reducing the density computation cost to  $O(T)$ . The density of  $\overleftarrow{q}_T$  is given by:

$$\overleftarrow{q}_T(x) = \pi(x) \cdot \frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi} (f_{\theta_t}^{-1} \circ \dots \circ f_{\theta_1}^{-1}(x)), \quad \forall x \in \mathcal{X}. \quad (5)$$

This mirrors the density formula of deterministic MixFlows (Eq. (6)), enabling the use of the random ergodic theorem (Theorem 2.2) to establish the same pointwise and total variation convergence.

**Theorem 3.2.** *If Assumption 2.1 holds and  $q_0 \ll \pi$ , then for  $\pi$ -a.e.  $x \in \mathcal{X}$  and  $\mu$ -almost all  $(\theta_t)_{t \in \mathbb{N}}$ :*

$$\overleftarrow{q}_T(x) \rightarrow \pi(x) \quad \text{and} \quad \text{TV}(\overleftarrow{q}_T, \pi) \rightarrow 0 \quad \text{as } T \rightarrow \infty.$$

### 3.3 Ensemble IRF MixFlows

All MixFlow variants discussed above—including deterministic MixFlows—are based on *ergodic averaging* along the flow. This inherently limits their convergence rate to  $O(1/T)$ , as the initial reference  $q_0$  always retains a  $1/T$  mixing weight. In contrast, MCMC methods often exhibit geometric convergence in their marginal distributions under suitable conditions [42; 45, Ch. 15]. Motivated



by this, we propose the *ensemble IRF MixFlows*, which instead uses an *ensemble average* of the endpoint of multiple IRF trajectories in an attempt to match  $T$ -step MCMC marginal distribution:

$$\tilde{q}_T^{(M)} := \frac{1}{M} \sum_{m=1}^M q_T^{(m)} = \frac{1}{M} \sum_{m=1}^M f_{\theta_T^{(m)}} \circ \cdots \circ f_{\theta_1^{(m)}} q_0,$$

where each  $\theta_1^{(m)}, \dots, \theta_T^{(m)}$  corresponds to an independent IRF realization. As in the case of the previous MixFlows, the  $M$  streams of randomness  $\theta_t^{(m)}$  are cached (i.e., *frozen*) when sampling and computing densities. The resulting density of the ensemble IRF MixFlow is given by:

$$\tilde{q}_T^{(M)}(x) = \pi(x) \cdot \frac{1}{M} \sum_{m=1}^M \frac{q_0}{\pi} \left( f_{\theta_1^{(m)}}^{-1} \circ \cdots \circ f_{\theta_T^{(m)}}^{-1}(x) \right),$$

whose computation costs  $O(TM)$  (or  $O(T)$  when parallelized across the  $M$  streams). Drawing  $X \sim \tilde{q}_T^{(M)}$  takes  $O(T + M)$  operations:

$$K \sim \text{Unif}\{1, \dots, M\} \quad X_0 \sim q_0 \quad X = f_{\theta_T^{(K)}} \circ \cdots \circ f_{\theta_1^{(K)}} q_0.$$

Intuitively, the flow length  $T$  controls the bias of the IRF system, while the ensemble size  $M$  controls the variance of the Monte Carlo average. This tradeoff is formalized in the following result.

**Theorem 3.3.** *Suppose that Assumption 2.1 holds, and that  $\forall x \in \mathcal{X}, \frac{q_0}{\pi}(x) \leq B < \infty$ . Then,*

$$\begin{aligned} \mathbb{E}_\theta \left[ \text{TV} \left( \tilde{q}_T^{(M)}, \pi \right) \right] &\leq \frac{1}{\sqrt{M}} \mathbb{E} \left[ \sqrt{\text{Var}_{\theta_{1:T}} \left[ \frac{q_0}{\pi} \left( f_{\theta_1}^{-1} \circ \cdots \circ f_{\theta_T}^{-1}(X) \right) \mid X \right]} \right] \\ &\quad + B \cdot \mathbb{E} \left[ \text{TV}(P^T \delta_X, \pi) \right], \quad X \sim \pi, \end{aligned}$$

where  $P$  is the Markov kernel for the IRF  $\{f_\theta\}$ , and  $\delta_X$  is the Dirac measure at  $X$ .

In the setting where  $\text{TV}(P^T \delta_X, \pi) = O(\rho^T)$  for some  $\rho \in (0, 1)$ , the convergence rate of  $\tilde{q}_T^{(M)}$  can be heuristically characterized as  $\text{TV}(\tilde{q}_T^{(M)}, \pi) = O\left(\rho^T \vee \frac{1}{\sqrt{M}}\right)$ , capturing the tradeoff between the bias (via  $T$ ) and variance (via  $M$ ). Given a fixed computational budget, choosing the balance between flow length  $T$  and ensemble size  $M$  is critical. In the extreme case of  $M = 1$ , convergence will fail entirely—any  $\pi$ -measure-preserving  $f$  satisfies  $\text{TV}(fq, \pi) = \text{TV}(q, \pi)$  [46, Theorem 1]. On the other hand, small  $T$  leads to high bias due to insufficient mixing. This tradeoff closely relates to recent studies on parallel MCMC algorithms [47, 48].

### 3.4 Discussion

**Relaxing ergodicity.** A major advantage of IRF-based MixFlows over deterministic MixFlows is that IRF-based MixFlows require only that the kernel  $P$  admits a unique invariant distribution (Assumption 2.1), a significantly weaker condition than the ergodicity assumed by deterministic MixFlows. In fact, whenever the set  $\Theta^* := \{\theta : f_\theta \text{ is } \pi\text{-ergodic}\}$  has positive  $\mu$ -measure, Assumption 2.1 automatically holds [33, Corollary 3.3]. Uniqueness of the invariant distribution is also easily verified by checking that  $P$  is irreducible [42, 45]. The IRFs we construct in Section 4 correspond to involutive MCMC kernels that are known to be irreducible, whereas establishing ergodicity in deterministic MixFlows is typically so difficult that it is assumed without proof [26, 29, 49].

**Which flow to choose?** All four flows are asymptotically exact, yet their density formulae reveal different bias-variance and cost-accuracy tradeoffs. In every case the density ratio takes the form  $\frac{\text{flow density}}{\pi}(x) = \frac{1}{K} \sum_{k=1}^K \frac{q_0}{\pi}(T_k(x))$ , where  $T_k$  is a composition of inverse IRF/ergodic maps. Hence practical convergence of each flow is dictated by how quickly  $\frac{1}{K} \sum_{k=1}^K \frac{q_0}{\pi}(T_k(x))$  converges to a constant. Empirically (see Appendix E.1.1) we find that IRF MixFlows often reach a given accuracy at shorter flow lengths than deterministic or backward IRF MixFlows, but a full theoretical comparison study is deferred to future work.

## 4 Invertible measure-preserving IRF from involutive MCMC

In this section, we provide a concrete, general construction of invertible and exactly measure-preserving IRFs based on involutive MCMC kernels. The key idea, originally developed for the MH sampler [27, 32], is to further augment the space with two additional variables  $u_v \in [0, 1]^d, u_a \in [0, 1]$ . The variable  $u_v$  pairs with the auxiliary variable  $v$ , and  $u_a$  encodes the randomness in the accept/reject decision. Let the augmented target  $\bar{\pi}$  and space  $\mathcal{S}$  be defined as:

$$\bar{\pi}(s) = \pi(x)\rho(v | x)\mathbb{1}_{[0,1]^d}(u_v)\mathbb{1}_{[0,1]}(u_a), \quad s = (x, v, u_v, u_a) \in \mathcal{S} := \mathcal{X} \times \mathcal{V} \times [0, 1]^d \times [0, 1].$$

The two uniform auxiliary variables  $u_v$  and  $u_a$  will be refreshed with two random parameters  $(\theta_v, \theta_a) \sim \mu = \text{Unif}[0, 1]^d \times \text{Unif}[0, 1]$ . Without loss of generality, we describe the IRF construction assuming a one-dimensional target  $\pi(x)$ . The IRF  $f_\theta(s) := f_\theta(x, v, u_v, u_a)$  is defined by the following steps (Algorithm 2):

1. Uniform auxiliary refreshment:  $u_v \leftarrow (u_v + \theta_v) \bmod 1, \quad u_a \leftarrow (u_a + \theta_a) \bmod 1$
2. Update  $(v, u_v)$  pair via CDF/inverse-CDF of  $\rho(\cdot|x)$ <sup>2</sup>:  $u'_v \leftarrow F_{\rho(\cdot|x)}(v), \quad \tilde{v} \leftarrow F_{\rho(\cdot|x)}^{-1}(u_v)$
3. Propose and compute the MH-ratio:  $(x', v') \leftarrow g(x, \tilde{v}), \quad r \leftarrow \frac{\bar{\pi}(x', v')}{\bar{\pi}(x, \tilde{v})} \left| \frac{\partial g(x, \tilde{v})}{\partial x, \tilde{v}} \right|$
4. Accept or reject: If  $u_a > r$ , reject and stay at the *pre-involution* state  $s' = (x, \tilde{v}, u'_v, u_a)$ . Otherwise, set  $u'_a \leftarrow \frac{u_a}{r}$  and accept the *post-involution* state  $s' = (x', v', u'_v, u'_a)$ .

The correspondence with involutive MCMC (Algorithm 1) is: Step 2 simulates  $v \sim \rho(\cdot|x)$  via inverse CDF sampling, Step 3 mirrors the involution and MH ratio computation, and Step 4 performs the accept/reject step while explicitly tracking the randomness  $u_a$  involved in the decision. Furthermore, one can use this map in a deterministic MixFlow by simply fixing  $\theta_v, \theta_a$  to some pre-specified constant values (rather than sampling from  $\mu$ ). And finally, using the same Jacobian computation as in [32, Eq. (25)], one can show that  $\forall \theta = (\theta_v, \theta_a) \in \Theta$ , the IRF  $f_\theta$  (Algorithm 2) is  $\bar{\pi}$ -measure-preserving.

**Proposition 4.1.** *The map given by Algorithm 2 satisfies Assumption 2.1 for  $\bar{\pi}$  if its induced Markov kernel  $P$  is irreducible.*

As discussed previously, one must be able to compute  $f_\theta^{-1}(\cdot)$  if  $f_\theta$  is to be used in a MixFlow. Steps 1–3 are straightforward to invert. The main challenge lies in inverting the accept/reject Step 4—we need to recover the accept/reject decision based solely on the output state  $s'$ . Depending on different decisions,  $s'$  could either be the pre-involution state  $(x, \tilde{v}, u'_v, u_a)$  or the post-involution state  $(x', v', u'_v, u'_a)$ . Since the transformation  $u'_a \leftarrow u_a/r$  only present in the acceptance branch, inferring the branch incorrectly would lead to the failure of recovering  $u_a$ .

We address this challenge (pseudocode in Algorithm 3). by exploiting the self-inverse property of the involution  $g$ . First note that  $g(x, \tilde{v}) = (x', v')$  and  $g(x', v') = (x, \tilde{v})$ . Suppose that  $s' = (x^\#, v^\#, u^\#, u_a^\#)$ .  $\{g(x^\#, v^\#), (x^\#, v^\#)\}$  is exactly the unordered pair  $\{(x, \tilde{v}), (x', v')\}$ . Then from the property of the Jacobian of  $g$ , we observe that

$$\frac{\pi(x', v')}{\pi(x, \tilde{v})} \left| \frac{\partial g(x, \tilde{v})}{\partial x, \tilde{v}} \right| = \left( \frac{\pi(x, \tilde{v})}{\pi(x', v')} \left| \frac{\partial g(x', v')}{\partial x', v'} \right| \right)^{-1}.$$

Hence, recomputing the MH-ratio as in Step 3 yields

$$\tilde{r} := \frac{\pi(x^\#, v^\#)}{\pi(g(x^\#, v^\#))} \cdot J_g(g(x^\#, v^\#)) \in \{r, r^{-1}\},$$

where  $r$  corresponds to the true MH-ratio as computed in the forward pass and  $J_g(x, v) = \left| \frac{\partial g(x, v)}{\partial x, v} \right|$ .

The key observation to infer the accept/reject decision then follows: If  $u_a^\# \cdot \tilde{r} < 1$ , then the acceptance branch was taken, so  $u_a = u_a^\# \cdot \tilde{r}$ ; otherwise the move was rejected as  $u_a$  cannot be larger than 1.

