Two-Phase Rapid Simulation-Based Inference with Differentiable Simulators

Uros Zivanovic[1, 2, 3], **Rahul Srinivasan**[3, 4], **Roberto Trotta**[3, 5, 6, 7], and **Andre Scaffidi**[3]

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]Max Planck Institute for Gravitational Physics (Albert Einstein Institute), Potsdam, Germany
[3]Scuola Internazionale Superiore di Studi Avanzati (SISSA), Trieste, Italy
[4]IFPU - Institute for Fundamental Physics of the Universe, Trieste, Italy
[5]INFN – National Institute for Nuclear Physics, Italy
[6]Italian Research Center on High-Performance Computing, Big Data and Quantum Computing, Casalecchio di Reno, Italy
[7]Imperial College London, United Kingdom

Abstract

We introduce Rapid Simulation Based Inference (RSBI), a diffusion-based variational approach to likelihood-free Bayesian inference that achieves high sampling efficiency under large prior-to-posterior volumes. RSBI builds on advances in Schrödinger Bridge (SB) diffusion sampling to handle multi-modal posteriors, with an initial likelihood supplied by a distance measure via a differentiable simulator. The key innovation of this initial surrogate likelihood is that it gives a highly efficient proposal distribution, replacing sequential rounds of other methods with an efficient one-shot proposal. Subsequent refinement then transports the proposal to the true posterior as specified by the simulator and the original prior. The SB sampler is warm-started from the proposal, and its energy is corrected by a neural likelihood ratio that is learned only where the proposal places mass. As a variational method, RSBI does not suffer from prior leakage commonly observed in standard sequential posterior estimation methods. To improve mode coverage we optionally utilize the well-tempered meta-dynamics framework, which also encourages exploration of the prior volume. We observe strong performance on standard SBI benchmarks, particularly when the prior volume is increased relative to the ground truth posterior, achieving a performance degradation of only $\sim 5\%$ on Two Moons under a highly constrained simulation budget. Additionally, we evaluate RSBI's performance for gravitational wave ring-down posterior estimation, in a real-world physics benchmark inspired by black hole spectroscopy.

1 Introduction

Many domains in science and engineering rely on stochastic simulators whose sampling distribution $p(x \mid \theta)$ is intractable but from which data samples x can be drawn given parameters θ. Bayesian parameter inference in such settings is the subject of *Simulation-Based Inference* (SBI) [1]. Neural SBI methods have matured rapidly over the past five years. Sequential Neural Posterior Estimation [SNPE; 2, 3] targets the posterior, Sequential Neural Likelihood Estimation [SNLE; 4] the likelihood, and Sequential Neural Ratio Estimation [SNRE; 5, 6] the likelihood-to-evidence ratio. Sequential Neural Variational Inference [SNVI; 7] combines these with variational inference (VI).

A common difficulty arises when the prior-to-posterior volume ratio ($\mathcal{R}_p$) is large: when $p(\theta)$ is broad compared to the posterior, most simulator calls fall in regions of negligible posterior density, and efficiency vanishes. Sequential variants can reduce this cost by retraining the proposal across

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

rounds, but this comes at a price. SNPE with atomic proposals suffers from posterior *leakage* outside prior support [3], and naive implementations of variational approaches collapse onto a single mode of the posterior [7]. Truncation schemes [8–10] instead restrict the effective prior, so their result is exact only if the truncated region contains the posterior, or else must be corrected at inference time, for example by importance sampling [11].

A growing class of scientific simulators are *differentiable*: GPU-native models written within the JAX [12]/PyTorch [13] frameworks, learnable physical simulators, and neural emulators trained to mimic expensive codes all expose the gradient $\nabla_\theta\, s(\theta)$ of the parameters θ under the simulator output, $s(\theta)$. This represents an opportunity for SBI: algorithms can exploit gradients directly rather than treating the simulator as a black box.

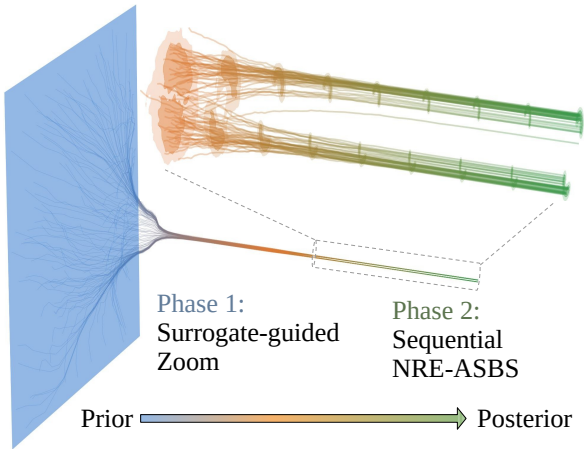

Figure 1: Actual diffusion trajectories from the RSBI pipeline on the Two Moons task detailed in Sec.5. RSBI is able to produce a high quality posterior despite the large prior size.

Recent advances in neural samplers, in particular the Adjoint Schrödinger Bridge Sampler (ASBS; 14), provide an opportunity to exploit such gradients. ASBS recasts sampling from an unnormalised target distribution as a Schrödinger bridge problem solved via stochastic optimal control: a controlled SDE transports a tractable source distribution to the target, with the control trained by adjoint matching and a learned corrector [15]. The recent Well-Tempered variant (WT-ASBS; 16) augments ASBS with a metadynamics-style bias [17], overcoming free-energy barriers between modes during training and enabling broad mode coverage at fixed simulation cost. Two properties make this family particularly well-suited to SBI. First, ASBS queries the target only through gradients of its energy (the negative log of the unnormalised target density), $\nabla_\theta E$, which a differentiable simulator supplies. Second, the well-tempered bias addresses the mode-seeking pathology that limits standard variational SBI [7] on multi-modal posteriors. To our knowledge, neither ASBS nor its well-tempered extension have previously been deployed for likelihood-free Bayesian inference.

We introduce **RSBI**, a two-phase SBI framework that leverages differentiable simulators while keeping the target posterior fixed as $p(\theta \mid x_o)$, where x_o denotes the observed data. Phase 1 places posterior mass rapidly using a cheap surrogate likelihood, parameterized by bandwidth ϵ and built from distance metric $\mathcal{D}_\epsilon(s, x_o)$ between observed and simulated data, plugged into a Well-Tempered Adjoint Schrödinger Bridge Sampler [14, 16] (WT-ASBS). Phase 2 refines the resulting Phase-1 proposal using WT-ASBS, augmented by an NRE trained on simulator samples drawn from the 'zoomed in' proposal. Figure. 1 outlines the process.

Our contributions are threefold:

- **A two-phase SBI method for differentiable simulators.** We introduce RSBI (Sec. 4), a diffusion-based variational approach to likelihood-free Bayesian inference that achieves high sampling efficiency under large prior-to-posterior volume ratios.

- **First application of WT-ASBS to SBI.** We adapt the well-tempered metadynamics extension of ASBS [16] to likelihood-free Bayesian inference, countering the mode-seeking pathology of reverse-KL variational SBI [7] on multimodal posteriors.

- **Benchmarking robustness at large prior-to-posterior volume ratio $\mathcal{R}_p$.** We benchmark RSBI and other methods on Two Moons [18], SLCP [18, 19], Lotka–Volterra [18], Bernoulli GLM [18], and a new differentiable gravitational-wave ringdown task ($\dim\theta = 6$, $\dim x = 820$), sweeping prior-to-posterior volume ratios up to $\log_{10}\mathcal{R}_p \approx 12$.

2 Related Work

RSBI sits at the intersection of three research threads. The first is *neural simulation-based inference*, in which a density or ratio estimator is trained on simulated (θ, x) pairs to target the posterior, likelihood, or likelihood-to-evidence ratio [2–6, 20]. Sequential variants iteratively refine their proposal to concentrate simulations in informative regions of parameter space. SNVI+fKL [7] pairs likelihood(-ratio) estimation with VI and is our closest methodological sibling. Like RSBI, SNVI combines a learned neural likelihood ratio estimated via Neural Ratio Estimation (NRE) with a variational sampler, and, like RSBI, it explicitly addresses the mode-seeking pathology of standard reverse-KL VI, in its case by training under the mass-covering forward KL with sampling–importance-resampling for proposal updates. RSBI instead handles mode coverage through the well-tempered metadynamics bias of WT-ASBS [16], which flattens free-energy barriers between modes. The two methods also differ in how the initial proposal is obtained: SNVI draws its first round of simulations from the original prior, whereas RSBI is initialised by a one-shot surrogate-guided Schrödinger bridge.

The second thread is the surrogate-likelihood lineage that connects classical ABC [21] to modern VI. Any differentiable distance metric $\mathcal{D}_\epsilon(s, x_o)$ induces a pseudo-likelihood whose gradients propagate through a simulator, an idea explored by ABC-VI [22], as well as score-based approaches [23, 24]. RSBI borrows from this for Phase 1, but rather than amortising over x or running an MCMC against the surrogate, we use it to construct the energy on which WT-ASBS is trained. This gives a non-mode-seeking proposal that Phase 2 then transports to the true posterior, with a likelihood specified by an NRE learned only where the proposal places mass. This use of a surrogate kernel as a warm-start for a downstream refinement, rather than as the inference target itself, can alleviate the bandwidth-versus-bias trade-off that limits standalone ABC-VI.

The third thread is the rapidly maturing ecosystem of GPU-native, auto-differentiable simulators across many scientific fields and neural emulators [25–28]. Direct access to simulation gradients $\nabla_\theta\, s(\theta)$ enables gradient-based inference for models that previously admitted only analytical-likelihood treatments: ASBS [14] queries its target exclusively through energy gradients, which the differentiable simulator supplies via a distance metric $\mathcal{D}_\epsilon(s, x_0)$, sidestepping the density-estimation step of standard SNPE/SNLE/SNRE. A growing number of works that exploit gradients have been used to improve neural SBI. Brehmer et al. [29] augment the training losses of ratio and likelihood estimators with the joint score and likelihood ratio, and relaxations such as the Gumbel-Softmax [30] extend such gradients to simulators with discrete randomness. To our knowledge, the only gradient-informed NPE method adds a score-matching term to the NPE loss [31], leaving the proposal, the simulation allocation and the sampler unchanged. In all of these, gradients regularise an estimator fit to prior-wide simulations. In RSBI they instead drive the Phase 1 sampler directly, through an auto-encoder-like comparison of simulations with the observation.

3 Background

3.1 Simulation Based Inference

We consider a stochastic simulator $s : \Theta \times \Omega \to \mathcal{X}$ with parameters $\theta \in \Theta \subseteq \mathbb{R}^d$ and outputs $x = s(\theta, \omega) \in \mathcal{X}$, where $\omega \in \Omega$ collects all internal randomness of the simulator (noise draws, random seeds) and is drawn from a distribution $p(\omega)$ that does not depend on θ. The simulator implicitly defines the conditional density $p(x \mid \theta)$ and the distribution of $s(\theta, \omega)$ under $\omega \sim p(\omega)$. The likelihood $\theta \mapsto p(x_o \mid \theta)$ is this density evaluated at the observed data x_o. It can be sampled by running the simulator but not evaluated. We call the simulator *differentiable* if $\theta \mapsto s(\theta, \omega)$ is differentiable for every ω, so that pathwise gradients $\nabla_\theta s(\theta, \omega)$ are available. We write $s(\theta)$ for $s(\theta, \omega)$ with $\omega \sim p(\omega)$, and $s(\theta, \omega_{1:K}) = \{s(\theta, \omega_k)\}_{k=1}^K$ with $\omega_k \overset{\text{iid}}{\sim} p(\omega)$ for K independent replicates at the same θ. Given a prior $p(\theta)$ and observation x_o, simulation-based inference (SBI) targets the posterior $p(\theta \mid x_o) \propto p(x_o \mid \theta)\, p(\theta)$ using only forward simulations [1].

Classical approximate Bayesian computation (ABC) accepts prior draws $\theta \sim p(\theta)$ whose simulations fall within a tolerance ϵ of the observation, $\mathcal{D}_\epsilon(s(\theta), x_o) < \epsilon$, usually after reducing the data to summary statistics [32]. It scales poorly with $\dim(x)$ and is sensitive to the choice of summaries and of ϵ. Neural SBI instead trains a density estimator q_ϕ on a corpus of simulations (θ_i, x_i) with $\theta_i \sim p(\theta)$ and $x_i \sim p(x \mid \theta_i)$, targeting the posterior (NPE) [2, 3], the likelihood (NLE) [4] or the

likelihood-to-evidence ratio (NRE) [5, 6, 20]. Amortised variants train once to learn $p(\theta)p(x \mid \theta)$ and apply to any x, whereas sequential variants concentrate simulations around x_o, giving up amortisation for simulation efficiency.

3.2 Diffusion Sampling

Neural diffusion samplers learn a Stochastic Differential Equation (SDE) which transports samples $X_0 \sim \mu(x)$ from source distribution $\mu(x)$; at time $t = 0$ to the Boltzmann distribution $v(x)$ at time $t = 1$ given by:

$$v(x) = \frac{e^{-E(x)/T}}{Z}, \qquad Z = \int_{\mathcal{X}} e^{-E(x)/T} dx < \infty \tag{1}$$

The distribution is characterized by an unnormalized energy $E : \mathcal{X} \subseteq \mathbb{R}^d \to \mathbb{R}$ and temperature T, with the normalizing constant Z being intractable. Diffusion samplers aim to provide faster convergence using fewer evaluations of the energy function than Markov Chain Monte Carlo (MCMC) [33] methods by learning better proposal distributions. Especially when $v(x)$ is defined through an energy function, diffusion samplers provide a scalable alternative to sampling methods such as score matching [34, 35] or normalizing flows [2, 4] whose naive implementations rely on samples from $v(x)$.

Adjoint Schrödinger Bridge Sampling: ASBS[14] is a state-of-the-art diffusion sampler building on Adjoint Sampling (AS)[15] that solves the Schrödinger Bridge (SB) problem between source distribution $\mu(x)$ and target $v(x)$ through stochastic optimal control. Within the context of RSBI, we consider the following drift-less SDE:

$$dX_t = \sigma_t u_t(X_t)dt + \sigma_t dW_t, \qquad X_0 \sim \mu(X_0), \tag{2}$$

where σ_t is the diffusion coefficient, u_t is a learned control network, and $(W_t)_t$ is a d-dimensional Brownian motion. The optimal control u_t^* is then given by:

$$u^* = \arg\min_u D_{\mathrm{KL}}(p^u || p^{\text{base}}) = \arg\min_u \mathbb{E}_{\mathbb{X}\sim p^u}\left[\int_0^1 \frac{1}{2}||u_t(X_t)||^2 dt\right], \tag{3a}$$

$$\text{s.t. } dX_t = \sigma_t u_t(X_t)dt + \sigma_t dW_t, \quad X_0 \sim \mu(X_0), \quad X_1 \sim v(X_1). \tag{3b}$$

Therefore, u^* minimizes the relative entropy between the path probability p^u characterized by (2) and the base path probability p^{base} given when $u = 0$. To solve (3), ASBS defines an additional corrector network h and iterates over two objectives; Adjoint Matching (AM):

$$u^{(k)} = \arg\min_u \mathbb{E}_{p^{\text{base}}_{t|0,1} p^{\bar{u}}_{0,1}}\left[\|u_t(X_t) + \sigma_t(\nabla_{X_1} E(X_1) + h^{(k-1)}(X_1))\|^2\right], \quad \bar{u} = \texttt{stopgrad}(u), \tag{4}$$

and Corrector Matching (CM):

$$h^{(k)} = \arg\min_h \mathbb{E}_{p^u_{0,1}}\left[\|h(X_1) - \nabla_{X_1} \log p^{\text{base}}(X_1|X_0)\|^2\right]. \tag{5}$$

Iterating through AM and CM stages constitutes the well-known iterative proportional fitting[36] algorithm.

Theorem 3.1 (Convergence of ASBS)**.** As $k \to \infty$ and both AM(4) and CM(5) stages reach their critical points, $u^{(k)} = u^*$, the SB solution of (3).

For a proof of Theorem 3.1 we refer to Liu et. al.[14]. Note that ASBS relies only on the energy gradient $\nabla_{X_1} E(X_1)$ during the AM stage, and does not require any energy evaluations during the CM stage. This makes the algorithm efficient in terms of energy evaluations, and is what makes it highly suitable for use in RSBI, where we only have access to the likelihood and prior's gradient. A consequence is that ASBS distinguishes between two kinds of low-density region. Because the energy is queried only at the terminal point X_1, trajectories can pass through regions of low or zero target density in transit without obstruction, unlike MCMC, whose chains must traverse every intermediate state. Regions that separate components of the terminal distribution are different. Adjoint Matching trains on trajectories drawn from the current controller, so a component that no trajectory reaches provides no training signal and can remain uncovered. The well-tempered bias of Appendix. A.1 addresses this by pushing samples towards regions not yet visited.

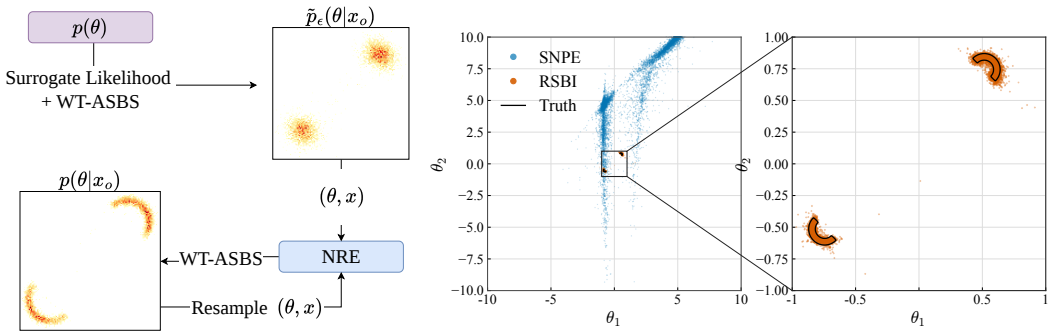

Figure 2: **Left:** Schematic representation of the two-phase procedure of RSBI. **Right:** Comparison of posteriors from RSBI (our method) and SNPE on a Two Moons task with 10^4 simulations and $\mathcal{R}_p \sim 10^{4.3}$, with two sequential steps for both SNPE and Phase 2 of RSBI.

4 Method

We adopt the SBI setting of Section 3.1: a simulator $s(\theta, \omega)$, prior $p(\theta)$, and observation x_o, with the unnormalized posterior $p(\theta \mid x_o)$ as the inference target. We additionally assume that s is differentiable in θ, so that $\nabla_\theta\, s(\theta, \omega)$ is available for any $\omega \sim p(\omega)$. A detailed illustration of RSBI can be found in Figure 2. We also provide simplified pseudocode for ASBS in Algorithm 1 and for RSBI in Algorithm 2.

4.1 Phase 1: Surrogate-Guided Zoom

When the prior-to-posterior volume ratio $\mathcal{R}_p$ is large, learning a neural likelihood or posterior over the full prior support is impractical: most simulator calls fall in regions of negligible posterior density. Phase 1 sidesteps this bottleneck with an analytic surrogate likelihood, in the spirit of ABC. Given a differentiable distance $\mathcal{D}_\epsilon(s(\theta), x_o)$ between simulated and observed data, we define the *surrogate likelihood*

$$\tilde{p}_\epsilon(x_o \mid \theta) \;\propto\; \exp\Big(-\, \mathbb{E}\, \mathcal{D}_\epsilon(s(\theta),\, x_o)\Big), \tag{6}$$

where $s(\theta)$ inside $\mathcal{D}_\epsilon$ denotes K replicates $s(\theta, \omega_{1:K})$ and the expectation is over $\omega_{1:K}$. The bandwidth $\epsilon > 0$ is the scale on which $\mathcal{D}_\epsilon$ compares simulated and observed data, and so controls how sharply $\tilde{p}_\epsilon$ concentrates around x_o. I.e, the noise variance of a Gaussian negative log-likelihood, or the length scale of an energy distance or MMD kernel. ϵ is either fixed or set adaptively from the replicates. We detail how the adaptive bandwidth is set for different surrogate likelihoods $\mathcal{D}_\epsilon$ and conduct ablations on it's choice in Appendix C. We also discuss a heuristic method that exploits a cheap survey of the gradient landscape in order to select ϵ in Appendix C.5.

The surrogate posterior $\tilde{p}_\epsilon(\theta \mid x_o) \propto \tilde{p}_\epsilon(x_o \mid \theta)\, p(\theta)$ has the Boltzmann form of (1). ASBS only queries the gradient of its energy, so $\mathcal{D}_\epsilon$ need only be known up to an additive constant:

Algorithm 1: Well-Tempered ASBS - One Round

def `wt_asbs`(*source* μ, *energy gradient* ∇E, *control* u, *corrector* h, *bias* $V(\xi(\theta))$, *stages* S)**:**
  for *stage* k *in* $1, 2, \ldots, S$ *and each epoch* **:**
    Obtain X_1 by drawing $X_0 \sim \mu$ and integrating (2)
    if k *is odd (AM stage)* **:**
      Add $\big(X_0,\, X_1,\, \nabla E(X_1) + h(X_1) + \nabla(V \circ \xi)(X_1)\big)$ to the replay buffer
      if $k < S$ **:** $V \leftarrow V +$ WT update (9) at $\xi(X_1)$
      Update u on the buffer through WT-AM (10)
    else: add (X_0, X_1) to the replay buffer and update h by solving CM (5)
  return u, h, V, *saved simulations* $\mathcal{B}$

Algorithm 2: Rapid Simulation-Based Inference

def `rsbi(`*prior* $p(\theta)$*, simulator* $s(\theta, \omega)$*, observation* x_o*, surrogate distance* $\mathcal{D}_\epsilon(s, x_o)$*, temperature* T*, ratio network* $r_\psi(\theta, x)$*, rounds* R`):`

`# Phase 1: Surrogate-guided zoom`

$E_s(\theta) \leftarrow [\mathbb{E}\,\mathcal{D}_\epsilon(s(\theta),\, x_o) - \log p(\theta)]/T$

$\bar{u},\, \bar{V},\, \mathcal{B} \leftarrow$ `wt_asbs(`$\mu = p(\theta),\, \nabla E = \nabla E_s$*, new* u*, new* h*,* $V = 0$`)`

Draw $\theta_i \sim$ `resample`$_{\bar{V}}(p_1^{\bar{u}})$, top up $\mathcal{B}$ with new $(\theta_i, s(\theta_i))$

$\mu \leftarrow p_1^{\bar{u}}$, optionally `resample`$_{\bar{V}}(p_1^{\bar{u}})$; $\quad u_0,\, h_0 \leftarrow$ fresh networks

`# Phase 2: Sequential NRE-ASBS`

for r *in* $1, 2, \ldots, R$ **:**

  Optimise $r_\psi(\theta, x)$ on $\mathcal{B}$ via the NRE-C loss (11)

  $E_r(\theta) \leftarrow -\log r_\psi(\theta, x_o) - \log p(\theta)$ (8)

  $u_r,\, h_r,\, V_r,\, _ \leftarrow$ `wt_asbs(`$\mu,\, \nabla E = \nabla E_r,\, u_{r-1},\, h_{r-1},\, V = 0$`)`

  if $r < R$ **:** draw $\theta_i \sim p_1^{u_r}$, top up $\mathcal{B}$

return $\mu,\, u_R,\, V_R$

Theorem 4.1 (Equivalence under differentiation)**.** Let $\tilde{\mathcal{D}}_\epsilon : \mathcal{X}^K \times \mathcal{X} \to \mathbb{R}$ be differentiable and equal to a positive-semi-definite (PSD) distance $\mathcal{D}_\epsilon$ up to a constant, $\mathcal{D}_\epsilon(x, x') = \tilde{\mathcal{D}}_\epsilon(x, x') + C$ for some $C \in \mathbb{R}$ and all x, x'. Then $\nabla\tilde{\mathcal{D}}_\epsilon = \nabla\mathcal{D}_\epsilon$.

The energy gradient can therefore be computed from any such $\tilde{\mathcal{D}}_\epsilon$,

$$\nabla_\theta E_s(\theta) \;=\; \mathbb{E}\,\nabla_\theta\tilde{\mathcal{D}}_\epsilon(s(\theta),\, x_o) \;-\; \nabla_\theta \log p(\theta), \tag{7}$$

where the gradient passes inside the expectation because $p(\omega)$ does not depend on θ. We hand (7) to ASBS with the prior $p(\theta)$ as source (full derivation in Appendix A.3). By Theorem 3.1, this converges to the Schrödinger-bridge solution that transports $p(\theta)$ to the surrogate posterior. Although it may seem trivial, Theorem 4.1 lets standard losses be used without their constants, such as the Gaussian NLL without its normalising constant and MSE, alongside PSD discrepancies such as the energy distance [37] and MMD [38]. A Gaussian NLL with adaptive variance is not of this form but still gives a proper surrogate posterior (Appendix C.1)

Because $\tilde{p}_\epsilon(\theta \mid x_o)$ is defined directly through the simulator rather than learned, Phase 1 needs no density estimator. Each simulator call moves the proposal towards x_o instead of providing a training example for an estimator fitted from prior draws, which is what limits sequential neural methods at large $\mathcal{R}_p$. The distance gradient points toward x_o in data space, which the differentiable simulator pulls back into parameter space via $\nabla_\theta s(\theta)$. Phase 1 therefore efficiently allocates posterior mass to high-density regions under a constrained simulator budget, producing the focused proposal $\tilde{p}_\epsilon(\theta \mid x_o)$ that Phase 2 refines. Appendix A.3 gives the admissibility class for $\mathcal{D}_\epsilon$, whilst Appendix. C.4 discusses the bandwidth–metric trade-offs inherited from ABC, and the surrogates used in Section 5.

4.2 Phase 2: Sequential NRE-ASBS Refinement

In Phase 2 we start from the surrogate posterior $\tilde{p}_\epsilon(\theta \mid x_o)$ produced by Phase 1 and refine it to recover the simulator posterior $p(\theta \mid x_o)$ (Section 3.1). Refinement proceeds over R rounds. Each round trains a neural ratio estimator [5, 6] $r_\psi(\theta, x)$ on (θ, x) pairs simulated so far, which are drawn from the round's proposal $q_r(\theta)$ as well as optionally earlier ones used during phase 1 training. It then solves the ASBS objective (3) against the resulting target energy, with $\tilde{p}_\epsilon(\theta \mid x_o)$ as the source distribution in every round. The SB therefore always transports samples from the Phase-1 posterior, and the sampler of round r provides the proposal q_{r+1} for the next round's simulations. We use the contrastive (NRE-C) loss [20] throughout. To improve mode-discovery over both phases we also optionally use a simplified version of the recent Well-Tempered metadynamics (WT-ASBS) [16] extension that augments the energy landscape, for more details on WT-ASBS we refer to Appendix A.1.

At optimum the classifier learns $r^*_\psi(\theta, x) = p(x \mid \theta)/q_r(x)$, where $q_r(x)$ is the local marginal. Because Phase 2 conducts multiple rounds of refinement, the proposal $q_r(\theta)$ changes between rounds as simulations concentrate in regions of high posterior probability. Following the same construction

as the Phase 1 energy (7), the round-r energy is:

$$-\nabla_\theta \log p(\theta \mid x_o) \;=\; -\nabla_\theta \log \frac{p(x_o \mid \theta)}{q_r(x_o)} \;-\; \nabla_\theta \log p(\theta) \;=\; \nabla_\theta E_r(\theta), \tag{8}$$

where the proposal-induced marginal $q_r(x_o)$ has zero gradient in θ and so drops out - this is a core appeal of using a diffusion sampler against an NRE-derived energy. Solving (3) via ASBS with energy $E_r(\theta)$ and source $\tilde{p}_\epsilon(\theta \mid x_o)$ therefore recovers the Schrödinger-bridge solution connecting the surrogate posterior and the simulator posterior. To avoid retraining from scratch each round we warm-start the controller u and corrector h from the previous round's parameters. Because r_ψ is only identified where the classifier has seen training parameters, a user who trusts the Phase 1 posterior to cover the simulator posterior can optionally truncate the ratio outside the region spanned by the Phase 1 samples (Appendix A.4). Because E_r in (8) explicitly retains $\log p(\theta)$ as an additive term, the round output requires no post-hoc correction for leakage outside prior support. Natural posterior leakage mitigation is an additional benefit of RSBI which we demonstrate with an additional experiment in Appendix B.2.

Because Phase 2 *transports* the Phase 1 sampler rather than refitting a global density over the prior, it remains accurate where SNPE and SNVI, warm-started from the same Phase 1 samples and given the same simulation budget, still degrade or collapse onto spurious regions of a wide prior. We demonstrate this in an ablation in Appendix B.3.

The full Phase 2 procedure including NRE-C training, the well-tempered bias update (9), and the augmented Adjoint Matching objective (10) is described in Appendices A.1 and A.2, alongside the convergence guarantee (Proposition A.1) for RSBI.

Preconditioning the Phase-2 sampler for unimodal posteriors.

The Phase 2 sampling process is isotropic, so on ill-conditioned posteriors (e.g strongly correlated parameters, narrow ridges) no single noise scale σ_t suits every direction. We therefore provide an option to run Phase 2 in coordinates whitened by the mean $\bar{\theta}$ and covariance $\Sigma = LL^\top$ of the Phase 1 samples, $z = L^{-1}(\theta - \bar{\theta})$, mapping only the energy gradient, $\nabla_z E_r = L^\top \nabla_\theta E_r$. The affine map leaves the target unchanged and costs no simulations. It is the analogue of mass-matrix adaptation in HMC [39, 40], with the metric supplied by Phase 1. As one covariance cannot describe several modes, Phase-2 preconditioning is useful for unimodal targets. For more discussion see Appendix. A.5.