<sup>2</sup>Typically,  $\rho(v|x)$  lies in a simple family; for instance, in HMC with a diagonal mass matrix,  $\rho$  is a diagonal Gaussian, whose CDF and inverse-CDF can be computed stably. For multidimensional  $v$ , a Gibbs-style update on the conditionals of  $\rho(\cdot|x)$  can be used.

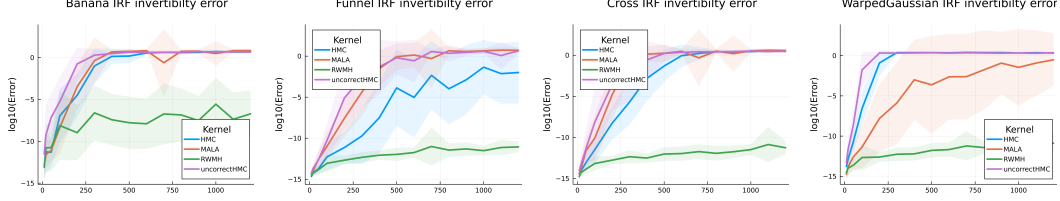


Figure 1: Inversion error of IRFs (based on HMC, uncorrected HMC, MALA, and RWMH) over increasing flow length  $T$ . Verticle axis shows the 2-norm error of reconstructing  $s = (x, v, u_v, u_a)$  ( $s = (x, v)$  for the uncorrected HMC IRF) sampled from  $q_0$  by the composing the forward simulation  $f_{\theta_T} \circ \dots \circ f_{\theta_1}(s)$  and its inverse. The lines indicate the mean, and error regions indicate the standard deviation over 32 independent initializations from  $q_0$ .

Fig. 1 empirically verifies that one can successfully invert the proposed IRF map for four MCMC-based IRFs—HMC, uncorrected HMC, MALA, and RWMH—on four synthetic targets listed in Appendix E. The same hyperparameters are used for every example: each (uncorrected) HMC transition consists of 50 leapfrog steps with step size 0.02; MALA uses step size 0.25; RWMH uses step size 0.3. We evaluate the 2-norm error of reconstructing  $s = (x, v, u_v, u_a)$  sampled from a mean-field Gaussian variational approximation  $q_0$  by the composing the forward simulation  $f_{\theta_T} \circ \dots \circ f_{\theta_1}(s)$  and its inverse. Both HMC variants and MALA remain reliably invertible up to  $T \approx 200$  iterations, while the RWMH IRF remain invertible up to  $T \approx 1000$  iterations. Notably, the corrected HMC IRF is consistently more stable than its uncorrected counterpart used in past MixFlows work, because the additional MH acceptreject step discards trajectories with large numerical error that would otherwise cause the dynamics to diverge. Although Algorithm 2 and Algorithm 3 are exact inverses in theory, floating-point round-off accumulates with  $T$  and can eventually fail the exact reconstruction [26, 37]. In practice, however, the resulting statistical error in downstream variational inference is often negligible, thanks to the *shadowing* property of chaotic dynamical systems [37].

## 5 Experiments

This section presents an empirical evaluation of the four proposed flows (three IRF flows and deterministic MixFlows, all referred to as “IRF flows” for brevity, since deterministic MixFlows can be viewed as a special case). We compare these to two black-box normalizing flows (RealNVP [19] and Neural Spline Flow (NSF) [50]), and to the No-U-Turn sampler (NUTS) [51]. Variational methods are evaluated via their (i) ELBO, (ii) normalization-constant ( $\log Z$ ) estimate accuracy, and (iii) importance sampling effective sample size (ESS) [52–54]. Sampling methods are evaluated via their Monte Carlo estimation error. In all cases, all flows start from the same reference distribution  $q_0$ : a mean-field Gaussian trained for 10K Adam steps with batch size 10 and learning rate  $10^{-3}$ . All IRF flows are evaluated with 64 i.i.d. draws, while normalizing flows use 1024. Full experimental details appear in Appendix E.

### 5.1 Synthetic examples

Our synthetic experiments consist of four 2-dimensional targets used by Xu et al. [26]: the Banana [55], Neal’s funnel [56], a cross-shaped Gaussian mixture, and a warped Gaussian distribution. Fig. 2 shows a comparison of the original Hamiltonian-MixFlow—built on an *uncorrected* HMC kernel—with our *corrected* version including the MH step. For each target we run both flows with identical hyper-parameters (50 leapfrog steps per transition, several step-sizes) and estimate the total-variation (TV) distance to the ground truth using 512 i.i.d. samples. Across all targets and step-sizes, the corrected HMC-based MixFlow consistently achieves lower TV error and remains robust as the step-size grows. In contrast, the uncorrected variant often deteriorates with longer flows because the inexact map error accumulates (e.g., the green dashed curve in the third panel). At larger step sizes the uncorrected flow frequently diverges, producing NaNs (marked by crosses), whereas the corrected flow remains stable—echoing the inversion stability results in Fig. 1.

We next compare the four IRF flows with RealNVP and NSF. Two IRF variants are examined: HMC-based (50 leapfrog steps per transition;  $T = 200$ ) and RWMH-based ( $T = 4000$ ). Each normalizing flow consists 6 flow layers, and is trained via 50,000 Adam steps with batch size 32; we tune the

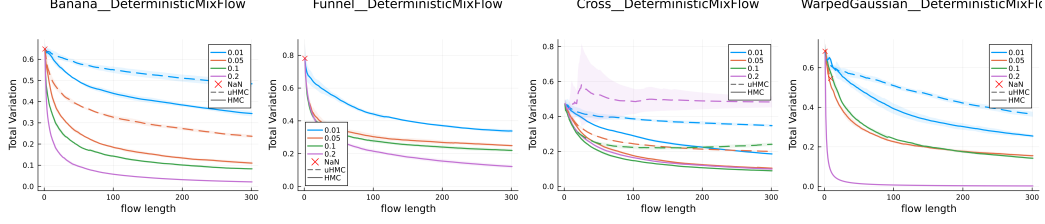


Figure 2: Total-variation error for deterministic MixFlow built on *corrected* (solid) versus *uncorrected* (dashed) HMC kernels, plotted against flow length  $T$  for several step sizes. Each curve is the mean over 32 independent runs; shaded bands ( $\pm 1$  SD) show run-to-run variability. A cross marks any setting where at least one run returned a NaN (instability), at which point the trace is terminated.

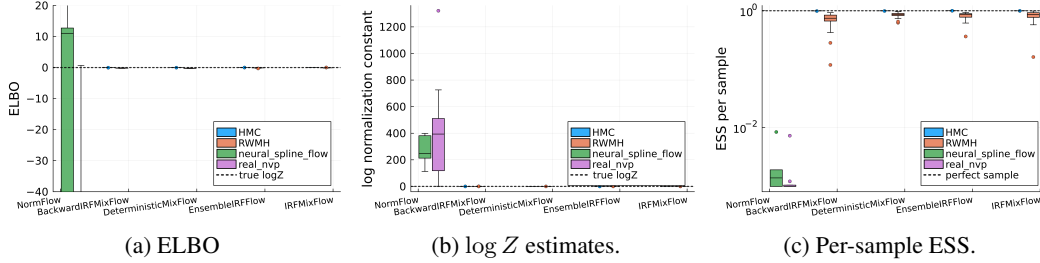


Figure 3: Variational approximation quality of IRF Flows versus Rea1NVP and NSF. Box plots for IRF flows are based on 32 independent runs, and 10 runs for the normalizing flows. The black dashed line in (c) indicates the optimal ESS of perfect i.i.d. samples.

learning rates in the grid  $\{10^{-4}, 10^{-3}, 10^{-2}\}$ , and report the results of the setting with smallest median TV distance over 5 runs. Additional implementation details can be found in Appendix E.1.

Figs. 3a and 3b display the ELBO and  $\log Z$  estimates (via importance sampling) for the Banana target; the remaining synthetic cases show the same pattern (Fig. 7 in Appendix E.1.3). As synthetic targets are normalized, a perfect variational approximation has both metrics near 0. The IRF flows meet this mark consistently across runs, whereas Rea1NVP and NSF exhibit high variability and often produce extreme ELBO or  $\log Z$  values. We restrict the vertical range of the ELBO plot for better visualization; full-range plots are in Fig. 7b. We also note that training instability is common for the normalizing flows: on the Funnel example, 10 of 15 Rea1NVP runs and all NSF runs diverged.

Fig. 3c further examine the per-sample importance sampling ESS (see Fig. 7d on similar results for other examples), which reflects the  $\chi^2$  divergence from the variational distribution to the target [57]. The ESS is orders of magnitude higher for IRF flows than for the normalizing flows. Additionally, we provide comparisons among the three ergodic averaging MixFlow variants in Appendix E.1.1, and ensemble-size/length trade-offs for ensemble IRF MixFlows are explored in Appendix E.1.2.

## 5.2 Real-data experiments

The real-data experiments include the Student-t-regression (TReg; 4-dimensional), and the Sparse linear regression (SparseReg; 84-dimensional) from [26], and a latent Brownian motion model (Brownian; 32-dimensional) and the Log-Gaussian Cox process model (LGCP; 1600-dimensional) from the Inference Gym library [58]. Each normalizing flow is trained via 50,000 Adam steps of batchsize 32; we grid-search both the learning rates  $\{10^{-4}, 10^{-3}, 10^{-2}\}$  and flow layers  $\{6, 10\}$ , and report the configuration with the highest median ELBO over 5 runs. A additional mean-field Gaussian baseline is optimized for the same number of steps and batch size with learning rate  $10^{-3}$ .

All IRF variants use RWMH kernel, with the step size tuned to achieve an 0.8 acceptance rate using bisection search between 0.001 and 10. In each search step, we estimate acceptance rate by running 5,000 RWMH-IRF iterations. We set  $T = 5000$  for the backward IRF and deterministic MixFlow and ensemble IRF MixFlow, and set  $T = 4000$  for the IRF MixFlow. Normalizing flow results are

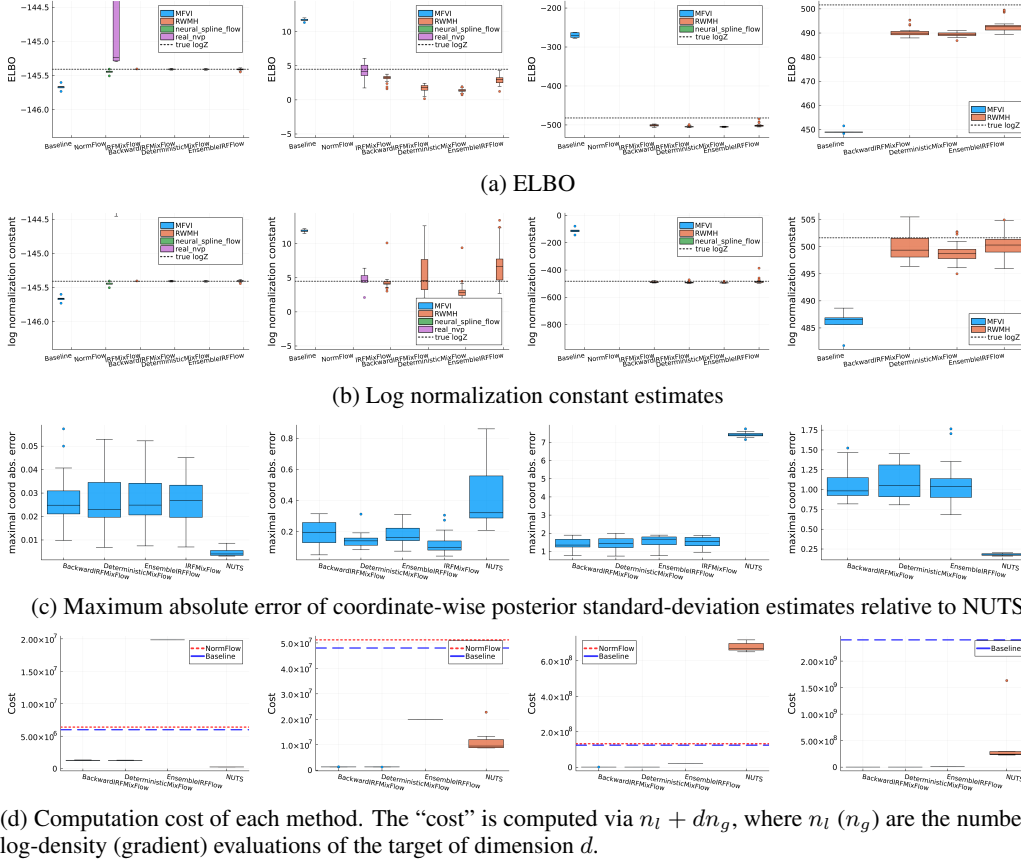


Figure 4: Results on real-data benchmarks (columns, from left to right): TReg( $d = 4$ ), Brownian( $d = 32$ ), SparseReg( $d = 82$ ), LGCP( $d = 1600$ )

omitted for LGCP, which did not finish training within 48 hours on the same hardware. Ground truth values are estimated using AIS with a dense temperature grid; see the details in Appendix E.2.