5 Experiments

We benchmark RSBI against two established neural-SBI methods: SNPE-C [3] and SNVI+fKL [7] (see Appendix E for further details), on Two Moons [3], SLCP [4], Bernoulli GLM [18] and Lotka–Volterra [18], as implemented in the `sbibm` [18] benchmarking suite, as well as a new differentiable gravitational-wave ringdown task (`gw_ringdown`). To make them differentiable, we re-implement the benchmarks from `sbibm` in PyTorch. For full details on the re-implementation we refer to Appendix D. We make public our full code and checkpoints[1].

Posterior quality is measured by the classifier two-sample test (C2ST, 41) against task-appropriate reference posteriors, obtained via rejection sampling for Two Moons, `sbibm`'s reference samplers for SLCP, Bernoulli GLM, resampling the `sbibm` default for Lotka–Volterra, and MCMC with `emcee` [42] for `gw_ringdown` (see Appendix D.7 for details). For each task we sweep several prior sizes (uniform boxes, or Gaussian prior scales for Bernoulli GLM and Lotka–Volterra) and three simulation budgets $N_{\text{sim}} \in \{10^3, 10^4, 10^5\}$, using the log prior-to-posterior volume ratio $\log_{10} \mathcal{R}_p$, averaged over the observations, to quantify the volume reduction (see App. F for the definition and App. D.8 for the values adopted here). The smallest priors are the ones implemented in the `sbibm` package. All baselines use their released `sbibm` or SNVI defaults at matched total simulator-call budget, with the number of SNVI rounds set to 5 and SNPE rounds set to 2-3 (instead of 10) to match RSBI closer and reduce computational cost. Lastly, the surrogate bandwidths were taken to be adaptive where possible. The Bernoulli-GLM experiment instead implemented the heuristic criteria of Appendix C.5 to chose ϵ. `gw_ringdown` takes $\epsilon = 1$. Full task definitions, prior scenarios, hyperparameters, reference-sampler configurations and the RSBI training schedule are in Appendix D.

[1] `https://github.com/Chromeilion/rsbi_supplementary`

Compute-Matched Experiment: Because doing the full forward + backward pass through a simulator is more expensive than only a forward pass, we also investigate RSBI's performance under a compute-matched budget. These results are shown in Appendix B.1.

5.1 Main results: gravitational-wave ringdown

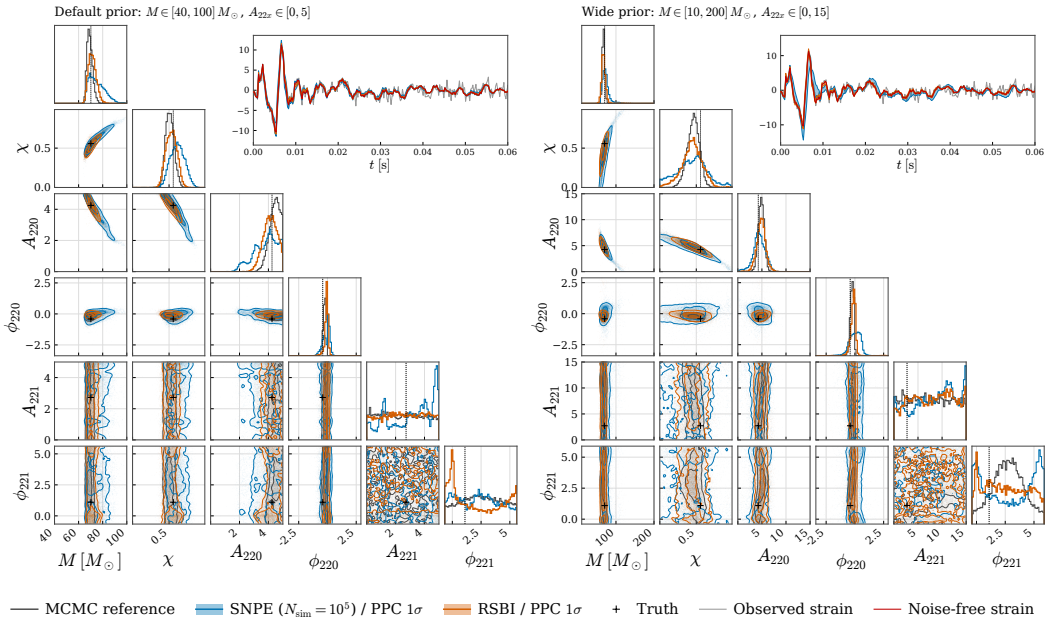

Figure 3: 6D `gw_ringdown` posteriors of RSBI, SNPE and the MCMC reference for observation 10 at $N_{\text{sim}} = 10^5$, under the default prior (left, $\log_{10} \mathcal{R}_p \approx 2.62$) and the wide prior (right, ≈ 2.84; Table 21). Insets: whitened observation (grey), noise-free injection (red) and the 1σ posterior-predictive bands of SNPE and RSBI from 200 re-simulated posterior samples each.

We first evaluate RSBI on a real-world inference task: estimating six ringdown parameters $(M, \chi, A_{220}, \varphi_{220}, A_{221}, \varphi_{221})$ from a black hole post-merger gravitational wave signal, with extrinsic dimensions fixed to truth (the natural single-detector setup with known sky position and distance). Simulator construction and reference posterior generation are described in Appendix D.2. Figure 3 shows the $d_\theta = 6$ 2-d and 1-d marginal posteriors on observation 10 at the default and wide priors from $N_{\text{sim}} = 10^5$: RSBI tracks the `emcee` reference closely across all six parameters, with a more constrained posterior in comparison to that of SNPE, as also evident in their respective C2ST scores (see Table 1) for the default prior (0.891 and 0.935, respectively) and wide prior (0.903 and 0.985, respectively). The strain inset shows the 2σ posterior-predictive band of RSBI tightly envelopes the injected ringdown signal, in comparison to the larger uncertainties of SNPE. We highlight the difference in the MCMC and RSBI posteriors, especially visible in the 1-dimensional marginal plot of A_{220} (see top row of the A_{220} corner column).

Table 1: Mean C2ST on `gw_ringdown` against the `emcee` reference at $N_{\text{sim}} = 10^5$, averaged over 10 simulated observations. For every method the phases φ_{220}, φ_{221} are wrapped to $[0, 2\pi)$ before scoring.

	Prior (see Table 21)	
Method	default	wide
$\log_{10} \mathcal{R}_p$	≈ 2.6	≈ 2.8
SNVI+fKL	0.958 (0.032)	0.986 (0.022)
SNPE	0.935 (0.071)	0.985 (0.015)
RSBI (ours)	**0.891** (**0.070**)	**0.903** (**0.069**)

5.2 Main results: robustness under $\mathcal{R}_p$ increase

Figure 4 reports C2ST against the simulation budget (N_{sim}) on the Two Moons, Lotka-Volterra, Bernoulli GLM, and SLCP benchmarks [18]. On Two Moons and Bernoulli GLM, all three methods start to converge towards 0.5 C2ST at $N_{\text{sim}} = 10^5$. On the Lotka-Volterra and SLCP benchmarks this

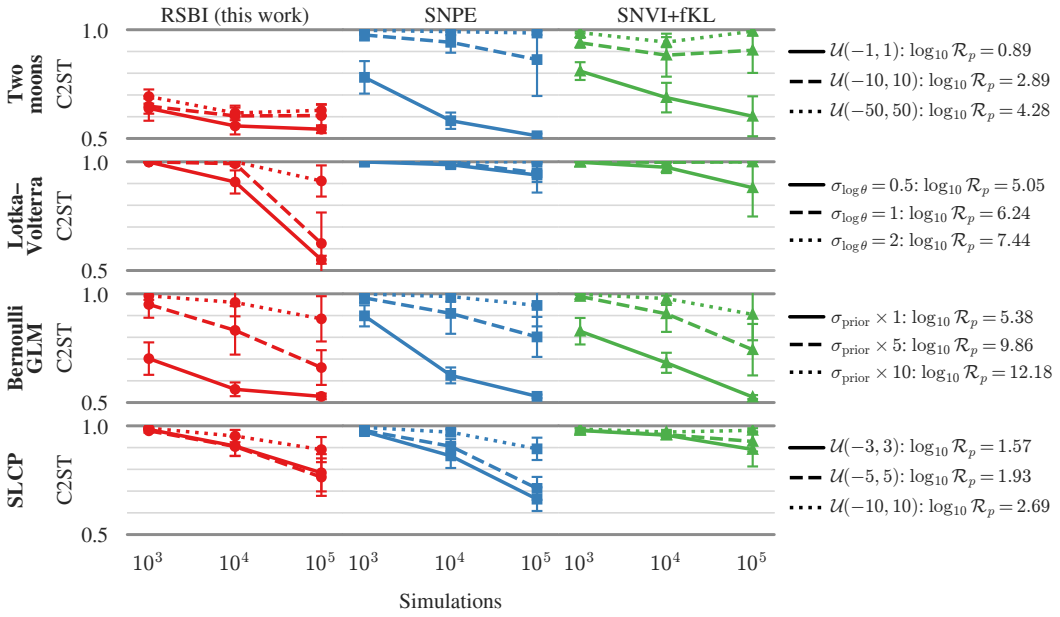

Figure 4: Robustness to prior-to-posterior volume ratio on Two Moons, Lotka–Volterra, Bernoulli GLM and SLCP benchmarks. Mean C2ST ($\pm 1\sigma$) calculated against the reference posterior vs. simulation budget N_{sim} over 10 observations. Legends report $\log_{10} \mathcal{R}_p$, full line is the `sbibm` default. C2ST $= 0.5$ matches the reference, 1.0 is fully distinguishable.

is not the case, with only RSBI converging on Lotka-Volterra. We also find that the C2ST gap widens as $\mathcal{R}_p$ increases. On Two Moons at $\log_{10} \mathcal{R}_p \approx 4.3$, the sequential sbi baselines saturate at C2ST ≈ 1 for every budget, while RSBI maintains C2ST $\lesssim 0.7$ throughout. On SLCP, all baselines degrade by 0.1–0.2 C2ST between the sbibm default and $\log_{10} \mathcal{R}_p \approx 2.69$ at 10^5 simulations. With the smallest $\mathcal{R}_p$ difference between priors, this experiment details a regime for which the gains of RSBI are not extpected to be as pronounced. On Bernoulli GLM, all methods agree at the default prior, but at $\log_{10} \mathcal{R}_p \approx 9.9$ RSBI reaches a C2ST of 0.66 at 10^5 simulations, against 0.74 for SNVI+fKL and 0.80 for SNPE, and at $\log_{10} \mathcal{R}_p \approx 12.2$ 0.89 against 0.91 and 0.95. With Lotka-Volterra we find only RSBI is able to achieve strong results within the simulation budget, reaching a C2ST of 0.55 at 10^5 simulations against 0.88 for SNVI+fKL and 0.94 for SNPE. On Lotka-Volterra we also implement the Phase 2 preconditioning of Sec.4.2 which considerably enhanced performance.

To investigate the convergence of RSBI under large simulation budgets, we also conduct an additional run on Two Moons, extending the number of sims to 10^6 on the default prior. For this run we achieve a mean C2ST ($\pm 1\sigma$) of 0.536 (± 0.014), a slight improvement over the 10^5 run, which achieves 0.543 (± 0.018).

6 Discussion

The RSBI methodology introduces two novel components that can each be adopted independently of ASBS. The first is a surrogate proposal from a differentiable simulator. (6) is an unnormalised log density that can be differentiated with respect to θ by backpropagating through the simulator, so any gradient-based method can target it. For example, VI can be run as in SNVI [7] with the learned likelihood replaced by the surrogate distance. Alternatively, ASBS can be swapped for annealed Langevin dynamics or another diffusion sampler designed for multimodal targets, which avoids the sampler-specific hyperparameters that ASBS requires.

The second component is diffusion-based NRE refinement via transport. A ratio estimator remains valid under any proposal, so every simulation is can be reused, and $\log p(\theta) + \log r_\psi(\theta, x_o)$ is sampled starting from the Phase 1 proposal rather than from the prior or a truncated version of it. The proposal is not sufficient on its own. Our ablation (Appendix B.3) shows that seeding SNPE and SNVI with the Phase 1 proposal improves them, but a substantial further gain comes only

when Phase 2 refines the Phase 1 proposal via NRE-ASBS transport. Additionally, preconditioning Phase 2 by whitening the Phase 1 proposal (Sec.4.2) can alleviate issues arising from ill-conditioned unimodal posteriors, potentially improving the performance of the ASBS. On gw_ringdown (Sec. 5.1) RSBI is the only method substantially below a C2ST of 1, but the remaining discrepancy with the reference is concentrated in the periodic phases ϕ_{220}, ϕ_{221}, which no method resolves well. The cost of backpropagating through the simulator is incurred only in Phase 1 (Appendix B.1).

A further consequence of driving the proposal with a surrogate is that it can be engineered in ways a learned estimator cannot. Since the surrogate only has to point towards the posterior, it can be relaxed for discrete simulators, as with the straight-through Bernoulli relaxation (Appendix D.5). The prior also enters both Phase 1 and Phase 2 energies exactly, mitigating posterior mass leakage outside the support (Appendix B.2), whereas truncation-based methods must correct for this after the fact.

Limitations. RSBI requires a differentiable simulator. For black-box simulators, classical sequential SBI remains the appropriate choice. A backward pass through the simulator costs 1.7 to 4.0 times a forward pass (Appendix B.1), although this applies only to the Phase 1 share of the budget. Phase 2 costs more per iteration than amortised NPE, so RSBI pays off when simulation rather than network training is the bottleneck. It is also not amortised across x_o, a new x_o requires re-running both phases. Phase 1 adds three inference-level hyperparameters, the surrogate bandwidth ϵ, the distance $\mathcal{D}_\epsilon$ and the temperature T (Appendices C.6 and C.7). Finally, MCMC is exact in the limit of infinite samples, whereas RSBI, like other neural SBI methods, is bounded by network capacity and optimization. On two moons, for example, accuracy saturates even at 10^6 simulations (Sec. 5.2).

A further set of limitations stems from our choice of ASBS as the sampler rather than from RSBI itself. ASBS is sensitive to its own hyperparameters, notably the reference noise $\sigma_{\max}$, the temperature and the training schedule. We set these per benchmark, keeping them as agnostic as possible to the surrogate bandwidth by using an adaptive variant or, for the ablations in Appendix C, by inspecting the gradient landscape. The guarantees of the RSBI target against prior leakage and mode collapse only carry over to the samples if the ASBS sampler converges (Appendix B.2). When the reference noise $\sigma_{\max}$ is comparable to the separation between modes, the mode-seeking objective and the self-refilled buffer can amplify small imbalances until modes empty. An unconverged controller can also scatter samples outside the support. These are failures of the sampler's optimisation rather than of the target, but in practice they require $\sigma_{\max}$ to be set relative to the posterior geometry. We stress that the novelty of RSBI, a surrogate-driven proposal refined against a target that retains the prior and the simulator, holds for any choice of sampler, and ASBS can be replaced by any method that requires only $\nabla_\theta E$.

7 Conclusion

We introduced RSBI, a sequential SBI method that uses gradients of a differentiable simulator to zoom from a broad prior to the posterior before any estimator is trained, subsequently refining the prediction by transporting the proposal via diffusion under an NRE target. Each RSBI component is compatible with existing sequential pipelines and with any gradient-based sampler or VI. We showed RSBI is competitive across simulation budgets on broad priors where prior-initialized methods spend most of their simulations far from x_o.

Future work. A systematic study of the gradient landscape induced by differentiable simulators would show when this signal is reliable, and could yield curricula that adapt automatically instead of being set per task. Furthermore, the Phase 2 preconditioning we utilize is a single global affine whitening estimated from Phase 1 samples (Appendix A.5). We will quantify how much preconditioning ASBS needs as the posterior becomes more anisotropic or curved, and replace the global map with adaptive or nonlinear alternatives, such as running estimates of the posterior covariance or learned transport maps.

Two high-impact scientific applications stand out. One is extreme-mass-ratio inspirals (EMRIs), where the broad-prior regime is even more pronounced than for ringdown. The other is inference with neural emulators: any trained emulator is a differentiable simulator usable within our framework, and it reduces the per-call simulator cost to a network forward pass. The RSBI target is independent of the sampler, so in future work we will pair it with other samplers such as Langevin dynamics, flow-based proposals, or ASBS with geometry-adapted reference processes.

8 Acknowledgments

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Uros Zivanovic. This work was supported by the Max Planck Artificial Intelligence Network (MP-AIX). RT and AS acknowledge funding from Next Generation EU, in the context of the National Recovery and Resilience Plan, Investment PE1 Project FAIR "Future Artificial Intelligence Research". This resource was co-financed by the Next Generation EU [DM 1555 del 11.10.22]. RT is partially supported by the Fondazione ICSC, Spoke 3 "Astrophysics and Cosmos Observations", Piano Nazionale di Ripresa e Resilienza Project ID CN00000013 "Italian Research Center on High-Performance Computing, Big Data and Quantum Computing" funded by MUR Missione 4 Componente 2 Investimento 1.4: Potenziamento strutture di ricerca e creazione di "campioni nazionali di R&S (M4C2-19)" - Next Generation EU (NGEU).

References

[1] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020. doi: 10.1073/pnas.1912789117. URL https://www.pnas.org/doi/abs/10.1073/pnas.1912789117.

[2] George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with bayesian conditional density estimation. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1028–1036, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/6aca97005c68f1206823815f66102863-Abstract.html.

[3] David S. Greenberg, Marcel Nonnenmacher, and Jakob H. Macke. Automatic posterior transformation for likelihood-free inference. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, Proceedings of Machine Learning Research, pages 2404–2414. PMLR, 2019. URL http://proceedings.mlr.press/v97/greenberg19a.html.

[4] George Papamakarios, David C. Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, Proceedings of Machine Learning Research, pages 837–848. PMLR, 2019. URL http://proceedings.mlr.press/v89/papamakarios19a.html.

[5] Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free MCMC with amortized approximate ratio estimators. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, Proceedings of Machine Learning Research, pages 4239–4248. PMLR, 2020. URL http://proceedings.mlr.press/v119/hermans20a.html.

[6] Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, Proceedings of Machine Learning Research, pages 2771–2781. PMLR, 2020. URL http://proceedings.mlr.press/v119/durkan20a.html.

[7] Manuel Glöckler, Michael Deistler, and Jakob H. Macke. Variational methods for simulation-based inference. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=kZOUYdhqkNY.

[8] Benjamin Kurt Miller, Alex Cole, Patrick Forré, Gilles Louppe, and Christoph Weniger. Truncated marginal neural ratio estimation. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://papers.neurips.cc/paper_files/paper/2021/file/01632f7b7a127233fa1188bd6c2e42e1-Paper.pdf.

[9] Michael Deistler, Pedro J. Gonçalves, and Jakob H. Macke. Truncated proposals for scalable and hassle-free simulation-based inference. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, pages 23135–23149. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9278abf072b58caf21d48dd670b4c721-Abstract-Conference.html.

[10] Maximilian Dax, Stephen R. Green, Jonathan Gair, Nihar Gupte, Michael Pürrer, Vivien Raymond, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Real-time inference for binary neutron star mergers using machine learning. *Nature*, 639:49–53, 2025. doi: 10.1038/s41586-025-08593-z. URL https://www.nature.com/articles/s41586-025-08593-z.

[11] Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Neural importance sampling for rapid and reliable gravitational-wave inference. *Physical Review Letters*, 130(17):171403, 2023. doi: 10.1103/PhysRevLett.130.171403. URL https://doi.org/10.1103/PhysRevLett.130.171403.

[12] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Yash Katariya, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/jax-ml/jax.

[13] Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, CK Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Michael Suo, Phil Tillet, Eikan Wang, Xiaodong Wang, William Wen, Shunting Zhang, Xu Zhao, Keren Zhou, Richard Zou, Ajit Mathews, Gregory Chanan, Peng Wu, and Soumith Chintala. PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, April 2024. doi: 10.1145/3620665.3640366. URL https://docs.pytorch.org/assets/pytorch2-2.pdf.

[14] Guan-Horng Liu, Jaemoo Choi, Yongxin Chen, Benjamin Kurt Miller, and Ricky T. Q. Chen. Adjoint schrödinger bridge sampler. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/174692c52dc84fad2b2e99dd8637ce6a-Abstract-Conference.html.

[15] Aaron J. Havens, Benjamin Kurt Miller, Bing Yan, Carles Domingo-Enrich, Anuroop Sriram, Daniel S. Levine, Brandon M. Wood, Bin Hu, Brandon Amos, Brian Karrer, Xiang Fu, Guan-Horng Liu, and Ricky T. Q. Chen. Adjoint sampling: Highly scalable diffusion samplers via adjoint matching. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR, 2025. URL https://proceedings.mlr.press/v267/havens25a.html.

[16] Juno Nam, Bálint Máté, Artur P. Toshev, Manasa Kaniselvan, Rafael Gomez-Bombarelli, Ricky T. Q. Chen, Brandon M. Wood, Guan-Horng Liu, and Benjamin Kurt Miller. Enhancing diffusion-based sampling with molecular collective variables. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=1bJN1EQByS.

[17] Alessandro Barducci, Giovanni Bussi, and Michele Parrinello. Well-tempered metadynamics: A smoothly converging and tunable free-energy method. *Phys. Rev. Lett.*, 100:020603, Jan 2008.

doi: 10.1103/PhysRevLett.100.020603. URL https://link.aps.org/doi/10.1103/PhysRevLett.100.020603.

[18] Jan-Matthis Lueckmann, Jan Boelts, David Greenberg, Pedro Goncalves, and Jakob Macke. Benchmarking simulation-based inference. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 343–351. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/lueckmann21a.html.

[19] George Papamakarios, Eric T. Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.*, 22(57):1–64, 2021. URL https://jmlr.org/papers/v22/19-1028.html.

[20] Benjamin Kurt Miller, Christoph Weniger, and Patrick Forré. Contrastive neural ratio estimation. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/159f7fe5b51ecd663b85337e8e28ce65-Abstract-Conference.html.

[21] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 12 2002. ISSN 1943-2631. doi: 10.1093/genetics/162.4.2025. URL https://doi.org/10.1093/genetics/162.4.2025.

[22] Alexander Moreno, Tameem Adel, Edward Meeds, James M. Rehg, and Max Welling. Automatic variational abc, 2016. URL https://arxiv.org/abs/1606.08549.

[23] Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 681–688. Omnipress, 2011. URL https://icml.cc/2011/papers/398_icmlpaper.pdf.

[24] Lorenzo Pacchiardi, Sherman Khoo, and Ritabrata Dutta. Generalized Bayesian likelihood-free inference. *Electronic Journal of Statistics*, 18(2):3628 – 3686, 2024. doi: 10.1214/24-EJS2283. URL https://doi.org/10.1214/24-EJS2283.

[25] Yin Li, Chirag Modi, Drew Jamieson, Yucheng Zhang, Libin Lu, Yu Feng, François Lanusse, and Leslie Greengard. Differentiable cosmological simulation with the adjoint method. *The Astrophysical Journal Supplement Series*, 270(2):36, feb 2024. doi: 10.3847/1538-4365/ad0ce7. URL https://doi.org/10.3847/1538-4365/ad0ce7.

[26] Thomas D. P. Edwards, Kaze W. K. Wong, Kelvin K. H. Lam, Adam Coogan, Daniel Foreman-Mackey, Maximiliano Isi, and Aaron Zimmerman. Differentiable and hardware-accelerated waveforms for gravitational wave data analysis. *Phys. Rev. D*, 110:064028, Sep 2024. doi: 10.1103/PhysRevD.110.064028. URL https://link.aps.org/doi/10.1103/PhysRevD.110.064028.

[27] Henrik Christiansen, Takashi Maruyama, Federico Errica, Viktor Zaverkin, Makoto Takamoto, and Francesco Alesiani. Fast, modular, and differentiable framework for machine learning-enhanced molecular simulations. *The Journal of Chemical Physics*, 163(18):182501, 11 2025. ISSN 0021-9606. doi: 10.1063/5.0277356. URL https://doi.org/10.1063/5.0277356.

[28] Satvik Mishra, Roberto Trotta, and Matteo Viel. Large, fast, and accurate H I intensity maps with latent overlap diffusion. *Monthly Notices of the Royal Astronomical Society*, 545(3): staf2071, January 2026. doi: 10.1093/mnras/staf2071. URL https://doi.org/10.1093/mnras/staf2071.

[29] Johann Brehmer, Gilles Louppe, Juan Pavez, and Kyle Cranmer. Mining gold from implicit models to improve likelihood-free inference. *Proceedings of the National Academy of Sciences*, 117(10):5242–5249, 2020. doi: 10.1073/pnas.1915980117. URL https://www.pnas.org/doi/abs/10.1073/pnas.1915980117.

[30] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=rkE3y85ee.

[31] Justine Zeghal, François Lanusse, Alexandre Boucaud, Benjamin Remy, and Eric Aubourg. Neural posterior estimation with differentiable simulators. In *ICML 2022 Workshop on Machine Learning for Astrophysics*, 2022.

[32] Jean-Michel Marin, Pierre Pudlo, Christian P. Robert, and Robin J. Ryder. Approximate bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, Nov 2012. ISSN 1573-1375. doi: 10.1007/s11222-011-9288-2. URL https://doi.org/10.1007/s11222-011-9288-2.

[33] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 06 1953. ISSN 0021-9606. doi: 10.1063/1.1699114. URL https://doi.org/10.1063/1.1699114.

[34] Valentin De Bortoli, Michael J. Hutchinson, Peter Wirnsberger, and Arnaud Doucet. Target score matching. *CoRR*, abs/2402.08667, 2024. doi: 10.48550/ARXIV.2402.08667. URL https://doi.org/10.48550/arXiv.2402.08667.

[35] Angus Phillips, Hai-Dang Dau, Michael John Hutchinson, Valentin De Bortoli, George Deligiannidis, and Arnaud Doucet. Particle denoising diffusion sampler. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, Proceedings of Machine Learning Research, pages 40688–40724. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/phillips24a.html.

[36] S. Kullback. Probability densities with given marginals. *The Annals of Mathematical Statistics*, 39(4):1236–1243, 1968. ISSN 00034851, 21688990. URL http://www.jstor.org/stable/2239692.

[37] Maria L. Rizzo and Gábor J. Székely. Energy distance. *WIREs Computational Statistics*, 8(1):27–38, 2016. doi: https://doi.org/10.1002/wics.1375. URL https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.1375.

[38] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. A kernel two-sample test. *J. Mach. Learn. Res.*, 13:723–773, 2012. URL https://dl.acm.org/doi/10.5555/2503308.2188410.

[39] Radford M. Neal. MCMC using Hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, editors, *Handbook of Markov Chain Monte Carlo*, chapter 5, pages 113–162. Chapman & Hall/CRC, 2011.

[40] Matthew D. Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014. URL https://dl.acm.org/doi/10.5555/2627435.2638586.

[41] David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=SJkXfE5xx.

[42] Daniel Foreman-Mackey, David W. Hogg, Dustin Lang, and Jonathan Goodman. emcee: The mcmc hammer. *Publications of the Astronomical Society of the Pacific*, 125(925):306–312, feb 2013. doi: 10.1086/670067. URL https://doi.org/10.1086/670067.

[43] P. G. Bissiri, C. C. Holmes, and S. G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):1103–1130, 11 2016. ISSN 1369-7412. doi: 10.1111/rssb.12158. URL https://doi.org/10.1111/rssb.12158.

[44] Gareth O. Roberts and Jeffrey S. Rosenthal. Optimal scaling for various metropolis-hastings algorithms. *Statistical Science*, 16(4):351–367, 2001. ISSN 08834237. URL http://www.jstor.org/stable/3182776.

[45] Arnak S. Dalalyan. Theoretical guarantees for approximate sampling from smooth and log-concave densities. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 79 (3):651–676, 06 2017. ISSN 1369-7412. doi: 10.1111/rssb.12183. URL https://doi.org/10.1111/rssb.12183.

[46] Heikki Haario, Eero Saksman, and Johanna Tamminen. An adaptive metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001. ISSN 1350-7265.

[47] Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32, 2017. doi: 10.18637/jss.v076.i01. URL https://www.jstatsoft.org/index.php/jss/article/view/v076i01.

[48] Mark Girolami and Ben Calderhead. Riemann manifold langevin and hamiltonian monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2): 123–214, 2011. doi: 10.1111/j.1467-9868.2010.00765.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2010.00765.x.

[49] Matthew Hoffman, Pavel Sountsov, Joshua V. Dillon, Ian Langmore, Dustin Tran, and Srinivas Vasudevan. Neutra-lizing bad geometry in hamiltonian monte carlo using neural transport, 2019. URL https://arxiv.org/abs/1903.03704.

[50] Adam Perbost, Francis Bach, and Pierre Marion. Diffusion removes Langevin's conditioning dependence: A sharp Gaussian analysis. *arXiv preprint arXiv:2610.12052*, 2026.

[51] Shadab Ahamed, Eshed Gal, Md Shahriar Rahim Siddiqui, Simon Ghyselincks, Moshe Eliasof, and Eldad Haber. Preconditioned flow matching. In *Advances in Neural Information Processing Systems*, volume 39, 2026. URL https://arxiv.org/abs/2603.02337. To appear. Preprint version: arXiv:2603.02337.

[52] Jonathan Goodman and Jonathan Weare. Ensemble samplers with affine invariance. *Communications in Applied Mathematics and Computational Science*, 5(1):65–80, 2010. doi: 10.2140/camcos.2010.5.65.