As in the synthetic experiments, our exact flows match—or modestly improve upon—the best-tuned ReaINVP and NSF in both ELBO (Fig. 4a) and log  $Z$  accuracy (Fig. 4b), and outperform the mean-field baseline by a wide margin. The per-sample importance-sampling ESS shows the same advantage (Fig. 8b). Crucially, normalizing flow training is orders of magnitude more expensive (Fig. 4d), whereas the exact flows achieve comparable accuracy at a fraction of the computational cost.

We further compare coordinate-wise posterior means (Fig. 8c in Appendix E.2) and standard deviations (Fig. 4c) against NUTS, reporting the maximum absolute error across dimensions relative to the estimated ground truth. NUTS is initialized with independent draws from  $q_0$  and run for 5,000 iterations including 2500 warm-up iterations. IRF flows outperform NUTS on two models and are slightly worse on the other two—yet they do so at lower computational cost (Fig. 4d). Note that the goal of this work is not to outperform MCMC, but rather to construct a variational family that provides asymptotic exactness and similar sampling performance; IRF MixFlows meet this standard.

## 6 Conclusion

We introduced a general framework for building asymptotically exact variational families from arbitrary involutive MCMC kernels. By constructing invertible, measure-preserving maps directly from these kernels, we overcome the main practical limitation of MixFlow [26] and enable the construction of a broad class of exact flows. We also provided a streamlined theoretical analysis for flows based on measure-preserving transformations and demonstrated their empirical advantages in density approximation and importance sampling. A promising direction is to pair our framework with recent automatic-tuning MCMC [38–41], developing truly tuning-free exact flows in practice.

## References

- [1] Michael Jordan, Zoubin Ghahramani, Tommi Jaakkola, and Lawrence Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.
- [2] Martin Wainwright and Michael Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.
- [3] David Blei, Alp Kucukelbir, and Jon McAuliffe. Variational inference: a review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.
- [4] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, 2015.
- [5] Rajesh Ranganath, Dustin Tran, and David Blei. Hierarchical variational models. In *International Conference on Machine Learning*, 2016.
- [6] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22:1–64, 2021.
- [7] Charles Margossian and Lawrence Saul. Variational inference in location-scale families: Exact recovery of the mean and correlation matrix. In *Advances in Neural Information Processing Systems*, 2024.
- [8] Solomon Kullback and Richard Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.
- [9] Herman Kahn and Andy W Marshall. Methods of reducing sample size in Monte Carlo computations. *Journal of the Operations Research Society of America*, 1(5):263–278, 1953.
- [10] Herman Kahn. Use of different Monte Carlo sampling techniques. Technical report, Rand Corporation, 1955.
- [11] Fangjian Guo, Xiangyu Wang, Kai Fan, Tamara Broderick, and David Dunson. Boosting variational inference. In *Advances in Neural Information Processing Systems*, 2016.
- [12] Andrew Miller, Nicholas Foti, and Ryan Adams. Variational boosting: iteratively refining posterior approximations. In *International Conference on Machine Learning*, 2017.
- [13] Xiangyu Wang. Boosting variational inference: theory and examples. Master’s thesis, Duke University, 2016.
- [14] Francesco Locatello, Rajiv Khanna, Joydeep Ghosh, and Gunnar Rätsch. Boosting variational inference: an optimization perspective. In *International Conference on Artificial Intelligence and Statistics*, 2018.
- [15] Francesco Locatello, Gideon Dresdner, Rajiv Khanna, Isabel Valera, and Gunnar Rätsch. Boosting black box variational inference. In *Advances in Neural Information Processing Systems*, 2018.
- [16] Trevor Campbell and Xinglong Li. Universal boosting variational inference. In *Advances in Neural Information Processing Systems*, 2019.
- [17] Ivan Kobyzev, Simon Prince, and Marcus Brubaker. Normalizing flows: an introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, 2021.
- [18] Abhinav Agrawal, Daniel Sheldon, and Justin Domke. Advances in black-box vi: Normalizing flows, importance weighting, and optimization. In *Advances in Neural Information Processing Systems*, 2020.
- [19] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. In *International Conference on Learning Representations*, 2017.
- [20] Rianne van den Berg, Leonard Hasenclever, Jakub Tomczak, and Max Welling. Sylvester normalizing flows for variational inference. In *Conference on Uncertainty in Artificial Intelligence*, 2018.
- [21] Frederic Koehler, Viraj Mehta, and Andrej Risteski. Representational aspects of depth and conditioning in normalizing flows. In *International Conference on Machine Learning*, 2021.
- [22] Zhifeng Kong and Kamalika Chaudhuri. The expressive power of a class of normalizing flow models. In *International Conference on Artificial Intelligence and Statistics*, 2020.



- [23] Christian Robert and George Casella. *Monte Carlo Statistical Methods*. Springer, 2nd edition, 2004.
- [24] Christian Robert and George Casella. A short history of Markov Chain Monte Carlo: subjective recollections from incomplete data. *Statistical Science*, 26(1):102–115, 2011.
- [25] Andrew Gelman, John Carlin, Hal Stern, David Dunson, Aki Vehtari, and Donald Rubin. *Bayesian data analysis*. CRC Press, 3rd edition, 2013.
- [26] Zuheng Xu, Naitong Chen, and Trevor Campbell. MixFlows: principled variational inference via mixed flows. In *International Conference on Machine Learning*, 2022.
- [27] Radford Neal. How to view an MCMC simulation as permutation, with applications to parallel simulation and improved importance sampling. *arXiv:1205.0070*, 2012.
- [28] Gian Carlo Diluvi, Benjamin Bloem-Reddy, and Trevor Campbell. Mixed variational flows for discrete variables. In *International Conference on Artificial Intelligence and Statistics*, 2024.
- [29] Greg van Steeg and Aram Galstyan. Hamiltonian dynamics with non-Newtonian momentum for rapid sampling. In *Advances in Neural Information Processing Systems*, 2021.
- [30] Jakob Robnik, G Bruno De Luca, Eva Silverstein, and Uroš Seljak. Microcanonical Hamiltonian monte carlo. *Journal of Machine Learning Research*, 24(311):1–34, 2023.
- [31] Kirill Neklyudov, Roberto Bondesan, and Max Welling. Deterministic gibbs sampling via ordinary differential equations. *arXiv:2106.10188*, 2021.
- [32] Iain Murray and Lloyd Elliott. Driving Markov chain Monte Carlo with a dependent random stream. *arXiv:1204.3187*, 2012.
- [33] Takehiko Morita. Deterministic version lemmas in ergodic theory of random dynamical systems. *Hiroshima mathematical journal*, 18:15–29, 1988.
- [34] Persi Diaconis and David Freedman. Iterated random functions. *SIAM review*, 41:45–76, 1999.
- [35] Luke Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22(4):1701–1728, 1994.
- [36] Kirill Neklyudov, Max Welling, Evgenii Egorov, and Dmitry Vetrov. Involutive MCMC: a unifying framework. In *International Conference on Machine Learning*, 2020.
- [37] Zuheng Xu and Trevor Campbell. Embracing the chaos: analysis and diagnosis of numerical instability in variational flows. In *Advances in Neural Information Processing Systems*, 2023.
- [38] Tiange Liu, Nikola Surjanovic, Miguel Biron-Lattes, Alexandre Bouchard-Côté, and Trevor Campbell. AutoStep: locally adaptive involutive MCMC. *arXiv:2410.18929*, 2024.
- [39] Miguel Biron-Lattes, Nikola Surjanovic, Saifuddin Syed, Trevor Campbell, and Alexandre Bouchard-Côté. autoMALA: Locally adaptive Metropolis-adjusted Langevin algorithm. In *International Conference on Artificial Intelligence and Statistics*, 2024.
- [40] Nawaf Bou-Rabee, Bob Carpenter, and Milo Marsden. GIST: Gibbs self-tuning for locally adaptive Hamiltonian Monte Carlo. *arXiv:2404.15253*, 2024.
- [41] Nawaf Bou-Rabee, Bob Carpenter, Tore Selland Kleppe, and Milo Marsden. Incorporating local step-size adaptivity into the No-U-Turn sampler using Gibbs self tuning. *arXiv:2408.08259*, 2024.
- [42] Gareth Roberts and Jeffrey Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.
- [43] Shizuo Kakutani. Random ergodic theorems and Markoff processes with a stable distribution. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, 1950.
- [44] Yuri Kifer. *Ergodic theory of random transformations*, volume 10. Springer Science & Business Media, 2012.
- [45] Randal Douc, Eric Moulines, Pierre Priouret, and Philippe Soulier. *Markov chains*. Springer, 2018.
- [46] Yu Qiao and Nobuaki Minematsu. A study on invariance of  $f$ -divergence and its application to speech recognition. *IEEE Transactions on Signal Processing*, 58(7):3884–3890, 2010.

- [47] Charles C Margossian, Matthew D Hoffman, Pavel Sountsov, Lionel Riou-Durand, Aki Vehtari, and Andrew Gelman. Nested  $\hat{R}$ : Assessing the convergence of Markov chain Monte Carlo when running many short chains. *Bayesian Analysis*, 1(1):1–28, 2024.
- [48] Pavel Sountsov, Colin Carroll, and Matthew D Hoffman. Running Markov chain Monte Carlo on modern hardware and software. *arXiv:2411.04260*, 2024.
- [49] Paul Tupper. Ergodicity and the numerical simulation of Hamiltonian systems. *SIAM Journal on Applied Dynamical Systems*, 4(3):563–587, 2005.
- [50] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, 2019.
- [51] Matthew Hoffman and Andrew Gelman. The No-U-Turn Sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.
- [52] Augustine Kong. A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Rep.*, 348:14, 1992.
- [53] Augustine Kong, Jun S Liu, and Wing Hung Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American statistical association*, 89(425):278–288, 1994.
- [54] Jun Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6:113–119, 1996.
- [55] Heikki Haario, Eero Saksman, and Johanna Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, pages 223–242, 2001.
- [56] Radford Neal. Slice sampling. *The Annals of Statistics*, 31(3):705–767, 2003.
- [57] Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, pages 405–431, 2017.
- [58] Pavel Sountsov, Alexey Radul, and contributors. Inference gym, 2020. URL [https://pypi.org/project/inference\\_gym](https://pypi.org/project/inference_gym).
- [59] Wilfred Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109, 1970.
- [60] Simon Duane, Anthony Kennedy, Brian Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics letters B*, 195(2):216–222, 1987.
- [61] Radford Neal. MCMC using Hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng, editors, *Handbook of Markov chain Monte Carlo*, chapter 5. CRC Press, 2011.
- [62] Tanja Eisner, Bálint Farkas, Markus Haase, and Rainer Nagel. *Operator Theoretic Aspects of Ergodic Theory*. Graduate Texts in Mathematics. Springer, 2015.
- [63] George Birkhoff. Proof of the ergodic theorem. *Proceedings of the National Academy of Sciences*, 17(12):656–660, 1931.
- [64] Kai Xu, Hong Ge, Will Tebbutt, Mohamed Tarek, Martin Trapp, and Zoubin Ghahramani. AdvancedHMC.jl: A robust, modular and efficient implementation of advanced HMC algorithms. In *Symposium on Advances in Approximate Bayesian Inference*, 2020.
- [65] Nicolas Chopin, Francesca Crucinio, and Anna Korba. A connection between tempering and entropic mirror descent. In *International Conference on Machine Learning*, 2024.
- [66] Saifuddin Syed, Alexandre Bouchard-Côté, Kevin Chern, and Arnaud Doucet. Optimised annealed sequential Monte Carlo samplers. *arXiv:2408.12057*, 2024.