[53] Alvaro Tejero-Cantero, Jan Boelts, Michael Deistler, Jan-Matthis Lueckmann, Conor Durkan, Pedro J. Gonçalves, David S. Greenberg, and Jakob H. Macke. sbi: A toolkit for simulation-based inference. *Journal of Open Source Software*, 5(52):2505, 2020. doi: 10.21105/joss.02505. URL https://doi.org/10.21105/joss.02505.

[54] George Papamakarios, Iain Murray, and Theo Pavlakou. Masked autoregressive flow for density estimation. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2338–2347, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/6c1da886822c67822bcf3679d04369fa-Abstract.html.

[55] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7509–7520, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/7ac71d433f282034e088473244df8c02-Abstract.html.

[56] Paul Fearnhead and Dennis Prangle. Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474, 2012. doi: 10.1111/j.1467-9868.2011.01010.x. URL https://doi.org/10.1111/j.1467-9868.2011.01010.x.

[57] Simon N. Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310):1102–1104, 2010. doi: 10.1038/nature09319. URL https://doi.org/10.1038/nature09319.

[58] Leah F. Price, Christopher C. Drovandi, Anthony Lee, and David J. Nott. Bayesian synthetic likelihood. *Journal of Computational and Graphical Statistics*, 27(1):1–11, 2018. doi: 10.1080/10618600.2017.1302882. URL https://doi.org/10.1080/10618600.2017.1302882.

[59] Espen Bernton, Pierre E. Jacob, Mathieu Gerber, and Christian P. Robert. Approximate Bayesian computation with the Wasserstein distance. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(2):235–269, 2019. doi: 10.1111/rssb.12312. URL https://doi.org/10.1111/rssb.12312.

[60] Mijung Park, Wittawat Jitkrittum, and Dino Sejdinovic. K2-abc: Approximate bayesian computation with kernel embeddings. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 398–407, Cadiz, Spain, 09–11 May 2016. PMLR. URL https://proceedings.mlr.press/v51/park16.html.

[61] Bai Jiang. Approximate Bayesian computation with Kullback-Leibler divergence as data discrepancy. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 1711–1721, 2018. URL https://proceedings.mlr.press/v84/jiang18a.html.

[62] Hien Duy Nguyen, Julyan Arbel, Hongliang Lü, and Florence Forbes. Approximate Bayesian computation via the energy statistic. *IEEE Access*, 8:131683–131698, 2020. doi: 10.1109/ACCESS.2020.3009878. URL https://doi.org/10.1109/ACCESS.2020.3009878.

[63] Christopher Drovandi and David T. Frazier. A comparison of likelihood-free methods with and without summary statistics. *Statistics and Computing*, 32(3):42, 2022. doi: 10.1007/s11222-022-10092-4. URL https://doi.org/10.1007/s11222-022-10092-4.

[64] Richard David Wilkinson. Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. *Statistical Applications in Genetics and Molecular Biology*, 12(2):129–141, 2013. doi: 10.1515/sagmb-2013-0010. URL https://doi.org/10.1515/sagmb-2013-0010.

[65] David A. Nix and Andreas S. Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 1, pages 55–60. IEEE, 1994. doi: 10.1109/ICNN.1994.374138. URL https://ieeexplore.ieee.org/document/374138/.

[66] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/9ef2ed4b7fd2c810847ffa5fa85bce38-Abstract.html.

[67] Gábor J. Székely and Maria L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143(8):1249–1272, 2013. ISSN 0378-3758. doi: 10.1016/j.jspi.2013.03.018. URL https://www.sciencedirect.com/science/article/pii/S0378375813000633.

[68] Dino Sejdinovic, Bharath Sriperumbudur, Arthur Gretton, and Kenji Fukumizu. Equivalence of distance-based and RKHS-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5):2263–2291, 2013. doi: 10.1214/13-AOS1140. URL https://projecteuclid.org/journals/annals-of-statistics/volume-41/issue-5/Equivalence-of-distance-based-and-RKHS-based-statistics-in-hypothesis/10.1214/13-AOS1140.full.

[69] Michael G. B. Blum. Approximate bayesian computation: A nonparametric perspective. *Journal of the American Statistical Association*, 105(491):1178–1187, 2010. ISSN 01621459. URL http://www.jstor.org/stable/27920142.

[70] Marco Crisostomi, Kallol Dey, Enrico Barausse, and Roberto Trotta. Neural posterior estimation with guaranteed exact coverage: The ringdown of gw150914. *Phys. Rev. D*, 108:044029, Aug 2023. doi: 10.1103/PhysRevD.108.044029. URL https://link.aps.org/doi/10.1103/PhysRevD.108.044029.

[71] Abhirup Ghosh, Nathan K. Johnson-McDaniel, Archisman Ghosh, Chandra Kant Mishra, Parameswaran Ajith, Walter Del Pozzo, Christopher P. L. Berry, Alex B. Nielsen, and Lionel London. Testing general relativity using gravitational wave signals from the inspiral, merger and ringdown of binary black holes. *Classical and Quantum Gravity*, 35(1):014002, January 2018. doi: 10.1088/1361-6382/aa972e.

[72] Swetha Bhagwat, Xisco Jiménez Forteza, Paolo Pani, and Valeria Ferrari. Ringdown overtones, black hole spectroscopy, and no-hair theorem tests. *Phys. Rev. D*, 101:044033, Feb 2020. doi: 10.1103/PhysRevD.101.044033. URL https://link.aps.org/doi/10.1103/PhysRevD.101.044033.

[73] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/a98846e9d9cc01cfb87eb694d946ce6b-Abstract-Conference.html.

[74] Luke Metz, C. Daniel Freeman, Samuel S. Schoenholz, and Tal Kachman. Gradients are not all you need. *CoRR*, abs/2111.05803, 2021. doi: 10.48550/ARXIV.2111.05803. URL https://doi.org/10.48550/arXiv.2111.05803.

[75] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In Andrea Pohoreckyj Danyluk, Léon Bottou, and Michael L. Littman, editors, *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, volume 382 of *ACM International Conference Proceeding Series*, pages 41–48. ACM, 2009. doi: 10.1145/1553374.1553380. URL https://doi.org/10.1145/1553374.1553380.

[76] H.G. Bock and K.J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. *IFAC Proceedings Volumes*, 17(2):1603–1608, 1984. ISSN 1474-6670. doi: 10.1016/S1474-6670(17)61205-9. URL https://www.sciencedirect.com/science/article/pii/S1474667017612059. 9th IFAC World Congress: A Bridge Between Control Science and Technology, Budapest, Hungary, 2-6 July 1984.

[77] Henning U. Voss, Jens Timmer, and Jürgen Kurths. Nonlinear dynamical system identification from uncertain and indirect measurements. *Int. J. Bifurc. Chaos*, 14(6):1905–1933, 2004. doi: 10.1142/S0218127404010345. URL https://doi.org/10.1142/S0218127404010345.

[78] Evren Mert Turan and Johannes Jäschke. Multiple shooting for training neural differential equations on time series. *IEEE Control. Syst. Lett.*, 6:1897–1902, 2022. doi: 10.1109/LCSYS.2021.3135835. URL https://doi.org/10.1109/LCSYS.2021.3135835.

[79] J. O. Ramsay, G. Hooker, D. Campbell, and J. Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007. doi: 10.1111/j.1467-9868.2007.00610.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2007.00610.x.

[80] Nicholas G. Polson, James G. Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013. doi: 10.1080/01621459.2013.829001. URL https://doi.org/10.1080/01621459.2013.829001.

Appendix contents

A Additional methodological details **19**

A.1 Countering Mode-Seeking 19

A.2 Training the ratio estimator 19

A.3 Derivation of the Surrogate Energy 20

A.4 Optional truncation of the ratio estimator 20

A.5 Preconditioning the Phase 2 sampler 21

B Additional Experiments **22**

B.1 Compute-Matched Performance 22

B.2 Posterior mass leakage mitigation 23

B.3 Ablation: replacing Phase 2 by standard sequential methods 25

C Surrogate Likelihoods: Details and ablations **27**

C.1 Gaussian negative log-likelihood with adaptive variance 27

C.2 Energy distance 29

C.3 Maximum mean discrepancy (MMD) 29

C.4 Bandwidth and Metric Trade-offs 30

C.5 Bandwidth and temperature hyperparameter guesses via energy and gradient landscape 30

C.6 Surrogate likelihood ablations 32

C.7 Surrogate bandwidth ablations 36

D Task details **40**

D.1 Compute Usage 41

D.2 GW ringdown 41

D.3 SLCP 42

D.4 Two Moons 42

D.5 Bernoulli GLM 43

D.6 Lotka–Volterra 44

D.7 Reference posteriors 45

D.8 Prior scenarios and volume ratios 46

E Baselines **47**

F Volume estimator for $\mathcal{R}_p$ **48**

G RSBI Hyperparameters **48**

G.1 Main Benchmarks 48

G.2 `gw_ringdown` 53

A Additional methodological details

A.1 Countering Mode-Seeking

RSBI is variational, so training the controlled SDE against E_r amounts to minimising a reverse KL between the controlled-SDE marginal and the round-r target. Standard reverse-KL VI is mode-seeking [7]: when the target has multiple disconnected modes, the optimal proposal concentrates on whichever mode is cheapest to cover and abandons the rest. ASBS partially counters this at no extra cost: its replay buffer maintains trajectories from earlier in training, preventing the controller from collapsing onto a single mode within an ASBS round, and its small base diffusion coefficient σ preserves separation between modes that have already been discovered. Neither mechanism, however, drives the controller toward modes it has not yet visited; on multimodal posteriors with high inter-mode barriers, the ASBS optimum can still be a strict subset of the true target's support.

We address this directly with the well-tempered ASBS extension of Nam et al. [16]: a low-dimensional collective variable $\xi : \Theta \to \mathcal{S} \subseteq \mathbb{R}^m$ ($m \le d_\theta$, chosen per-task) is introduced, and a bias $V_k(\xi(\theta))$ is deposited online during training. The bias accumulates in regions of CV space the sampler has already visited and pushes future samples toward novel ones. With kernel height h_{wt}, kernel width σ_{cv}, and bias factor $\gamma > 1$, the update rule is

$$V_{k+1}(s) \;=\; V_k(s) \;+\; h_{\mathrm{wt}} \sum_{i=1}^{N} \exp\!\Big(-\frac{\beta}{\gamma-1}\, V_k\big(\xi(\theta^{(i)})\big)\Big)\, K_{\sigma_{\mathrm{cv}}}\Big(s,\, \xi(\theta^{(i)})\Big)\,, \tag{9}$$

where $K_{\sigma_{\mathrm{cv}}}$ is a Gaussian kernel and the $\theta^{(i)}$ are the i.i.d. samples produced by the current controlled SDE. The well-tempering factor $\exp(-\beta V_k/(\gamma-1))$ shrinks the deposition rate as bias accumulates, so V_k converges to a finite limit rather than diverging. The Adjoint Matching objective (4) is augmented with the bias gradient through the chain rule:

$$u^{(k)} = \arg\min_{u}\, \mathbb{E}_{p^{\mathrm{base}}_{t|0,1}\, p^{\bar{u}}_{0,1}} \Big[\big\| u_t(X_t) + \sigma_t\big(\nabla E(X_1) + \nabla(V_k \circ \xi)(X_1) + h^{(k-1)}(X_1)\big)\big\|^2\Big]. \tag{10}$$

After training, the bias is removed via importance reweighting by $w(\theta) \propto \exp(\beta V^*(\xi(\theta)))$, recovering weighted samples from the original Phase-2 target.

Proposition A.1 (Convergence of RSBI)**.** Within each Phase-2 round, under the regularity conditions of Proposition 3.1 of Nam et al. [16], compact CV domain, positive-definite Gaussian kernel, and Robbins-Monro step schedule, the bias V_k in (9) converges almost surely to $V^*(s) = -(1 - 1/\gamma)\, F_r(s) + \mathrm{const}$, where $F_r(s)$ is the round-r PMF along ξ. Combined with Theorem 3.1 and the consistency of the NRE-C estimator $r_{\psi_r} \to p(x \mid \theta)/q_r(x)$ (Appendix A.2), when Algorithm 2 is run to convergence the round-r marginal of the controlled SDE, reweighted by w, converges to the target posterior $p(\theta \mid x_o)$.

Algorithm 2 details the full RSBI procedure. The structure mirrors that of Glöckler et al. [7]: a proposal is refined across rounds, but the target $p(\theta \mid x_o)$ is held fixed throughout and reweighting is analytic rather than learned.

A.2 Training the ratio estimator

Phase 2's energy E_r requires a ratio estimator $r_{\psi_r}(\theta, x_o)$ that tracks the log-ratio $\log p(x_o \mid \theta)/q_r(x_o)$, whose dependence on θ is proposal-invariant. We train r_ψ with the contrastive NRE-C loss of Miller et al. [20], which generalises NRE-A [5] and NRE-B [6]. Given a minibatch of B paired samples $(\theta_b, x_b)_{b=1}^{B}$ with $\theta_b \sim q_r(\theta)$ and $x_b = s(\theta_b, \omega_b)$ from the round-r training set $\mathcal{B}$ of Algorithm 2, where q_r is the mixture of all proposals so far, plus in-batch contrastive parameters $\theta_k \sim q_r$ drawn for each anchor from the other pairs, the NRE-C objective with K_{class} classes is

$$\begin{aligned}\mathcal{L}_{\text{NRE-C}}(\psi) \;=\; & -\frac{1}{1+\eta}\, \mathbb{E}_{q_r(x)\, q_r^{\otimes K_{\mathrm{class}}}(\theta_{1:K_{\mathrm{class}}})} \left[\log \frac{K_{\mathrm{class}}}{K_{\mathrm{class}} + \eta \sum_{k=1}^{K_{\mathrm{class}}} r_\psi(\theta_k, x)}\right] \\ & -\frac{\eta}{1+\eta}\, \mathbb{E}_{q_r(\theta_0)\, p(x|\theta_0)\, q_r^{\otimes (K_{\mathrm{class}}-1)}(\theta_{1:K_{\mathrm{class}}-1})} \left[\log \frac{\eta\, r_\psi(\theta_0, x)}{K_{\mathrm{class}} + \eta \sum_{k=0}^{K_{\mathrm{class}}-1} r_\psi(\theta_k, x)}\right],\end{aligned} \tag{11}$$

where η weights the joint against the marginal term. We use $\eta = 1$ throughout (NRE-B is recovered as $\eta \to \infty$, and NRE-A at $K_{\text{class}} = 1$, $\eta = 1$). The minimiser is the ratio $r^*_\psi(\theta, x) = p(x \mid \theta)/q_r(x)$, whose gradient in θ does not depend on the proposal q_r used to generate training data, the property our Phase-2 derivation in Section 4.2 explicitly relies on, and the property that distinguishes the NRE-C target from the proposal-conditioned target of SNPE-C/APT [3].

A.3 Derivation of the Surrogate Energy

Starting from the surrogate posterior $\tilde{p}_\epsilon(\theta \mid x_o)$ specified through the surrogate likelihood (6):

$$\log \tilde{p}_\epsilon(\theta \mid x_o) = \; - \, \mathbb{E}\, \mathcal{D}_\epsilon(s(\theta), x_o) + \log p(\theta). \tag{12}$$

This has the same exponential structure as the Boltzmann distribution (1), with the energy occupying the entire exponent. Sign-reversing and differentiating gives the surrogate energy gradient under a sufficient distance measure $\mathcal{D}_\epsilon$:

$$\nabla_\theta E_s(\theta) \;=\; \mathbb{E}\, \nabla_\theta \mathcal{D}_\epsilon(s(\theta), x_o) \;-\; \nabla_\theta \log p(\theta).$$

For $\mathcal{D}_\epsilon$ to define a proper probability density $\tilde{p}_\epsilon(\theta \mid x_o)$, the requirement is that of a loss in a general Bayesian update [43]:

Definition A.1 (Sufficient distance measure)**.** A function $\mathcal{D}_\epsilon : \mathcal{X}^K \times \mathcal{X} \to \mathbb{R}$ is a *sufficient distance measure* for a prior $p(\theta)$ and observation x_o if it is differentiable in its first argument and the surrogate posterior has a finite, non-zero normalising constant:

$$0 \;<\; \int_\Theta \exp\Big(- \, \mathbb{E}\, \mathcal{D}_\epsilon\big(s(\theta), x_o\big)\Big)\, p(\theta)\, d\theta \;<\; \infty.$$

Since by Theorem 4.1, any differentiable $\tilde{\mathcal{D}}_\epsilon$ that equals $\mathcal{D}_\epsilon$ up to an additive constant satisfies $\nabla\tilde{\mathcal{D}}_\epsilon = \nabla\mathcal{D}_\epsilon$, as far as ASBS is concerned (7) we can replace $\mathcal{D}_\epsilon$ with $\tilde{\mathcal{D}}_\epsilon$, equivalent under differentiation. In practice this relaxation matters because it admits standard losses whose constants are conventionally dropped, such as the $\frac{d_x}{2}\log 2\pi$ of the Gaussian NLL.

A.4 Optional truncation of the ratio estimator

A ratio estimator is only identified where its training data have support. Trained on (θ_i, x_i) with $\theta_i \sim q_r(\theta)$, the classifier $r_\psi(\theta, x_o)$ is constrained near the bulk of q_r, but in regions the proposal never visited its value is an extrapolation of the network. Because the Phase 2 sampler minimises $E_r = -\log r_\psi - \log p(\theta)$ over the whole prior, a spuriously large extrapolated ratio can pull samples into regions with no posterior mass. We optionally allow RSBI to guard against this by restricting where the ratio is trusted. This may be the case, for example, when the Phase 1 zoom is assumed to be locally concentrated around the true posterior mass. The choice firmly remains with the user.

Let $\{\theta_j\}$ be the parameters the region is fitted on (the Phase 1 sampler draws, or all NRE training parameters of the round). We compute their mean $\bar{\theta}$ and covariance Σ, discard the points whose Mahalanobis distance exceeds the q-quantile of the distances, and recompute $\bar{\theta}$ and Σ on the remaining core, so that outlying draws do not inflate the region. The distance of a parameter to the core is $d(\theta) = \sqrt{(\theta - \bar{\theta})^\top \Sigma^{-1} (\theta - \bar{\theta})}$. For multimodal posteriors a single ellipsoid around separated modes would cover most of the prior, so we first split $\{\theta_j\}$ into J clusters with k-means, fit $(\bar{\theta}_c, \Sigma_c)$ per cluster as above and set $d(\theta) = \min_c d_c(\theta)$. The truncated ratio is

$$\log \bar{r}_\psi(\theta, x_o) = \log r_\psi(\theta, x_o) - \lambda\, \mathrm{relu}\big(d(\theta) - k\big)^2, \tag{13}$$

which leaves the ratio unchanged where $d(\theta) \le k$ and adds a soft quadratic wall of strength λ beyond. Phase 2 then uses $E_r = -\log \bar{r}_\psi - \log p(\theta)$. The wall is refitted after every NRE update, acts only on the ratio, leaves the prior term untouched and needs no simulations beyond those already in the budget.

Restricting inference to a region that provably contains the posterior is a common device in sequential SBI. Truncated sequential NPE (TSNPE) [9] draws each round's simulations from the prior restricted to the highest-probability region of the current posterior estimate. Truncated marginal NRE (TMNRE)

[8] shrinks the prior to a box around the region where the ratio estimate is non-negligible and retrains within it. All of these methods, and ours, rest on the same observation: a learned density or ratio is only reliable where it has data, so inference is confined to that region rather than trusted everywhere. Our variant differs in three respects. (i) The region is fitted on samples that are already available (the Phase 1 output and the round's training parameters), so it costs no additional simulations. (ii) The truncation is soft and acts on the energy rather than on the prior or the proposal, so the sampler is steered back into the region rather than forbidden from leaving it, and the prior in E_r stays the original one. (iii) The region is an ellipsoid (or a union of ellipsoids), not an axis-aligned box, and so follows correlated posteriors.

Inside $\{d(\theta) \le k\}$ the truncated energy equals the untruncated one, so the Phase 2 target is unchanged wherever the region covers the posterior. As for TSNPE and TMNRE, the truncation biases the result only if non-negligible posterior mass lies outside the region, which k and the trim quantile q control. A larger k is safer but restricts less, and a smaller q fits a tighter core. The wall is a safeguard against extrapolation and is not needed when the ratio is well behaved, so it is optional and off unless stated otherwise.

Among the benchmark tasks the truncation is used only for Lotka–Volterra (App. D.6), where it is enabled for all prior widths and simulation budgets. Its Phase 1 posterior reliably covers the reference posterior there, while the oscillatory likelihood makes the ratio extrapolate poorly away from the training parameters. We use a single cluster ($J = 1$) fitted on the NRE training parameters of the current round and refitted every round, with $\lambda = 50$, $k = 3$ and $q = 0.99$. The only exception is the large prior at $N_{\text{sim}} = 10^5$, where $k = 2$ and $q = 0.9$ give a tighter core. For Lotka–Volterra, $\log r_\psi$ is in addition softly capped at the 0.999-quantile of its values over the training parameters plus a margin of 2, which guards against the same far-field blow-up of the classifier logits. Two Moons, SLCP, Bernoulli GLM and `gw_ringdown` use the untruncated ratio.

A.5 Preconditioning the Phase 2 sampler

The Phase 2 sampler is an adjoint Schrödinger bridge sampler (ASBS; 14) whose reference process is isotropic with the same noise scale σ_t is applied in every direction of θ-space. This choice is natural when the target has a comparable width in all directions, but it is poorly suited to *ill-conditioned* targets, whose posterior is very narrow along some directions and broad along others. Such geometry arises whenever parameters are strongly correlated, when the posterior concentrates on a curved ridge, or when the posterior widths of different parameters differ by orders of magnitude. A noise scale small enough to resolve the stiff directions moves far too slowly along the shallow ones, while a noise scale matched to the shallow directions overshoots the stiff ones; in either case the sampler concentrates poorly within a fixed training budget. The same difficulty is well known for gradient-based MCMC: the efficiency of the Metropolis-adjusted and unadjusted Langevin algorithms degrades with the condition number of the target [44, 45], and Hamiltonian Monte Carlo with an identity mass matrix must choose a step size limited by the narrowest direction [39]. The standard remedy is to precondition the dynamics with a metric that approximates the posterior covariance, either fixed [39], adapted from earlier samples [40, 46, 47], position-dependent [48], or given by a learned transport map [49]. For diffusion samplers the difficulty lies in learning rather than in sampling: on Gaussian targets the sampling error of a diffusion sampler with optimised hyperparameters does not depend on the condition number [50], but regression-trained drifts fit high-variance directions quickly and low-variance ones slowly when the target is ill-conditioned, which preconditioning into an isotropic space removes [51].

In our method, Phase 1 already provides the information needed for such a metric, at no extra simulation cost. Its samples are concentrated around the posterior: we use the round-0 NRE training parameters $\{\theta_j\}$ for Lotka–Volterra, and 10^4 fresh draws from the trained Phase-1 sampler for `gw_ringdown`. Their mean $\bar\theta$ and covariance $\Sigma = LL^\top$, with L the Cholesky factor, are estimated from the samples whose Mahalanobis distance to the sample mean lies below its q-quantile, where q is the *whitening quantile*. We use $q = 0.99$, so that a few stray Phase-1 samples do not inflate the covariance (at 10^5 simulation budget, $q = 0.9$ for the large Lotka–Volterra prior, whose Phase-1 proposal has heavier tails). Together, $\bar\theta$ and Σ describe the location, scale and orientation of the posterior. We then run Phase 2 in the whitened coordinates

$$z = L^{-1}(\theta - \bar\theta), \qquad \theta = \bar\theta + Lz, \tag{14}$$

in which the Phase-1 sample cloud has approximately zero mean and unit covariance. The energy is still evaluated in the original coordinates, and only its gradient is transformed:

$$E_r(z) = E_r(\bar{\theta} + Lz), \qquad \nabla_z E_r = L^\top \nabla_\theta E_r, \tag{15}$$

where $E_r(\theta) = -\log p(\theta) - \log r_{\psi_r}(\theta, x_o)$ is the round-r energy of (8).

Because the map is an affine change of variables with constant Jacobian, it leaves the target posterior unchanged: samples drawn in z are mapped back by $\theta = \bar{\theta} + Lz$, and all simulations, NRE training and C2ST evaluation take place in θ. The transform requires no additional simulator calls, since it is computed only from Phase-1 samples. It is fixed after round 0, so the Phase 2 networks, which are warm-started from one round to the next, remain valid across rounds. The construction is the analogue of mass-matrix adaptation in HMC [39, 40, 47], and of the affine invariance of ensemble samplers [52], with the metric supplied by Phase 1 rather than by a separate warm-up or tuning run. One consequence is that the reference noise scale σ_t is specified in z-units, that is, relative to the spread of the Phase 1 proposal rather than in the units of θ.

Preconditioning is most useful for unimodal posteriors with an ill-conditioned energy landscape, such as strongly correlated parameters, long narrow ridges, or widths that differ by orders of magnitude between directions. For Lotka–Volterra, for instance, the posterior is about $40\times$ narrower across its ridge than along it. There, a single affine map removes most of the anisotropy, after which the isotropic reference process is appropriate. For multimodal posteriors a global affine map is of little use, the mean and covariance of a multimodal Phase 1 proposal span all modes and match the local geometry of none of them. Handling such targets would require a position-dependent metric [48] or a nonlinear transport map [49]; here we leave the sampler unpreconditioned. Developing such geometry-adapted reference processes for diffusion samplers, including learned transport maps and position-dependent metrics built from the Jacobian of the differentiable simulator, is the subject of our future work. For posteriors that are already well conditioned, preconditioning reduces to a near-identity rescaling and has little effect. In our experiments we therefore precondition Phase 2 for Lotka–Volterra and `gw_ringdown`, and not for the multimodal Two Moons and SLCP posteriors.

Important to note is that preconditioning does not give RSBI an advantage over the baselines, since SNPE and SNVI represent the posterior with normalising flows, whose learned bijections adapt to the posterior's scale and orientation by construction. Preconditioning simply gives the Phase 2 sampler the same affine information, estimated from RSBI's Phase 1 samples without additional simulator calls.

B Additional Experiments

B.1 Compute-Matched Performance

For all three simulators we time 20 forward passes and 20 forward + backward passes using a batch size of 32 and 5 warmup batches. Gradients are calculated with regards to the sum of the outputs of the simulator (e.g., `x.sum()`). The timings are done on an NVIDIA GTX 1660 Ti laptop GPU. The per-batch timings are reported in Table 2, C2ST under a compute-matched budget is reported in Figure 5, and the number of simulator calls used is reported in Table 3. Across all simulators we see an increased cost when doing the additional backward pass, ranging from $\sim 1.7\times$ to $\sim 4.0\times$. We note that although the computational costs of doing the backward pass are significant, RSBI only requires this extra compute during Phase 1, which can take up only a small portion of the total simulation budget. We find across all benchmarks that RSBI maintains a strong performance across all simulation budgets even when compute-constrained. Especially in Two Moons and Bernoulli GLM, RSBI maintains it's strong C2ST scores. We stress that the measured timings in Table 2 are hardware-dependent.

Table 2: Effect of backwards pass on per-batch mean compute time across 3 `sbibm` benchmarks.

Simulator	**Forward (ms)**	**Forward+Backward (ms)**	**Ratio**
Two Moons	0.1401	0.3528	2.52×
SLCP	0.2113	0.8524	4.03×
Bernoulli GLM	0.8057	1.3854	1.72×

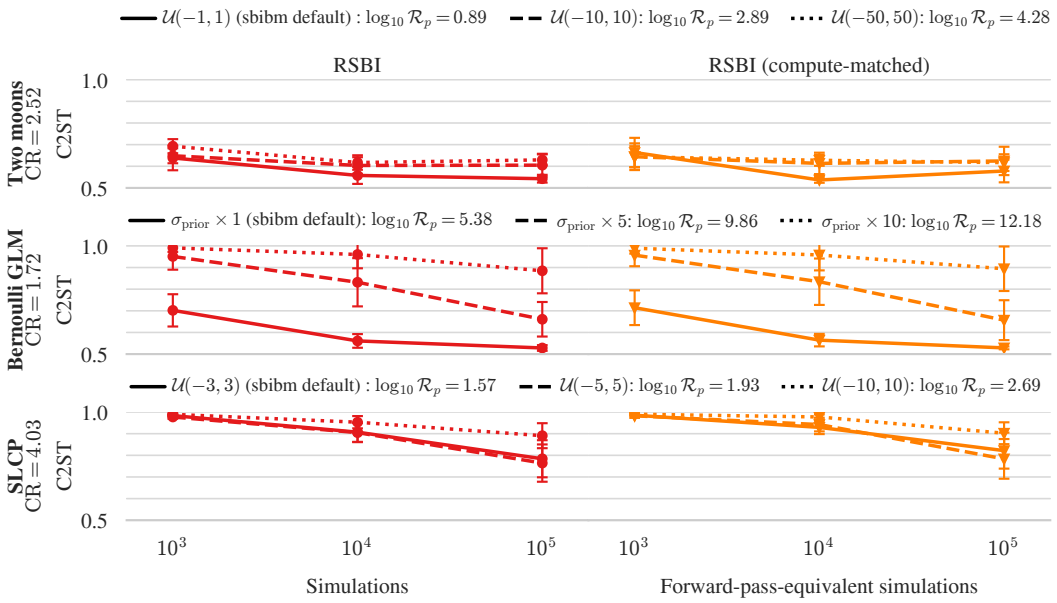

Figure 5: **Left:** C2ST of RSBI against number of simulations. **Right:** C2ST of RSBI against the equivalent compute-matched simulation budget (reported in Table 3). Calculated for Two Moons, Bernoulli GLM and SLCP benchmarks. Y-axis labels include the compute ratio CR charged per Phase-1 simulation (Table 2). Lines are the prior scenarios of Figure 4 with error bars being the standard deviation over 10 observations.