## NeurIPS Paper Checklist

### 1. Claims

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

Answer: [\[Yes\]](#)

Justification: The main claim of this work is a general framework of constructing exact variational families, which is justified with precise theoretical results and empirical demonstrations.

Guidelines:

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

### 2. Limitations

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

Answer: [\[Yes\]](#)

Justification: Limitation of the methodology is discussed right after we introduce the method.

Guidelines:

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

### 3. Theory assumptions and proofs

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

Answer: [\[Yes\]](#)

Justification: We make precise statement of our theoretical results, and defer all the proofs to the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.

- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

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

Answer: [Yes]

Justification: We provide pseudo-code for our algorithm in the Appendix and include code and data in the supplementary materials.

Guidelines:

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

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

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

Answer: [Yes]

Justification: Code and data are provided as a Zip file in the supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so No is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

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

Answer: [\[Yes\]](#)

Justification: We describe high-level experimental setting in the experiment section (Section 5.), and defer additional details in the Appendix.

Guidelines:

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

## 7. Experiment statistical significance

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

Answer: [\[Yes\]](#)

Justification: Error bars of each figure is described in the caption of the figure or explained in the corresponding text discussing the figure.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

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

Answer: [\[Yes\]](#)

Justification: We provide informations of used computational resources in the Appendix, submitted as PDF in the supplementary materials.

Guidelines:

- The answer NA means that the paper does not include experiments.

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

## 9. Code of ethics

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

Answer: [Yes]

Justification:

Guidelines:

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

## 10. Broader impacts

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

Answer: [NA]

Justification: This paper focus on theoretical and methodological development in the field of computational Statistics. The societal consequences need not to be discussed for this work.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

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

Answer: [NA]

Justification: This paper focus on theoretical and methodological development in the field of computational Statistics, which poses no risk for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.



- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

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

Answer: [Yes]

Justification: We refer all the used models/code in the paper.

Guidelines:

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

## 13. New assets

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

Answer: [Yes]

Justification: Code and datasets for reproducing our results is documented and submitted in an anonymized zip file as supplementary materials.

Guidelines:

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

## 14. Crowdsourcing and research with human subjects

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

Answer: [NA]

Justification: This work does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

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

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

725 Answer: [NA]

726 Justification: This work does not involve crowdsourcing nor research with human subjects.

727 Guidelines:

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

738 **16. Declaration of LLM usage**

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

743 Answer: [NA]

744 Justification: The methodological development does not involve LLMs; the only LLM usage is for  
745 checking grammar mistakes and typos of the text.

746 Guidelines:

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

751	<b>Contents</b>	
752	<b>1 Introduction</b>	<b>1</b>
753	<b>2 Background</b>	<b>2</b>
754	2.1 Deterministic MixFlows . . . . .	2
755	2.2 Involutive MCMC . . . . .	3
756	2.3 Iterated random functions . . . . .	3
757	<b>3 Variational flows based on IRFs</b>	<b>3</b>
758	3.1 IRF MixFlows . . . . .	3
759	3.2 Backward IRF MixFlows . . . . .	4
760	3.3 Ensemble IRF MixFlows . . . . .	4
761	3.4 Discussion . . . . .	5
762	<b>4 Invertible measure-preserving IRF from involutive MCMC</b>	<b>6</b>
763	<b>5 Experiments</b>	<b>7</b>
764	5.1 Synthetic examples . . . . .	7
765	5.2 Real-data experiments . . . . .	8
766	<b>6 Conclusion</b>	<b>9</b>
767	<b>A Pseudocode</b>	<b>21</b>
768	A.1 Involution MCMC . . . . .	21
769	A.2 Involutive MCMC IRF . . . . .	22
770	A.3 Inverse IRF based on involutive MCMC . . . . .	22
771	<b>B Additional introduction to involutive MCMC</b>	<b>23</b>
772	B.1 Reversibility . . . . .	23
773	B.2 Examples . . . . .	23
774	<b>C Additional background materials</b>	<b>23</b>
775	C.1 Deterministic Mixflow . . . . .	23
776	C.2 Iterated random function . . . . .	24
777	<b>D Proofs</b>	<b>25</b>
778	D.1 Convergence of the deterministic MixFlow . . . . .	25
779	D.2 Convergence of the IRF MixFlow . . . . .	25
780	D.2.1 Convergence in the product space . . . . .	26
781	D.2.2 From the joint convergence to Theorem 3.1 . . . . .	27
782	D.3 Convergence of the backward IRF MixFlow . . . . .	28
783	D.4 Convergence of the ensemble IRF MixFlow . . . . .	28
784	D.5 Proof of Proposition 4.1 . . . . .	28
785	<b>E Additional experimental details</b>	<b>30</b>

786	E.1 Synthetic experiments . . . . .	30
787	E.1.1 Relative performance of deterministic, IRF, and backward IRF MixFlows . . . . .	31
788	E.1.2 Ensemble IRF MixFlows: scaling up $M$ or $T$ . . . . .	32
789	E.1.3 Additional results for synthetic examples . . . . .	33
790	E.2 Additional results for real-data experiments . . . . .	34

## 791 A Pseudocode

### 792 A.1 Involution MCMC

---

**Algorithm 1** Involution MCMC kernel  $K(x', v' | x, v)$

---

**Require:** current state  $x$ , target  $\pi$ , auxiliary distribution  $\rho(\mathrm{d}v|x)$ , involution  $g$

- 1:  $v \sim \rho(\mathrm{d}v|x)$  ▷ sample auxiliary variable
- 2:  $(x', v') \leftarrow g(x, v)$  ▷ generate proposal via the involution
- 3:  $\alpha \leftarrow \min \left( 1, \frac{\pi(x', v')}{\pi(x, v)} \left| \frac{\partial g(x, v)}{\partial x, v} \right| \right)$  ▷ compute the acceptance probability
- ▷ Accept or reject
- 4:  $u \sim \text{Unif}[0, 1]$
- 5: **if**  $u > \alpha$  **then**
- 6:  $x' \leftarrow x$  ▷ reject
- 7: **end if**
- 8: **return**  $x', v'$

---

**Algorithm 2** IRF based on involutive MCMC  $f_\theta(s)$ 


---

**Require:** joint state  $s = (x, v, u_v, u_a)$ , random parameters  $\theta = (\theta_v, \theta_a)$

- ▷ update uniform auxiliary variables
- 1:  $u_v \leftarrow (u_v + \theta_v) \bmod 1$
- 2:  $u_a \leftarrow (u_a + \theta_a) \bmod 1$
- ▷ involutive MCMC with target  $\pi(x)$ , auxiliary distribution  $\rho(\mathrm{d}v|x)$ , involution  $g$
- 3:  $u'_v \leftarrow F_{\rho(\cdot|x)}(v)$
- 4:  $\tilde{v} \leftarrow F_{\rho(\cdot|x)}^{-1}(u_v)$
- 5:  $(x', v') \leftarrow g(x, \tilde{v})$
- 6:  $r \leftarrow \frac{\bar{\pi}(x', v')}{\bar{\pi}(x, \tilde{v})} \left| \frac{\partial g(x, \tilde{v})}{\partial x, \tilde{v}} \right|$  ▷ Compute MH ratio
- 7: **if**  $u_a > r$  **then**
- 8:     **return**  $x, \tilde{v}, u'_v, u_a$  ▷ reject and return pre-involution state
- 9: **end if**
- 10:  $u'_a \leftarrow \frac{u_a}{r}$  ▷  $u_a \leq r$  implies that  $u'_a \in [0, 1]$
- 11: **return**  $x', v', u'_v, u'_a$  ▷ accept and return after-involution state

---

**Algorithm 3** Inverse IRF of involutive MCMC  $f_\theta^{-1}(s')$ 


---

**Require:** joint state  $s' = (x', v', u'_v, u'_a)$ , random parameters  $\theta = (\theta_v, \theta_a)$

- ▷ recover pre- and post-involution pair
- 1:  $(x, \tilde{v}) \leftarrow g(x', v')$
- ▷ this will either be  $r$  in line 6 of Algorithm 2 if accepted, or  $r^{-1}$  otherwise
- 2:  $\tilde{r} \leftarrow \frac{\bar{\pi}(x', v')}{\bar{\pi}(x, \tilde{v})} \left| \frac{\partial g(x, \tilde{v})}{\partial x, \tilde{v}} \right|$
- ▷ check accept or reject
- 3:  $u_a \leftarrow u'_a \cdot \tilde{r}$  ▷ update  $u_a$  (line 10 of Algorithm 2) as if the forward pass was an accept
- 4: **if**  $u_a > 1$  **then** ▷ forward pass was a reject (see line 6-7 of Algorithm 2)
- ▷ pre-involution state
- 5:      $(x, \tilde{v}) \leftarrow x', v'$
- 6:      $u_a \leftarrow u'_a$
- 7: **end if**
- ▷ inverse of line 3-4 of Algorithm 2
- 8:  $v \leftarrow F_{\rho(\cdot|x)}^{-1}(u_v)$
- 9:  $u_v \leftarrow F_{\rho(\cdot|x)}(\tilde{v})$
- ▷ inverse update of the uniform auxiliary variables (line 1-2 of Algorithm 2)
- 10:  $u_v \leftarrow (u_v + 1 - \theta_v) \bmod 1$
- 11:  $u_a \leftarrow (u_a + 1 - \theta_a) \bmod 1$
- 12: **return**  $x, v, u_v, u_a$

---



## B Additional introduction to involutive MCMC

### B.1 Reversibility

A key property of the involutive Markov kernel  $K$  is its *reversibility* with respect to both the augmented target  $\bar{\pi}(x, v)$  and its marginal  $\pi(x)$  [36, Prop. 2].

**Proposition B.1.** *The involutive MCMC kernel  $K(x', v'|x, v)$  (defined in Algorithm 1) satisfies that*

$$K(x', v'|x, v)\bar{\pi}(x, v) = K(x, v|x', v')\bar{\pi}(x', v'), \quad \hat{K}(x'|x)\pi(x) = \hat{K}(x|x')\pi(x'),$$

where  $\hat{K}$  is the marginalized kernel defined as:  $\hat{K}(x'|x) := \int \hat{K}(x', v' | x, v)\rho(\mathrm{d}v|x)\mathrm{d}v'$ .

### B.2 Examples

Here, we illustrate how the generic Metropolis-Hastings (MH) algorithm [35], random-walk Metropolis-Hastings (RWMH) [59], and Hamiltonian Monte Carlo (HMC) [60, 61], fit into this framework by specifying the corresponding auxiliary distribution  $\rho(\cdot|x)$  and the involution map  $g$ .

**Example B.2** (MH sampler; Section B.3. of [36]). *The Metropolis-Hastings sampler with proposal distribution  $\rho(\mathrm{d}x'|x)$  can be cast as an involutive MCMC method by defining the auxiliary distribution as  $\rho(\mathrm{d}v|x)$ , and using the swap involution  $g : (x, v) \mapsto (v, x)$ .*

**Example B.3** (RWMH sampler; Section 2. of [38]). *RWMH with step size  $\epsilon$  is obtained by setting*

$$g(x, v) = (x + \epsilon v, -v), \quad v \sim \rho(\mathrm{d}v|x) = \mathcal{N}(0, I).$$

**Example B.4** (HMC; Section B.6. of [36]). *In the involutive formulation of HMC, the auxiliary variable  $v$  corresponds to the momentum variable, and  $\rho(v|x)$  is the momentum distribution, typically a Gaussian distribution independent of  $x$ . The involution map  $g$  consists of applying  $k$  steps of the leapfrog integrator, followed by a momentum sign flip:*

$$g\left(\begin{bmatrix} x \\ v \end{bmatrix}\right) = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} L^k \left(\begin{bmatrix} x \\ v \end{bmatrix}\right),$$

where  $L : (x, v) \rightarrow (x', v')$  denotes a single leapfrog step (of step size  $\epsilon$ ) given by

$$\begin{aligned} v_{1/2} &\leftarrow v + \frac{\epsilon}{2} \nabla \log \pi(x) \\ x' &\leftarrow x + \epsilon v_{1/2} \\ v' &\leftarrow v_{1/2} + \frac{\epsilon}{2} \nabla \log \pi(x'). \end{aligned}$$

## C Additional background materials

To clarify notation: throughout this work, given a transformation  $f$  and a distribution  $p$ , we use  $f(p(x))$  to denote the function  $f$  evaluated at the value  $p(x)$ , and  $fp(x)$  to denote the density of the pushforward distribution  $fp$  evaluated at  $x$ .