Table 3: Total simulator calls made by the compute-matched runs of Figure 5.

Task	Prior	$N_{\text{sim}} = 10^3$	$N_{\text{sim}} = 10^4$	$N_{\text{sim}} = 10^5$
Two Moons	default	724	5 064	70 037
	wide10x	800	9 203	99 203
	wide50x	800	9 203	99 203
Bernoulli GLM	default	820	8 200	83 455
	medium	820	8 200	82 015
	large	820	8 200	82 015
SLCP	default	624	6 364	59 859
	wide5x	624	6 364	59 859
	wide10x	624	6 364	59 859

B.2 Posterior mass leakage mitigation

Leakage arises when an algorithm learns a density over θ that is not tied to the prior support. Atomic SNPE-C fits a conditional density estimator $q_\psi(\theta \mid x)$, an unbounded flow or Gaussian mixture on $\mathbb{R}^d$, to simulated pairs, and its proposal correction normalises the estimator only over the finite set of contrasting atoms. Nothing in the loss evaluates the estimator outside the support, so when the posterior modes sit at the boundary, the smooth estimator spills mass across it and the leaked mass must be removed afterwards by rejection. Truncated SNPE (TSNPE; 9) avoids the atomic correction by drawing simulations from the prior truncated to a high-probability region and training with the standard NPE loss, which removes the optimisation pathology behind leakage; its estimator is nevertheless still a free density on $\mathbb{R}^d$, so any residual out-of-support mass is again removed by rejection. RSBI never learns such a density. Its Phase 2 sampler is trained against the energy $E_r(\theta) = -\log p(\theta) - \log r_{\psi_r}(\theta, x_o)$, which contains the prior exactly. Outside the support $\log p(\theta) = -\infty$, so the target assigns zero mass there whatever the ratio estimator outputs: an unconstrained classifier can be wrong about r_{ψ_r} off the support, but the product $p(\theta)\, r_{\psi_r}(\theta, x_o)$ cannot be. Because the sampler is fitted to this target, mass placed outside the support is penalised by the energy, and its gradient pushes trajectories back inside. In practice we implement the box

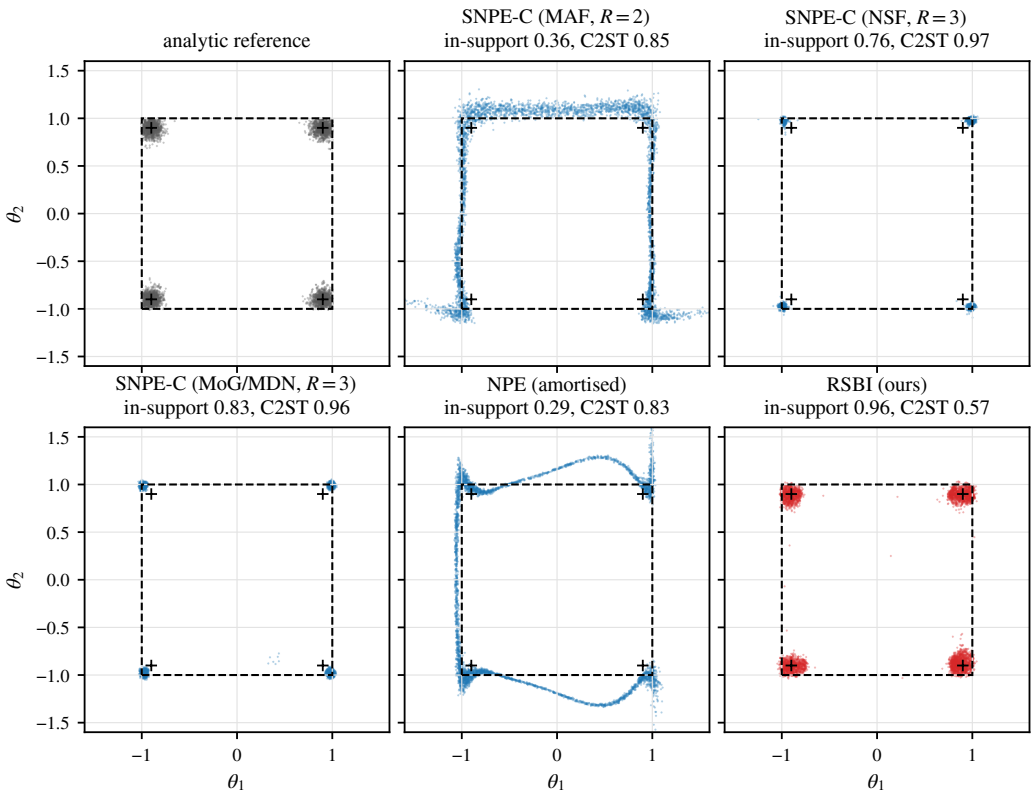

Figure 6: Demonstration of posterior leakage and it's mitigation with RSBI: Posterior samples on the tractable squared-location task ($d = 2$, $N_{\text{sim}} = 10^4$, one observation). Dashed square: the support of the uniform prior, $[-b, b]^2$; crosses: the four true modes at $(\pm c, \pm c)$, which lie close to the corners of the support. Each panel shows the raw sampler output of one method (before rejection for the flows, before removing the metadynamics bias for RSBI), with the fraction of its mass inside the support. The C2ST against the analytic reference (0.5 = indistinguishable) is computed on each method's final posterior restricted to the support: the in-support samples of the flows and NPE, and the in-support reweighted samples of RSBI. The atomic SNPE-C flows and amortised NPE place much of their mass outside the support; RSBI keeps it inside while covering all four modes, like the analytic reference (top left).

prior as a stiff quadratic wall rather than an infinite one, to keep gradients finite, so a small residual fraction of the raw samples falls just outside the boundary (Figure 6). This fraction is set by the wall stiffness and the sampler's training error rather than by the estimator, and it vanishes from the final posterior, which is restricted to the support. The same argument applies to SNVI [7], whose variational posterior is likewise fitted to the prior-weighted ratio $p(\theta)\, r_\psi(\theta, x_o)$ rather than learned as a free density, and whose final importance-resampling step assigns zero weight to samples outside the support. This guarantee holds only to the extent that the ASBS sampler fits its target. When the controller fails to converge, for instance by collapsing onto a subset of modes or scattering trajectories across the box, its samples need not respect the energy, and out-of-support mass can reappear. This may not be in general true for other samplers/models used in Phase 1 of RSBI, which we discuss as a limitation of choosing WT-ASBS in Section 6.

To isolate how bounded prior support affects inference (leakage), we use a tractable benchmark with a closed-form posterior. The parameters are drawn from a uniform box, $\theta \sim \mathcal{U}([-b, b]^d)$, and the data are $x = \theta^2 + \sigma\epsilon$, $\epsilon \sim \mathcal{N}(0, I)$, so the posterior for $x_o = (c^2, \ldots, c^2)$ has 2^d symmetric modes at $(\pm c, \ldots, \pm c)$. Taking c close to b places every mode near a corner of the prior support (in $d = 2$, a square), the regime in which density-based posteriors are known to "leak" probability mass outside the support. Because the reference posterior is available analytically, we can measure leakage exactly. We use $d = 2$, $b = 1$, $c = 0.9$ and $\sigma = 0.1$.

At a fixed simulation budget ($N_{\text{sim}} = 10^4$) we compare atomic SNPE-C [3], using the standard `sbi` implementation [53], with masked autoregressive flow [54] and neural spline flow [55] density estimators, the non-atomic mixture-of-Gaussians variant (whose posterior estimator is a Gaussian mixture), and amortised NPE (a single round with a masked autoregressive flow), against RSBI. We report, on the raw sampler output, the fraction of mass inside the prior support and the fraction of the 2^d modes recovered (mode recall), and the classifier two-sample test (C2ST) against the analytic reference on each method's final posterior restricted to the support.

For RSBI, Phase 1 uses the Gaussian negative log-likelihood surrogate with the exact noise variance σ^2 and reference noise scale $\sigma_{\max} = 0.3$, and is stopped after 5×10^3 simulations. Phase 2 spends the remaining 5×10^3 simulations in a single round: a contrastive NRE is trained on fresh draws from the Phase 1 sampler, its ratio is evaluated only within the truncation region of Appendix A.4, and the Phase 2 sampler, again with $\sigma_{\max} = 0.3$, targets the posterior under the uniform box prior.

The flow-based amortised and sequential methods place a substantial fraction of their mass outside the prior support, and the mixture-of-Gaussians variant mitigates but does not eliminate this (Figure 6). In contrast, RSBI samples under a constrained energy and keeps almost all of its mass inside the support while recovering all four modes. This controlled experiment shows that the leakage pathology is structural to unconstrained density estimation on boundary-mode targets, and that the constrained Phase 1 and 2 samplers remove it by construction.

B.3 Ablation: replacing Phase 2 by standard sequential methods

RSBI has two components that could each explain its advantage on wide priors. Phase 1 localises the posterior by training ASBS on a surrogate, so it zooms from the prior onto the region that carries posterior mass. Phase 2 then refines this estimate with an NRE and an ASBS sampler that transports the Phase 1 sampler locally. To separate the two, we keep Phase 1 fixed and replace Phase 2 with established sequential methods that use Phase 1 as their first proposal, at the same simulation budget. The obvious candidate, atomic SNPE-C [3], is poorly suited to this setting. Its contrastive loss normalises the posterior only over a few atoms drawn from the proposal, so posterior mass leaks into the large region of the prior that the localised proposal never visits. With the Phase 1 sampler as its first proposal it reaches a C2ST of only 0.96 and 0.93 at 10^3 and 10^4 simulations, far from RSBI. We therefore use truncated SNPE (TSNPE) [9], which restricts the prior to a highest-density region (HPR) of the current posterior estimate and trains with the maximum-likelihood loss. It avoids leakage by construction and is the natural way to hand a localised proposal to a neural posterior estimator. As a second, variational baseline we use SNVI [7], which, like Phase 2 of RSBI, fits a sampler to the prior times a learned ratio and draws the next round of simulations from it. Unlike RSBI, it fits a normalising flow over the whole prior by forward KL, so the flow is free to move mass anywhere in the prior box, including regions where the ratio estimator has never seen data. On a wide prior the classifier extrapolates there without control. In SNVI's self-normalised importance weights a single proposal in such a region can dominate all others, so the flow is pulled onto spurious high-ratio regions even when initialised at the Phase 1 sampler, and the next round of simulations is spent there. RSBI's Phase-2 instead, transports particles locally from the Phase-1 proposal due to ASBS using Phase 1's $\tilde{p}_\epsilon\left(\theta \mid x_o\right)$ as the source distribution.

We use Two Moons with the widest prior, $[-50, 50]^2$ ($\log_{10} \mathcal{R}_p \approx 4.3$), at $N_{\text{sim}} \in \{10^3, 10^4, 10^5\}$ and observations 1–10. For each cell we reload the trained Phase 1 sampler of the corresponding RSBI run, without retraining. TSNPE then runs two rounds with the same per-round allocation as RSBI's ratio estimator, $600/1\,000/5\,000$ simulations in round 1 and $400/9\,000/95\,000$ in round 2. Round 1 draws from the prior truncated to the $1 - \epsilon$ HPR of a 10-component Gaussian mixture fitted to the Phase 1 samples, and round 2 from the prior truncated to the HPR of the round-1 posterior at x_o, with $\epsilon = 10^{-4}$. TSNPE cannot be trained on the simulations RSBI's ratio estimator uses. Maximum-likelihood NPE trained on draws from a proposal $\tilde{p}(\theta)$ targets $\tilde{p}(\theta \mid x) \propto \tilde{p}(\theta)\, p(x \mid \theta)$, which equals the posterior inside the truncation region only when $\tilde{p}$ is the truncated prior. The ratio estimator instead learns $p(x \mid \theta)/q(x)$ (Eqn. 8) for any proposal whose support covers the posterior, so RSBI can train on the Phase 1 pairs and on draws from its Phase 2 sampler directly. Using the same pairs for TSNPE would require atomic SNPE-C, which suffers from leakage on wide priors, or importance weights $p(\theta)/\tilde{p}(\theta)$, which need a sampler density that ASBS does not provide. We therefore use the Phase 1 samples only to define the first truncation region and simulate afresh in both rounds, so TSNPE uses n_1 more simulations than RSBI, which favours TSNPE. This flexibility

Table 4: Phase-2 ablation on Two Moons with the widest prior ($[-50, 50]^2$): mean (sd) C2ST over ten observations at matched simulation budget. Phase 1 + TSNPE uses the trained Phase 1 sampler of each RSBI run as the first truncation of TSNPE [9], with the same per-round simulation allocation as RSBI's ratio estimator (n_1 more simulations, since the Phase 1 pairs are not truncated-prior draws). Phase 1 + SNVI replaces RSBI's NRE and ASBS sampler by SNVI [7] (SNRE-B and a normalising flow trained by forward KL), trained on exactly the same simulations in round 1 and the same number in round 2. SNPE and SNVI+fKL from the prior are the main-benchmark configurations. Phase 1 alone (524 simulations) reaches 0.995 (0.001).

Method	$N_{\text{sim}} = 10^3$	10^4	10^5
SNPE (prior, R=2)	0.997 (0.002)	0.991 (0.015)	0.985 (0.009)
SNVI+fKL (prior, R=5)	0.987 (0.004)	0.943 (0.023)	0.993 (0.020)
Phase 1 + TSNPE (R=2)	0.925 (0.032)	0.873 (0.083)	0.738 (0.107)
Phase 1 + SNVI (R=2)	0.938 (0.061)	0.770 (0.101)	0.848 (0.023)
RSBI (Phase 1 + NRE/ASBS)	**0.693** (**0.033**)	**0.617** (**0.034**)	**0.629** (**0.026**)

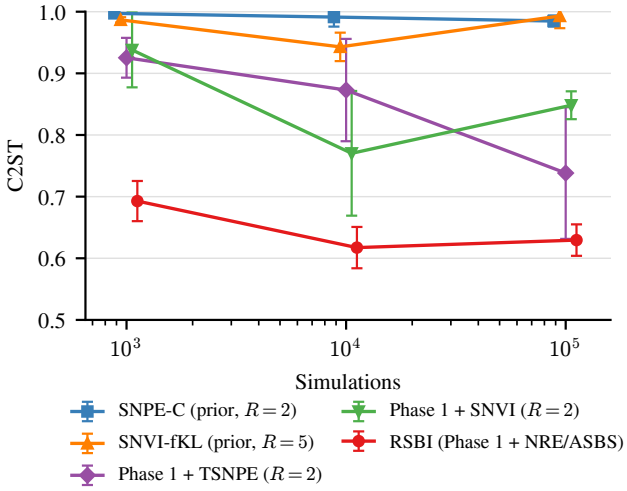

Figure 7: C2ST against the reference posterior on Two Moons with the widest prior (mean $\pm$ sd over ten observations, lower is better). Values as in Table 4.

in the choice of proposal is a structural advantage of the ratio-based Phase 2. The density estimator (NSF, 50 hidden units, independent z-scoring) and the C2ST protocol are those of the main SNPE baseline. In every cell both truncation regions contain all reference samples, so the truncation never removes posterior mass.