### C.1 Deterministic Mixflow

Given a  $\pi$ -e.m.p. map  $f$ , the distribution  $\bar{q}_T$  converges to  $\pi$  in both density and total variation (TV) as  $T$  increases, regardless of any tuning of  $f$ . Proof of Theorem C.1 can be found in Appendix D.1.

**Theorem C.1.** *Suppose that  $f$  is a  $\pi$ -e.m.p diffeomorphism, and  $q_0 \ll \pi$ . Then, as  $T \rightarrow \infty$ ,*

$$\bar{q}_T(x) \rightarrow \pi(x), \quad \pi\text{-a.e. } x \in \mathcal{X}, \quad \text{and} \quad \mathrm{TV}(\bar{q}_T, \pi) \rightarrow 0.$$

The original convergence of MixFlow [26, Theorem 4.2] was established using general operator theory for e.m.p dynamical systems [62]. We provide an intuitive explanation by directly assessing the density of  $\bar{q}_T$  with an additional requirement that the mapping  $f$  is diffeomorphic (which is needed in practice for tractability anyway). This enables a substantially simplified proof of the convergence of deterministic MixFlow. The proof of Theorem C.1 can be found in Appendix D.1.

Using Lemma C.2, we can simplify the density of  $\bar{q}_T$ —Eq. (1)—to

$$\bar{q}_T(x) = \frac{1}{T} \sum_{t=0}^{T-1} f^t q_0(x) = \pi(x) \cdot \frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi}(f^{-t}(x)), \quad \forall x \in \mathcal{X}. \quad (6)$$

828 If  $f$  is  $\pi$ -e.m.p, the Birkhoff ergodic theorem [63; 62, p. 212] implies that  $\frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi}(f^{-t}(x)) \rightarrow 1$ . Con-  
 829 sequently, for  $\pi$ -a.e.  $x \in \mathcal{X}$ ,  $\bar{q}_T(x) \rightarrow \pi(x)$  as  $T \rightarrow \infty$ . Eq. (6) can be directly used for implementation as  
 830 the potentially unknown constant of  $\pi$  is canceled. This offers additional convenience for the implementation  
 831 because practitioners now do not need to track the Jacobian of each flow transformation.

832 We present here the key lemma that enables a simplified expression for density function of MixFlow.

833 **Lemma C.2.** *Let  $f : \mathcal{X} \rightarrow \mathcal{X}$  be a diffeomorphism,  $\pi$  be a probability distribution on  $\mathcal{X}$ , with density (denoted  
 834 by  $\pi(x)$ ) with respect to a dominating measure  $\lambda$ . Then,*

- 835 1.  *$f$  is  $\pi$ -measure-preserving if and only if  $f^{-1}$  is  $\pi$ -measure-preserving.*
- 836 2.  *$f$  is  $\pi$ -measure-preserving if and only if for  $\lambda$ -a.e.  $x \in \mathcal{X}$ ,  $J_f(x) := |\det \nabla f^{-1}(x)| = \frac{\pi(x)}{\pi(f^{-1}(x))}$ .*

837 *Proof of Lemma C.2.* By definition,  $f$  is  $\pi$ -preserving if and only if  $f\pi(x) = \pi(x) = f^{-1}\pi(x)$ . Examining  
 838 the density of the pushforward  $f\pi$  via the change-of-variable formula, we have

$$\forall x \in \mathcal{X}, \quad f\pi(x) = \pi(f^{-1}(x))J_f(x) = \pi(x) \Leftrightarrow J_f(x) = \frac{\pi(x)}{\pi(f^{-1}(x))}.$$

839 The second claim follows from the fact that  $\pi = (f \circ f^{-1})\pi = (f^{-1} \circ f)\pi$ . □

## 840 C.2 Iterated random function

841 As introduced in the main text, the IRF  $f_\theta$  induces a Markov kernel given by:

$$\forall x \in \mathcal{X}, \quad \forall B \in \mathcal{B}, \quad P(x, B) := \int_{\Theta} \mathbb{1}_B(f_\theta(x))\mu(d\theta).$$

842 This yields a simple characterization of the action of the Markov process  $P$  on a distribution  $q$ :

$$(Pq)(y) := \int_{\mathcal{X}} P(x, y)q(dx) = \mathbb{E}[f_\theta q(y)], \quad \theta \sim \mu, \quad f_\theta q: \text{pushforward of } q \text{ under } f_\theta.$$

843 We can further characterize the Markov kernel  $R(\cdot, \cdot)$  induced by the *inverse IRF*  $f_{\theta_t}^{-1}$ :

$$\forall x \in \mathcal{X}, \quad \forall A \in \mathcal{B}, \quad R(x, A) := \int_{\Theta} \mathbb{1}_A(f_\theta^{-1}(x))\mu(d\theta).$$

844 which is precisely the *reversal* of  $P(\cdot, \cdot)$ :

$$\pi \otimes P(A \times B) = \pi \otimes R(B \times A) = \int \pi(f_\theta(A) \cap B)\mu(d\theta), \tag{7}$$

845 where  $\pi \otimes P(A \times B) := \int_A P(x, B)\pi(dx)$ . See Kakutani [43, Eq. (4.5)] for the detailed derivation. Notice  
 846 that if  $P$  is reversible wrt  $\pi$ , i.e.,  $\pi \otimes P = \pi \otimes R$ , both the IRF  $f_\theta$  and its inverse  $f_\theta^{-1}$  induce the same Markov  
 847 process  $P$ . In other words,  $P = R$ . From Eq. (7), we can see that a sufficient and necessary condition so that  
 848  $P = Q$  is that

$$\int \pi(f_\theta(A) \cap B)\mu(d\theta) = \int \pi(f_\theta^{-1}(A) \cap B)\mu(d\theta).$$

849 Moreover, the forward IRF  $f_\theta$  and the inverse IRF  $f_\theta^{-1}$  are closely related. For instance, Assumption 2.1 holds  
 850 for  $f_\theta$  and its induced Markov process  $P$  if and only if Assumption 2.1 holds for  $f_\theta^{-1}$  and its induced  $R$  [43,  
 851 Theorem 3.].

## 852 D Proofs

### 853 D.1 Convergence of the deterministic MixFlow

854 **Definition D.1** (Ergodic map [62, pp. 73, 105]).  $f : \mathcal{X} \rightarrow \mathcal{X}$  is ergodic for  $\pi$  if for all measurable sets  $A \subseteq \mathcal{X}$ ,  
855  $f(A) = A$  implies that  $\pi(A) \in \{0, 1\}$ .

856 The most notable implication of a  $\pi$ -e.m.p  $f$  is that the long-run average of repeated applications of  $f$  converges  
857 to the expectation under  $\pi$ , a result known as the Birkhoff ergodic theorem [63; 62, p. 212]. The full statement  
858 is given in Theorem D.2.

859 **Theorem D.2** (Ergodic Theorem [63; 62, p. 212]). Suppose  $f : \mathcal{X} \rightarrow \mathcal{X}$  is measure-preserving and ergodic  
860 for  $\pi$ , and  $\phi \in L^1(\pi)$ . Then

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \phi(f^t x) = \int \phi d\pi, \quad \pi\text{-a.e. } x \in \mathcal{X}.$$

861 **Lemma D.3** (Scheffé's lemma). Let  $\phi_n$  be a sequence of integrable functions on a measure space  $(\mathcal{X}, \mathcal{B}, \pi)$   
862 that converges  $\pi$ -a.s. to  $\phi$ . Then

$$\int |\phi_n(x) - \phi(x)| \pi(dx) \rightarrow 0, \quad n \rightarrow \infty,$$

863 if and only if

$$\int |\phi_n(x)| \pi(dx) \rightarrow \int |\phi(x)| \pi(dx), \quad n \rightarrow \infty.$$

864 *Proof of Theorem C.1.* Note that the Jacobian of the  $\pi$ -e.m.p  $f$  is  $\pi(x)/\pi(f^{-1}(x))$  by Lemma C.2, allowing  
865 the density of  $\bar{q}_T$  to be expressed as:

$$\bar{q}_T(x) = \frac{1}{T} \sum_{t=0}^{T-1} f^t q_0(x) = \pi(x) \cdot \frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi}(f^{-t}(x)), \quad \forall x \in \mathcal{X}.$$

866 The pointwise density convergence is the direct consequence of Eq. (8). Specifically, provided  $q_0 \ll \pi$ , we  
867 have  $q_0/\pi \in L^1(\pi)$ , so the Birkhoff ergodic theorem [63; 62, p. 212] (see Theorem D.2) ensures:

$$\frac{1}{T} \sum_{t=0}^{T-1} \frac{q_0}{\pi}(f^{-t}(x)) \rightarrow 1, \quad \pi - \text{a.e. } x \in \mathcal{X}, \quad \text{as } T \rightarrow \infty. \quad (8)$$

868 The total variation convergence is then by the direct application of the *Scheffé's lemma* Lemma D.3. Notice that

$$\text{TV}(\hat{q}_T, \pi) = \int \left| \frac{\hat{q}_T}{\pi}(x) - 1 \right| \pi(dx) = \int \left| \frac{1}{T} \sum_{t=1}^T \frac{q_0}{\pi}(f_{\theta}^{-t}(x)) - 1 \right| \pi(dx).$$

869 To apply Lemma D.3, we set  $\phi_t(x) := \frac{1}{T} \sum_{t=1}^T \frac{q_0}{\pi}(f_{\theta}^{-t}(x))$ , and set  $\phi(x) := 1$ . Because  $q_0 \ll \pi$ , all  $\phi_n$ 's  
870 are  $\pi$ -integrable. Then, for all  $n \in \mathbb{N}$ , we obtain that

$$\begin{aligned} \int |\phi_n(x)| \pi(dx) &= \int \phi_n(x) \pi(dx) \\ &= \frac{1}{T} \sum_{t=1}^T \int \frac{q_0}{\pi}(f_{\theta}^{-t} x) \pi(dx) \\ &= \int q_0(dx) \quad (\text{as } f_{\theta} \pi = \pi) \\ &= 1 = \int |\phi(x)| \pi(dx), \end{aligned}$$

871 yielding the second convergence in Lemma D.3. □

### 872 D.2 Convergence of the IRF MixFlow

873 As hinted in the main text, the proof of Theorem 3.1 involves interpreting the IRF as a time-homogeneous,  
874 e.m.p. dynamical system on the joint space  $\Theta^{\mathbb{N}} \times \mathcal{X}$ . Specifically, we define a map  $\Phi$  (Eq. (9)) whose iterates  
875 evolve both the state  $X_t$  and the parameter sequence  $(\theta_t)_{t \in \mathbb{N}}$ . Overall, the proof proceeds in two steps. First,  
876 we show that the joint law of  $(\theta_t, X_t)$  converges in total variation to  $\mathbb{P} \otimes \pi$ . Second, we deduce marginal  
877 convergence for  $X_t$ . Section D.2.1 establishes the joint result, while Section D.2.2 explains why it suffices to  
878 prove Theorem 3.1.

### 879 D.2.1 Convergence in the product space

880 The key technique for proving the joint convergence is to interpret the iterative process Eq. (2) as an autonomous,  
 881 ergodic, and measure-preserving dynamical system in the joint space  $\Theta^{\mathbb{N}} \times \mathcal{X}$ . Given this framework, the joint  
 882 convergence follows immediately, as substantiated by Xu et al. [26, Theorem 4.2] (which is based on the *mean*  
 883 *ergodic theorem*).

884 For brevity, we define  $\Omega = \Theta^{\mathbb{N}}$ ,  $\mathcal{F}_{\mathbb{N}} = \mathcal{F}^{\otimes \mathbb{N}}$ , and  $\mathbb{P}$  be the joint distribution of  $(\theta_t)_{t \in \mathbb{N}}$  with independent  
 885 marginal distribution  $\mu$ . Define the *shift operator*  $\sigma : \Omega \rightarrow \Omega$  by

$$\sigma\omega : (\omega_0, \omega_1, \dots) \mapsto (\omega_1, \omega_2, \dots).$$

886 And let  $(\theta_n)_{n \in \mathbb{N}}$  be the coordinate process on  $(\Omega, \mathcal{F}_{\mathbb{N}}, \mathbb{P})$ , i.e., for all  $\omega = (\omega_0, \omega_1, \dots) \in \Omega$ ,

$$\theta_n(\omega) = \omega_n.$$

887 By definition, we have  $\theta_{n+1} = \theta_n \circ \sigma$ , and  $(f_{\theta_n})_{n \in \mathbb{N}}$  with  $(\theta_n)_{n \in \mathbb{N}} \stackrel{\text{iid}}{\sim} \mu$  can be formally understood as  
 888  $(f_{\theta_n(\omega)})_{n \in \mathbb{N}}, \omega \sim \mathbb{P}$  satisfying that  $f_{\theta_n(\omega)} = f_{\theta_0 \circ \sigma^n(\omega)} = f_{\theta_0(\sigma^n \omega)}$ . For the rest of this work, we abuse the  
 889 notation by writing  $f_{\theta_n(\omega)}$  as  $f_{\sigma^n \omega}$  for all  $n \in \mathbb{N}$ .

890 Now consider the product probability space  $(\Omega \times \mathcal{X}, \mathcal{F}_{\mathbb{N}} \otimes \mathcal{B}, \mathbb{P} \times \pi)$ , where  $\mathbb{P} \times \pi$  denotes the joint dis-  
 891 tribution with independent marginals  $\mathbb{P}$  and  $\pi$  on  $\Omega$  and  $\mathcal{X}$  respectively. We define the transformation  
 892  $\Phi : \Omega \times \mathcal{X} \rightarrow \Omega \times \mathcal{X}$  by

$$\Phi(\omega, x) = (\sigma\omega, f_{\sigma\omega}(x)), \quad \forall (\omega, x) \in \Omega \times \mathcal{X}. \quad (9)$$

893 Note that Eq. (9) equivalently describes the iterative process Eq. (2) with i.i.d.  $(\theta_n)_{n \in \mathbb{N}}$ . For the rest of the  
 894 proof, we will focus on the autonomous dynamical system  $(\Omega \times \mathcal{X}, \mathcal{F}_{\mathbb{N}} \otimes \mathcal{B}, \mathbb{P} \times \pi, \Phi)$ .

895 **Theorem D.4.** *Under the same assumption of Theorem 3.1, we have*

$$\text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} \Phi^n(\mathbb{P} \times q_0), \mathbb{P} \times \pi \right) \rightarrow 0, \quad \text{as } N \rightarrow \infty. \quad (10)$$

896 *Proof of Theorem D.4.* We first show that  $\Phi$  preserves  $\mathbb{P} \times \pi$ , namely,  $\Phi(\mathbb{P} \times \pi) = \mathbb{P} \times \pi$ . For all  $\xi \in L^1(\mathbb{P} \times \pi)$ ,

$$\begin{aligned} \Phi(\mathbb{P} \times \pi)(\xi) &:= \int_{\Omega \times \mathcal{X}} \xi(\omega, x) \Phi(\mathbb{P} \times \pi)(d\omega, dx) \\ &= \int_{\Omega \times \mathcal{X}} \xi \circ \Phi(\omega, x) \mathbb{P} \times \pi(d\omega, dx) \\ &= \int_{\Omega} \int_{\mathcal{X}} \xi(\sigma\omega, f_{\sigma\omega}(x)) \pi(dx) \mathbb{P}(d\omega) \end{aligned} \quad (11)$$

897 Since  $\sigma$  is measure-preserving for  $\mathbb{P}$  due to the i.i.d. assumption, and  $x \mapsto f_{\omega}(x)$  is  $\pi$ -measure-preserving by  
 898 hypothesis, we obtain that

$$\begin{aligned} \Phi(\mathbb{P} \times \pi)(\xi) &= \int_{\Omega} \int_{\mathcal{X}} \xi(\omega, f_{\omega}(x)) \pi(dx) \mathbb{P}(d\omega) \\ &= \int_{\Omega} \int_{\mathcal{X}} \xi(\omega, x) (f_{\omega} \pi)(dx) \mathbb{P}(d\omega) \\ &= \int_{\Omega} \int_{\mathcal{X}} \xi(\omega, x) \pi(dx) \mathbb{P}(d\omega) \\ &= \int_{\Omega \times \mathcal{X}} \xi(\omega, x) \mathbb{P} \times \pi(d\omega, dx) \\ &=: (\mathbb{P} \times \pi)(\xi). \end{aligned}$$

899 This concludes that  $(\Omega \times \mathcal{X}, \mathcal{F}_{\mathbb{N}} \otimes \mathcal{B}, \mathbb{P} \times \pi, \Phi)$  is a measure-preserving dynamical system.

900 We further show that  $(\Omega \times \mathcal{X}, \mathcal{F}_{\mathbb{N}} \otimes \mathcal{B}, \mathbb{P} \times \pi, \Phi)$  is an ergodic dynamical system. Morita [33, Theorem 4.1]  
 901 shows that it is equivalent to show the ergodicity of the shift dynamical system— $(\mathcal{X}^{\mathbb{N}}, \mathcal{B}^{\otimes \mathbb{N}}, \mathbb{P}_{\pi}, \tau)$ —induced  
 902 by the Markov process associated to Eq. (2). Here  $\mathbb{P}_{\pi}$  is the unique probability measure on  $(\mathcal{X}^{\mathbb{N}}, \mathcal{B}^{\otimes \mathbb{N}})$  so  
 903 that the coordinate process  $(X_1, X_2, \dots)$  is a Markov chain with kernel  $P$  (Eq. (3)) and initial distribution  
 904  $\pi$ , and  $\tau$  is the shift operator on  $\mathcal{X}^{\mathbb{N}}$ , i.e.,  $\tau(X_0, X_1, \dots) = (X_1, X_2, \dots)$ . Douc et al. [45, Theorem 5.2.6]  
 905 further guarantees that if  $\pi$  is the unique invariant probability measure of  $P$ , then  $(\mathcal{X}^{\mathbb{N}}, \mathcal{B}^{\otimes \mathbb{N}}, \mathbb{P}_{\pi}, \tau)$  is both  
 906 measure-preserving and ergodic. Hence, the second assertion of Assumption 2.1 guarantees the ergodicity of  
 907  $(\Omega \times \mathcal{X}, \mathcal{F}_{\mathbb{N}} \otimes \mathcal{B}, \mathbb{P} \times \pi, \Phi)$ .

908 Finally, we apply Theorem 4.2 in Xu et al. [26] to finish the proof. Given that  $\Phi$  is measure-preserving and  
 909 ergodic for  $\mathbb{P} \times \pi$ , it remains to show that  $q \ll \pi$  implies that  $\mathbb{P} \times q \ll \mathbb{P} \times \pi$ . For all  $B \in \mathcal{B}$  and  $F \in \mathcal{F}_{\mathbb{N}}$ ,

$$0 = (\mathbb{P} \times \pi)(F, B) = \mathbb{P}(F) \times \pi(B) \implies \mathbb{P}(F) = 0 \text{ or } \pi(B) = 0.$$

910 Since  $(\mathbb{P} \times q)(F, B) = \mathbb{P}(F) \times q(B)$ , if  $\mathbb{P}(F) = 0$ , then  $\mathbb{P}(F) \times q(B) = 0$ , and if  $\pi(B) = 0$ , then  $q(B) = 0$   
 911 by hypothesis and  $\mathbb{P}(F) \times q_0(B) = 0$  as well. Therefore, Xu et al. [26, Theorem 4.2] yields the desired  
 912 result.  $\square$

## 913 D.2.2 From the joint convergence to Theorem 3.1

914 Finally, we justify why Eq. (10) is sufficient for Eq. (4).

915 *Proof of Theorem 3.1.* We first derive the explicit expression of  $\Phi(\mathbb{P} \times q_0)$  and examine its conditional proba-  
 916 bility measure. Following the same derivation as Eq. (11), for all  $\xi \in L^1(\mathbb{P} \times q_0)$ ,

$$\begin{aligned} \Phi(\mathbb{P} \times q_0)(\xi) &= \int_{\Omega} \int_{\mathcal{X}} \xi(\sigma\omega, f_{\sigma\omega}(x)) q_0(dx) \mathbb{P}(d\omega) \\ &= \int_{\Omega} \int_{\mathcal{X}} \xi(\omega, f_{\omega}(x)) q_0(dx) \mathbb{P}(d\omega) \\ &= \int_{\Omega} \int_{\mathcal{X}} \xi(\omega, x) (f_{\omega} q_0)(dx) \mathbb{P}(d\omega), \end{aligned} \quad (12)$$

917 where the second equality is by the fact that  $\sigma$  is measure-preserving for  $\mathbb{P}$ . Eq. (12) demonstrates that  $\Phi(\mathbb{P} \times q_0)$   
 918 can be disintegrated into the marginal distribution  $\mathbb{P}(d\omega)$  on  $\Omega$  and the conditional distribution  $(f_{\omega} q_0)(dx)$ ,  
 919 yielding that

$$X_n | (\theta_i)_{i \in \mathbb{N}} \sim f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0.$$

920 Hence, disintegration of  $\frac{1}{N} \sum_{n=0}^{N-1} \Phi^n(\mathbb{P} \times q_0)$  on the slice  $(\theta_0, \theta_1, \dots) \in \Omega$  is

$$\frac{1}{N} \sum_{n=0}^{N-1} f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0.$$

921 Then we show that the total variation convergence of the joint distribution (Theorem D.4) implies the total  
 922 variation convergence of the conditionals (Theorem 3.1). For all  $N \in \mathbb{N}$ ,

$$\text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} \Phi^n(\mathbb{P} \times q_0), \mathbb{P} \times \pi \right) = \int_{\Omega} \int_{\mathcal{X}} \left| \frac{1}{N} \sum_{n=0}^{N-1} \frac{d\Phi^n(\mathbb{P} \times q_0)}{d(\mathbb{P} \times \pi)} - 1 \right| \pi(dx) \mathbb{P}(d\theta)$$

923 Notice that for all  $n \in \mathbb{N}$ , the Radon-Nikodym derivative  $\frac{d\Phi^n(\mathbb{P} \times q_0)}{d(\mathbb{P} \times \pi)}$  always exists given that  $\mathbb{P} \times q_0 \ll \mathbb{P} \times \pi$   
 924 and  $\Phi$  is  $\mathbb{P} \times \pi$ -measure-preserving. And explicitly, since  $\mathbb{P} \times q_0$  and  $\mathbb{P} \times \pi$  have same marginal distributions  
 925 on  $\Omega$ , we have

$$\frac{d\Phi^n(\mathbb{P} \times q_0)}{d(\mathbb{P} \times \pi)} = \frac{f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0}{\pi}.$$

926 Hence,

$$\begin{aligned} \text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} \Phi^n(\mathbb{P} \times q_0), \mathbb{P} \times \pi \right) &= \int_{\Omega} \int_{\mathcal{X}} \left| \frac{1}{N} \sum_{n=0}^{N-1} \frac{f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0(x)}{\pi(x)} - 1 \right| \pi(dx) \mathbb{P}(d\theta) \\ &= \mathbb{E} \left[ \text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0, \pi \right) \right], \quad (\theta_n)_{n \in \mathbb{N}} \sim \mathbb{P} \end{aligned}$$

927 Since  $\text{TV}(\cdot, \cdot)$  is always non-negative, the left-hand side converges to 0 as  $N \rightarrow \infty$  yields that the following  
 928 convergence holds in probability  $\mathbb{P}$ :

$$\text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0, \pi \right) \rightarrow 0, \quad \text{as } N \rightarrow \infty. \quad (13)$$

929 One may notice a subtle difference on the indices of  $\theta$ 's between Eq. (13) and Eq. (10). Due to the i.i.d.  
 930 assumption on  $(\theta_n)_{n \in \mathbb{N}}$ ,

$$\text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} f_{\theta_{n-1}} \circ \cdots \circ f_{\theta_0} q_0, \pi \right) \stackrel{d}{=} \text{TV} \left( \frac{1}{N} \sum_{n=0}^{N-1} f_{\theta_n} \circ \cdots \circ f_{\theta_1} q_0, \pi \right),$$

931 which completes the proof.  $\square$

### 932 D.3 Convergence of the backward IRF MixFlow

933 *Proof of Theorem 3.2.* The pointwise density convergence is the direct consequence of Eq. (5) via Theorem 2.2.  
 934 The total variation convergence is then established using identical strategy as the proof of Theorem C.1 via  
 935 Scheffé’s lemma Lemma D.3.  $\square$

### 936 D.4 Convergence of the ensemble IRF MixFlow

937 *Proof of Theorem 3.3.* By the definition of the total variation,

$$\begin{aligned} \text{TV} \left( \tilde{q}_T^{(M)}, \pi \right) &= \int \left| \frac{\tilde{q}_T^{(M)}}{\pi}(x) - 1 \right| \pi(\mathrm{d}x) \\ &= \int \left| \frac{1}{M} \sum_{m=1}^M \frac{q_0}{\pi} \left( f_{\theta_1^{(m)}}^{-1} \circ \dots \circ f_{\theta_T^{(m)}}^{-1}(x) \right) - 1 \right| \pi(\mathrm{d}x). \end{aligned}$$

938 By the triangle inequality,

$$\begin{aligned} &\leq \int \left| \frac{1}{M} \sum_{m=1}^M \frac{q_0}{\pi} \left( f_{\theta_1^{(m)}}^{-1} \circ \dots \circ f_{\theta_T^{(m)}}^{-1}(x) \right) - R^T \left( \frac{q_0}{\pi} \right) (x) \right| \pi(\mathrm{d}x) + \int \left| \int \frac{q_0}{\pi}(y) R^T \delta_x(\mathrm{d}y) - 1 \right| \pi(\mathrm{d}x). \end{aligned} \quad (14)$$

939 We derive upper bounds for two terms on the right-hand side separately.

940 For the first term, taking the expectation with respect to the randomness of  $\theta \sim \mu$ , and interchange the order of  
 941 integrations,

$$\begin{aligned} &\mathbb{E} \left[ \int \left| \frac{1}{M} \sum_{m=1}^M \frac{q_0}{\pi} \left( f_{\theta_1^{(m)}}^{-1} \circ \dots \circ f_{\theta_T^{(m)}}^{-1}(x) \right) - R^T \left( \frac{q_0}{\pi} \right) (x) \right| \pi(\mathrm{d}x) \right] \\ &= \mathbb{E} \left[ \int \left| \frac{1}{M} \sum_{m=1}^M \frac{q_0}{\pi} \left( f_{\theta_1^{(m)}}^{-1} \circ \dots \circ f_{\theta_T^{(m)}}^{-1}(X) \right) - R^T \left( \frac{q_0}{\pi} \right) (X) \right| \mu \left( \mathrm{d}\theta_{1:T}^{(1:M)} \right) \right], \quad X \sim \pi \end{aligned}$$

942 Notice that  $\forall x \in \mathcal{X}$ ,  $\left\{ f_{\theta_1^{(m)}}^{-1} \circ \dots \circ f_{\theta_T^{(m)}}^{-1}(x) \right\}_{m=1}^M \stackrel{\text{iid}}{\sim} R^T \delta_x$ , where the randomness comes from the inde-  
 943 pendent realization of  $\theta$ s, where  $R$  is the induced the Markov process of  $f_\theta^{-1}$ . Therefore, applying Jensen’s  
 944 inequality yields

$$\leq \frac{1}{\sqrt{M}} \mathbb{E} \left[ \sqrt{\text{Var}_{\theta_{1:T}} \left[ \frac{q_0}{\pi} \left( f_{\theta_1}^{-1} \circ \dots \circ f_{\theta_T}^{-1}(X) \right) \mid X \right]} \right],$$

945 For the second term of Eq. (14), since  $\frac{q_0}{\pi}$  is globally bounded by constant  $B < \infty$ , we have that

$$\begin{aligned} &\int \left| \int \frac{q_0}{\pi}(y) R^T \delta_x(\mathrm{d}y) - 1 \right| \pi(\mathrm{d}x) \\ &= \int \left| \int \frac{q_0}{\pi}(y) R^T \delta_x(\mathrm{d}y) - \int \frac{q_0}{\pi}(y) \pi(\mathrm{d}y) \right| \pi(\mathrm{d}x) \\ &\leq B \int \text{TV}(R^T \delta_x, \pi) \pi(\mathrm{d}x) \\ &= B \cdot \mathbb{E} \left[ \text{TV}(R^T \delta_X, \pi) \right], \quad X \sim \pi. \end{aligned}$$

946 This completes the proof.

947  $\square$

### 948 D.5 Proof of Proposition 4.1

949 *Proof of Proposition 4.1.* We first verify that the map defined in Algorithm 2 is  $\bar{\pi}$ -measure-preserving, invoking  
 950 the second part of Proposition 4.1. The algorithm has four steps (see Section 4); we compute the Jacobian of  
 951 each step. Steps 3-4 involve a discrete accept/reject decision, so we treat the two branches separately—within  
 952 a branch the transformation is a diffeomorphism, making the Jacobian well defined.

953 1. Step 1 describes constant shifts applied to uniform random variables, which preserves  $\text{Unif}_{[0,1]}(\mathrm{d}u_v)$   
 954 and  $\text{Unif}_{[0,1]}(\mathrm{d}u_a)$  with Jacobian 1.



955 2. Step 2 is the CDF/inverse-CDF transformation of  $\rho(\cdot|x)$ . As long as the CDF  $F(\cdot|x)$  is well-defined,  
 956 this step describes a diffeomorphism in  $\mathcal{V} \times [0, 1]$ . The corresponding Jacobian is given by:

$$\frac{\rho(\tilde{v}|x)}{\rho(v|x)}$$

957 3. We analyze step 3 and 4 together. In the rejection branch, no additional transformation is applied, so  
 958 the Jacobian is 1. In the acceptance branch, step 3 involves the involution mapping, with Jacobian  
 959  $|\frac{\partial g(x, \tilde{v})}{\partial x, \tilde{v}}|^{-1}$ , and step 4 rescale  $u_a$  by the MH-ratio  $r$ , yielding a combined Jacobian with step 3  
 960  $\frac{\pi(x', v')}{\pi(x, \tilde{v})}$ .

961 Hence, in the rejection branch, the combined jacobian of step 1-4 evaluated on  $s' = (x, \tilde{v}, u'_v, u_a)$  is

$$\frac{\rho(\tilde{v}|x)}{\rho(v|x)} = \frac{\pi(x, \tilde{v}, u'_v, u_a)}{\pi(x, v, u_v, u_a)}.$$

962 In the acceptance branch, the combined jacobian of step 1-4 evaluated on  $s' = (x', v', u'_v, u'_a)$  is

$$\frac{\rho(\tilde{v}|x)}{\rho(v|x)} \frac{\pi(x', v')}{\pi(x, \tilde{v})} = \frac{\pi(x', v', u'_v, u'_a)}{\pi(x, v, u_v, u_a)}.$$

963 Both satisfy the criterion of Proposition 4.1; the map is therefore  $\pi$ -measure-preserving.

964 Finally, we show uniqueness of the invariant distribution. By Douc et al. [45, Corollary 9.2.16], an irreducible  
 965 kernel has at most one invariant distribution. Because each  $f_\theta$  preserves  $\pi$ , the induced Markov kernel  $P$  must  
 966 admit  $\pi$  as an invariant distribution. If  $P$  is irreducible, then  $\pi$  is its unique invariant distribution.

967 □

## E Additional experimental details

For all deterministic MixFlows variants, the uniform-shift parameters were fixed to  $\theta_v = \pi/8$  and  $\theta_a = \pi/7$ . For NUTS benchmarks, we the Julia package `AdvancedHMC.jl` [64] with default settings throughout. The normalizing flow architectures were implemented as follows. In `RealNVP`, the affine coupling layers consist of two separate multilayer perceptrons (MLPs)—one for scaling and one for shifting—each with three fully connected layers and LeakyReLU activations. For `Neural Spline Flows` (NSF), we set the spline bandwidth to  $B = 30$ , and used  $K = 11$  knots. For synthetic examples, the hidden dimension in each MLP was set to 32 for `RealNVP` and 64 for NSF. For real-data examples, the hidden dimension was set to  $\min(d, 64)$ , where  $d$  is the dimensionality of the target posterior distribution.

We conducted experiments on two computing platforms: a local machine equipped with an AMD Ryzen 9 5900X CPU and 64 GB of RAM, and a high-performance computing cluster (details to be provided upon acceptance). All code used in the experiments is included in the submitted `.zip` file.

### E.1 Synthetic experiments

The four target distributions used in this experiment are as follows:

1. the banana distribution [55]:

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} 100 & 0 \\ 0 & 1 \end{bmatrix} \right), \quad x = \begin{bmatrix} y_1 \\ y_2 + by_1^2 - 100b \end{bmatrix}, \quad b = 0.1;$$

2. Neals’ funnel [56]:

$$x_1 \sim \mathcal{N}(0, \sigma^2), \quad x_2 | x_1 \sim \mathcal{N} \left( 0, \exp \left( \frac{x_1}{2} \right) \right), \quad \sigma^2 = 36;$$

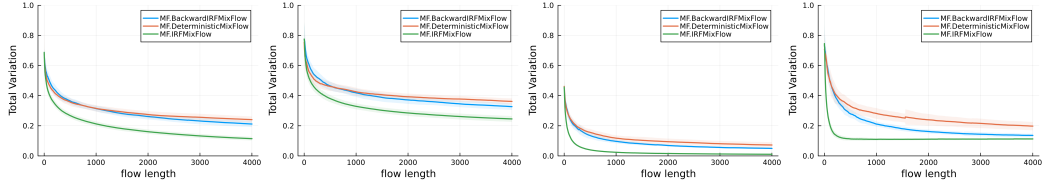
3. a cross-shaped distribution: in particular, a Gaussian mixture of the form

$$x \sim \frac{1}{4} \mathcal{N} \left( \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 0.15^2 & 0 \\ 0 & 1 \end{bmatrix} \right) + \frac{1}{4} \mathcal{N} \left( \begin{bmatrix} -2 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0.15^2 \end{bmatrix} \right) \\ + \frac{1}{4} \mathcal{N} \left( \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 0.15^2 \end{bmatrix} \right) + \frac{1}{4} \mathcal{N} \left( \begin{bmatrix} 0 \\ -2 \end{bmatrix}, \begin{bmatrix} 0.15^2 & 0 \\ 0 & 1 \end{bmatrix} \right);$$

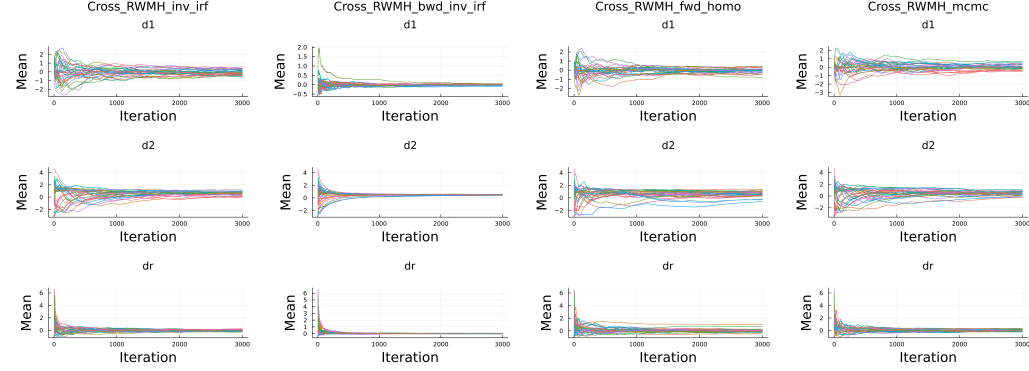
4. and a warped Gaussian distribution

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \sim \mathcal{N} \left( 0, \begin{bmatrix} 1 & 0 \\ 0 & 0.12^2 \end{bmatrix} \right), \quad x = \begin{bmatrix} \|y\|_2 \cos \left( \text{atan2}(y_2, y_1) - \frac{1}{2} \|y\|_2 \right) \\ \|y\|_2 \sin \left( \text{atan2}(y_2, y_1) - \frac{1}{2} \|y\|_2 \right) \end{bmatrix},$$

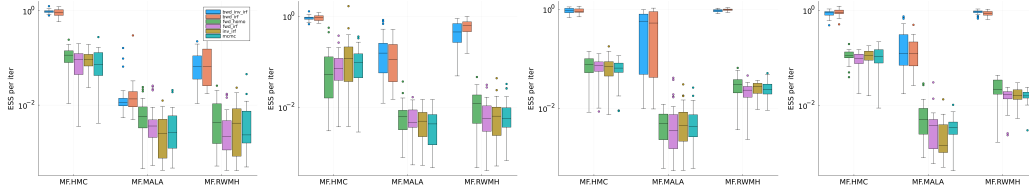
where  $\text{atan2}(y, x)$  is the angle, in radians, between the positive  $x$  axis and the ray to the point  $(x, y)$ .



Total-variation error for deterministic, IRF, and backward IRF MixFlows built on RWMH kernels, plotted against flow length  $T$  for the most performant step sizes among  $\{0.05, 0.2, 1.0\}$ . Each curve is the mean over 32 independent runs; shaded bands ( $\pm 1$  SD) show run-to-run variability.



Running mean estimates over 3000 iterates from different IRF and MCMC dynamics based on RWMH, evaluated on the Cross distribution across 32 independent runs. Each line represents the trajectory of a single run. From top to bottom, the rows show the running mean of the test functions  $(x_1, x_2) \mapsto x_1$ ,  $(x_1, x_2) \mapsto x_2$ , and  $(x_1, x_2) \mapsto \frac{q_0}{\pi}(x_1, x_2)$ . From left to right, the columns correspond to the dynamic of inverse IRF  $f_{\theta}^{-1}$ , the backward process of the inverse IRF, time-homogeneous dynamics  $f_{\theta^*}$ , and the standard RWMH MCMC.



Per-sample MCMC effective sample size (ESS) estimates on the test function  $\frac{q_0}{\pi}$ , computed from trajectories generated by various IRF and MCMC dynamics based on HMC, MALA, and RWMH kernels. The trajectory lengths are set to 300 for HMC-based dynamics, 2000 for MALA, and 4000 for RWMH. Each ESS value is computed from a single trajectory, and the boxplots summarize the ESS estimates over 32 independent runs per method. The per-sample ESS for i.i.d. samples will be 1.

Figure 5: Results showing difference between deterministic, IRF, and backward IRF MixFlows

Fig. 5a compares the total variation (TV) errors of Deterministic, IRF, and Backward-IRF MixFlows constructed from RWMH kernels. Overall, Deterministic and Backward-IRF MixFlows perform similarly, though the latter exhibits slightly improved accuracy at longer flow lengths. IRF MixFlow consistently outperforms both, achieving faster TV convergence and lower variability across runs. As discussed in Section 3.4, this improvement stems from differences in the convergence behavior of the series  $\frac{1}{K} \sum_{k=1}^K \frac{q_0}{\pi}(T_k(x))$ , where  $T_k$  represents the sequence of transformations used in the density computation of each MixFlow variant.

Fig. 5b further illustrates this effect by showing running mean estimates over 3000 iterations for the Cross distribution. From top to bottom, each row shows the mean of the test functions  $(x_1, x_2) \mapsto x_1$ ,  $(x_1, x_2) \mapsto x_2$ , and  $(x_1, x_2) \mapsto \frac{q_0}{\pi}(x_1, x_2)$ . From left to right, the columns correspond to the inverse IRF  $f_{\theta}^{-1}$  (Backward-IRF MixFlow), the backward process of the inverse IRF (IRF MixFlow), the time-homogeneous flow  $f_{\theta^*}$  (deterministic MixFlow), and standard RWMH MCMC. The backward process exhibits significantly faster convergence in all cases, consistent with the superior TV performance of IRF MixFlows under equal flow lengths. This advantage arises from reduced autocorrelation in the backward iterates.

Fig. 5c reports the per-sample MCMC effective sample size (ESS) for the test function  $\frac{q_0}{\pi}$ , estimated from trajectories generated using various IRF and MCMC dynamics based on HMC, MALA, and RWMH. This metric captures the degree of autocorrelation in  $\frac{q_0}{\pi}(T_k(x))$  across iterations. Backward process dynamics consistently yield ESS values orders of magnitude higher than other methods—often approaching the ideal of independent sampling, with relative ESS close to 1 in some cases.

### E.1.2 Ensemble IRF MixFlows: scaling up $M$ or $T$

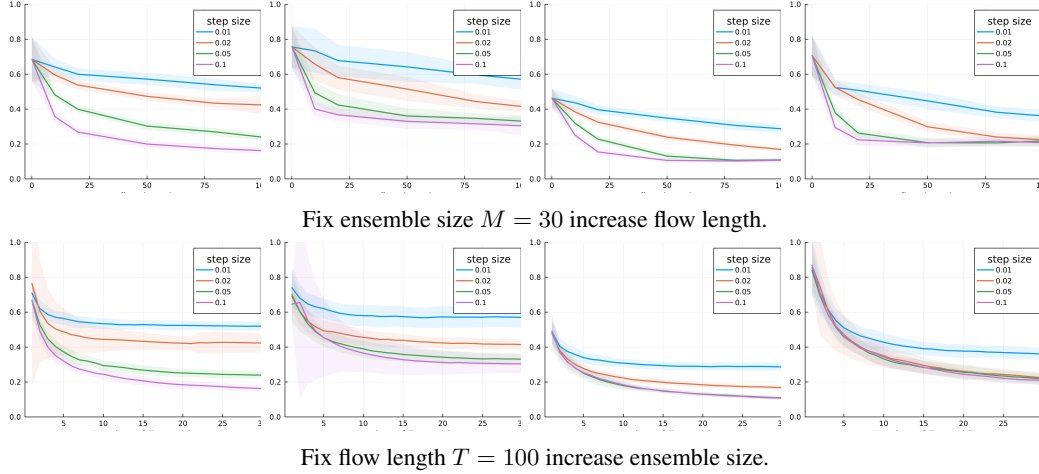


Figure 6: TV error of ensemble IRF MixFlows based on HMC over increasing ensemble size  $M$  and flow length  $T$ . Each curve is the mean over 32 independent runs; shaded bands ( $\pm 1$  SD) show run-to-run variability.

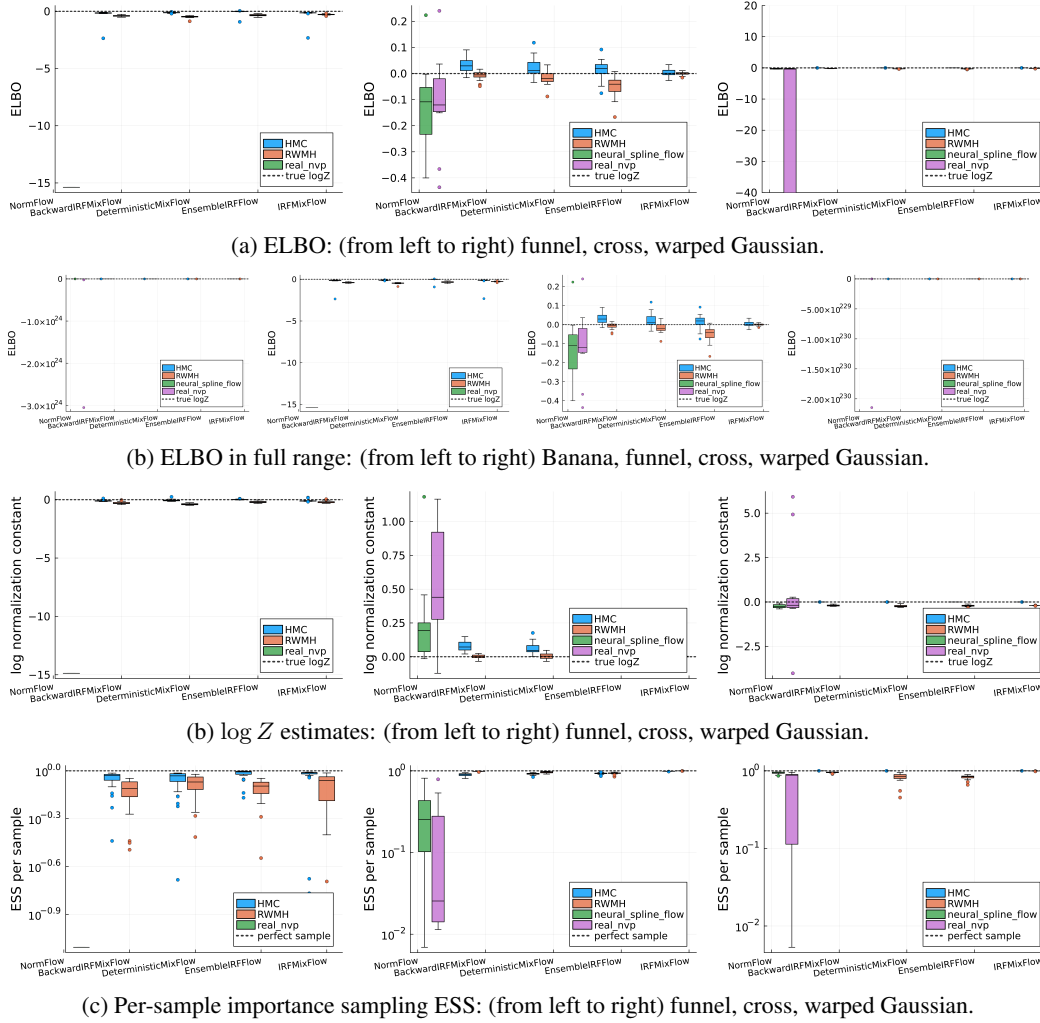


Figure 7: Variational approximation quality of IRF Flows versus Rea1NVP and NSF. Box plots for IRF flows are based on 32 independent runs, and 10 runs for the normalizing flows.

## 1008 E.2 Additional results for real-data experiments

1009 To approximate the ground truth, we ran an AIS procedure with 4096 particles with adaptive schedule selection.  
 1010 The initial temperature schedule was generated via mirror descent [65] with a small step size of 0.005; the  
 1011 schedule was then refined for five rounds using the adaptive scheme of Syed et al. [66], yielding more than 1000  
 1012 annealing steps for each data set. All reference values are taken as the median estimates across 10 independent  
 1013 runs of the above procedure.

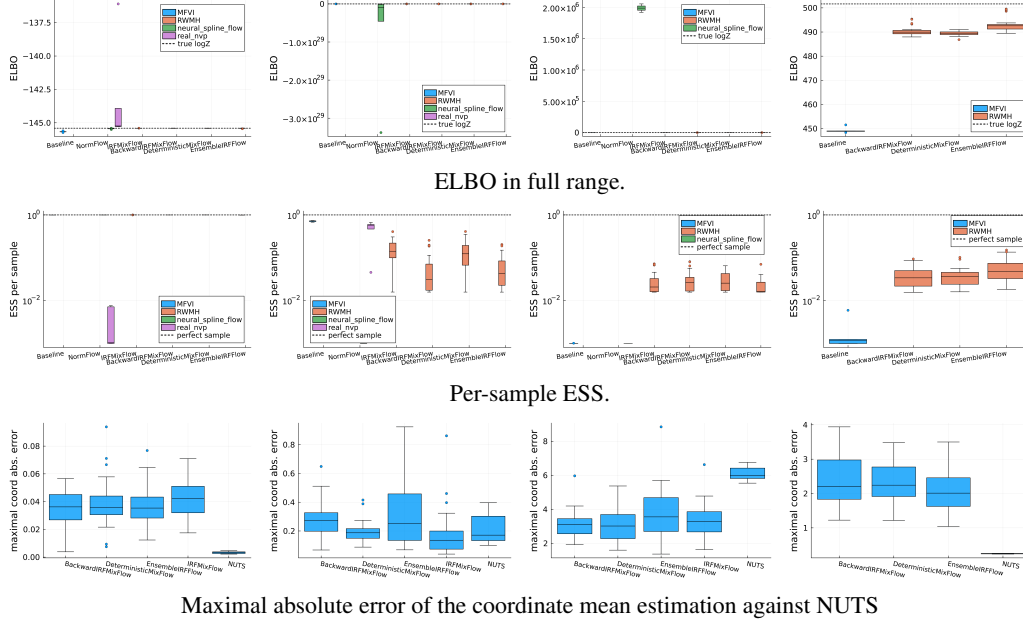


Figure 8: Results on real-data benchmarks (columns, from left to right): TReg( $d = 4$ ), Brownian( $d = 32$ ), SparseReg( $d = 83$ ), LGCP( $d = 1600$ )