Phase 1 + SNVI uses the configuration of our SNVI+fKL baseline (SNRE-B with a ResNet classifier, a neural spline flow fitted by forward KL, and sampling-importance resampling) and is matched to RSBI in both simulations and rounds. Its ratio estimator is trained in round 1 on the same $600/1\,000/5\,000$ Phase 1 pairs as RSBI's, and round 2 simulates the remaining budget from the round-1 variational posterior. Like the ASBS sampler of RSBI, the flow starts at the Phase 1 sampler, fitted to its samples by maximum likelihood. We compute the importance weights of the forward KL in log space, which avoids a numerical overflow of the released implementation that is detailed in Appendix E.

Table 4 and Figure 7 show that both phases are needed. On its own, Phase 1 is a poor posterior estimate (C2ST 0.995) because the surrogate is coarse, but it localises the posterior reliably. Using it as the first proposal improves TSNPE by 0.07, 0.12 and 0.25 in C2ST at 10^3, 10^4 and 10^5 simulations. SNPE from the prior barely improves at any budget, so without this zoom a standard sequential method spends almost its whole budget in regions of negligible posterior mass.

The zoom alone is not sufficient, however. Given the same Phase 1 and the same per-round budget, replacing TSNPE by RSBI's variational Phase 2 lowers the C2ST by a further 0.23, 0.26 and 0.11,

and RSBI is better than Phase 1 + TSNPE on 10, 10 and 7 of the ten observations. Phase 2 accounts for most of the gain at 10^3 and 10^4 simulations, and RSBI reaches at 10^3 simulations (0.693) an accuracy that Phase 1 + TSNPE does not reach at 10^5 (0.738). Two properties of the variational Phase 2 explain this. First, the ratio estimator is valid under any proposal, so every simulation is used, including the Phase 1 pairs, whereas TSNPE must discard them and draw prior-shaped samples from the truncation region. Second, ASBS samples directly from the current posterior estimate, the prior times the learned ratio, so the round-2 simulations concentrate where the posterior has mass. TSNPE can only spread them over the prior restricted to an HPR of its round-1 estimate, and after a coarse zoom on the wide prior this region is still large compared with the posterior. At 10^5 simulations TSNPE closes part of the gap, as expected from a consistent estimator trained on a well-localised proposal, but it remains less accurate and much less stable across observations (sd 0.107 against 0.026). Phase 1 + SNVI confirms that substantial gain comes from how Phase 2 refines the posterior and not from the ratio estimator alone. With the same Phase 1 pairs and the same rounds as RSBI, it improves on SNVI+fKL from the prior at every budget, but RSBI is better on all ten observations at every budget. At 10^3 simulations the variational posterior moved onto a spurious high-ratio region at the boundary of the prior in 4 of the 10 observations (C2ST 1.000), the failure described above. Both SNVI variants also share a non-monotonic dependence on the budget, with C2ST rising from 0.943 to 0.993 (prior, R=5) and from 0.770 to 0.848 (Phase 1, R=2) between 10^4 and 10^5 simulations. The two use different implementations, numbers of rounds and initialisations, so this degradation reflects SNVI's variational refinement on a wide prior. RSBI improves monotonically with the budget up to 10^4 simulations and remains stable at 10^5. The advantage of RSBI on wide priors in this ablation study is therefore derivative of both the Phase 1 zoom and the variational ASBS refinement that follows it.

C Surrogate Likelihoods: Details and ablations

This appendix expands on the surrogate likelihood used in Phase 1 (Section 4.1). We stress at the outset that the choice of distance, and hence of surrogate likelihood, is task dependent and deserves care from the analyst. The ABC literature offers a wide range of choices, each motivated by a particular class of problems. These include hand-crafted or learned summary statistics [56], Gaussian synthetic likelihoods [57, 58], and full-data discrepancies such as the Wasserstein distance [59], the maximum mean discrepancy [60], the Kullback-Leibler divergence [61] and the energy statistic [62]. Comparative studies find that no single choice is best across problems [63], and the ABC kernel itself can be read as a model of the error between simulator and data [64], so the bandwidth ϵ is a modelling choice as much as a tuning parameter.

We first describe the surrogate likelihoods used in this work and then discuss the bandwidth and metric trade-offs known from the ABC literature. Next, we show how inspecting the energy and gradient landscape of a surrogate can inform the choice of the bandwidth ϵ. We close with ablations over the three surrogate likelihoods and over ϵ. Figure 8 shows the Phase-1 posteriors obtained with the surrogates used for the main results. Phase 1 already concentrates on the posterior support and recovers its modes before any NRE is trained.

C.1 Gaussian negative log-likelihood with adaptive variance

For a simulator with output $x = s(\theta, \omega) \in \mathbb{R}^{d_x}$, we model per-dimension residuals as zero-mean Gaussian with diagonal covariance $\mathrm{diag}\big(\epsilon_1(\theta), \dots, \epsilon_{d_x}(\theta)\big)$, which plays the role of the bandwidth ϵ. The surrogate distance is

$$\mathcal{D}_\epsilon^{\mathrm{NLL}}\big(s(\theta), x_o\big) \;=\; \tfrac{1}{2}\sum_{d=1}^{d_x} \frac{\big(x_o^{(d)} - \bar{x}^{(d)}(\theta)\big)^2}{\epsilon_d(\theta)} \;+\; \tfrac{1}{2}\sum_{d=1}^{d_x} \log \epsilon_d(\theta), \tag{16}$$

where, as in Sec. 4.1, $s(\theta)$ denotes the K replicates $x_k = s(\theta, \omega_k)$, $\omega_k \overset{\text{iid}}{\sim} p(\omega)$, sharing the parameter θ but differing in the simulator's internal randomness, and $\bar{x}^{(d)}(\theta)$ and $\epsilon_d(\theta)$ are their per-dimension mean and unbiased variance. Heteroscedastic Gaussian losses of this form have a long history in neural regression [65] and are now standard for deep predictive uncertainty quantification [66]. We use PyTorch's `gaussian_nll_loss` with `full=False`, which omits the constant $\frac{d_x}{2}\log 2\pi$ (irrelevant for gradients by Theorem 4.1). The variance term in (16) acts as an entropy-like penalty that prevents the bridge from concentrating in regions where the simulator is highly stochastic.

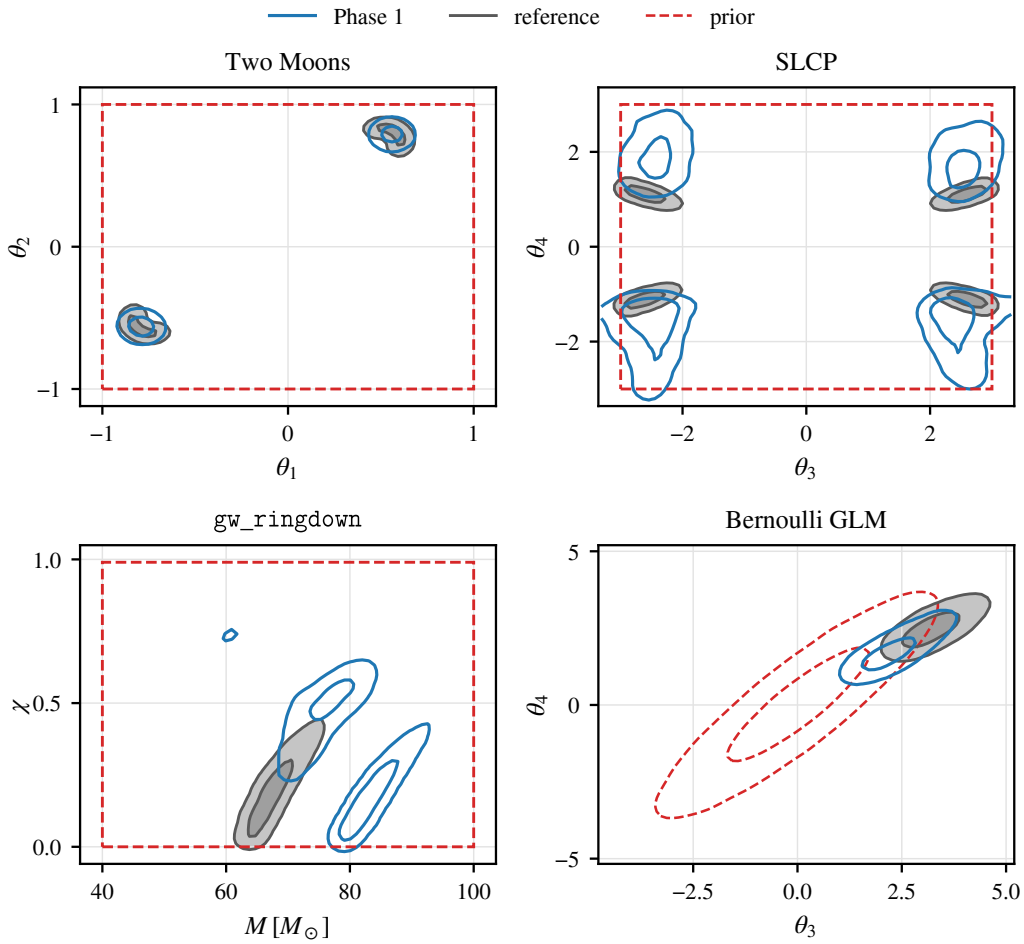

Figure 8: Phase-1 posteriors used for the main results in Fig. 4 for Two Moons, SLCP, `gw_ringdown` and the Bernoulli GLM (top left to bottom right), each at its default prior with $N_{\text{sim}} = 10^5$ at observation 1 (see Sec. G for the Phase-1 configuration). We show the 1- and 2-σ contours of the Phase-1 (blue) and reference (grey) posteriors over the prior (dashed red: the prior box for the uniform priors, and the 1- and 2-σ contours of the Gaussian prior for the Bernoulli GLM). For SLCP and `gw_ringdown` we show the 2-D marginals of the full 5-D and 6-D posteriors for the two parameters with the most pronounced structure. These are (θ_3, θ_4), which separate the four SLCP modes, and the mass and spin pair (M, χ), whose posterior is a single curved ridge (Phase 1 also places small islands of mass off the ridge). For the unimodal Bernoulli GLM we show the most strongly correlated pair of its 10 parameters.

We support two options for the variance $\epsilon_d(\theta)$. **(i) Fixed:** $\epsilon_d(\theta) = \epsilon_d^\star$ for a user-supplied constant. A single simulator call per θ suffices ($K = 1$), appropriate when the noise model is exact. **(ii) Adaptive:** $\epsilon_d(\theta)$ is recomputed at every θ from the K replicates. It varies smoothly with θ but is noisy at small K.

With fixed variance, (16) is a weighted squared distance plus the constant $\frac{1}{2}\sum_d \log \epsilon_d^\star$, hence a sufficient distance measure (Definition A.1) up to a constant, covered by Theorem 4.1. In the adaptive case, $\epsilon_d = \epsilon_d(\theta, \omega_{1:K})$ is the sample variance of the replicates drawn at θ, recomputed at every evaluation and differentiated through. The term $\frac{1}{2}\sum_d \log \epsilon_d$ is then not a constant, since its expectation varies with θ wherever the simulator noise is heteroscedastic, so (16) is not a PSD distance up to a constant. It nonetheless defines a proper surrogate: the variance is floored, so $\mathcal{D}_\epsilon^{\text{NLL}}$ is bounded below, $\exp(-\mathbb{E}\,\mathcal{D}_\epsilon^{\text{NLL}})\,p(\theta)$ is integrable for any proper prior as per Def.A.1, and (7)

remains the gradient of a well-defined energy that ASBS targets as before. Any mismatch with the true likelihood is removed in Phase 2.

C.2 Energy distance

We also consider the energy statistic of Székely and Rizzo [67], a characteristic-kernel discrepancy known to be equivalent to MMD with a distance-induced kernel [38, 68], whose sample estimator differentiates only through the (smooth) simulator output and never through a non-differentiable comparison. We instantiate it as $\mathcal{D}_\epsilon$ by identifying P with the simulator distribution $p(x \mid \theta)$ and Q with the (point-mass) data distribution at x_o. For two distributions P, Q on $\mathbb{R}^{d_x}$ and the Euclidean kernel $k(u, v) = -\|u - v\|_2$, the squared energy distance is

$$\mathcal{E}^2(P, Q) \;=\; 2\,\mathbb{E}_{X\sim P,\,Y\sim Q}\|X - Y\| \;-\; \mathbb{E}_{X,X'\sim P}\|X - X'\| \;-\; \mathbb{E}_{Y,Y'\sim Q}\|Y - Y'\|, \tag{17}$$

with $\mathcal{E}^2(P, Q) = 0$ iff $P = Q$ [67]. The bracket in Eq. (18) is its minimum-variance unbiased U-statistic. For each θ in the training minibatch we draw K i.i.d. simulator replicates $\{x_k\}_{k=1}^K = \{s(\theta;\, \omega_k)\}_{k=1}^K$ as samples from P, and treat the N available observations $\{y_n\}_{n=1}^N$ as samples from Q (in our setting $N = 1$ and $y_1 = x_o$). The unbiased U-statistic estimator, divided by a length scale $\epsilon > 0$, is then

$$\begin{aligned}\mathcal{D}_\epsilon^{\mathrm{ED}}\big(s(\theta), x_o\big) = \frac{1}{\epsilon}\Bigg[\frac{2}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}\|x_k - y_n\| \;-\; \frac{1}{K(K-1)}\sum_{k\neq k'}\|x_k - x_{k'}\| \\ -\; \frac{1}{N(N-1)}\sum_{n\neq n'}\|y_n - y_{n'}\|\Bigg],\end{aligned} \tag{18}$$

with the diagonals $k = k'$ and $n = n'$ excluded; the third term vanishes when $N = 1$. For a fixed ϵ the estimator is unbiased, and its expectation over the replicates is exactly the population quantity, $\mathbb{E}\,\mathcal{D}_\epsilon^{\mathrm{ED}}(s(\theta), x_o) = \mathcal{E}^2\big(p(x \mid \theta), \delta_{x_o}\big)/\epsilon$, so the surrogate of Sec. 4.1 is the energy distance itself. For $N = 1$ the estimator is also non-negative, since $\|x_k - x_{k'}\| \le \|x_k - x_o\| + \|x_{k'} - x_o\|$, so it is a sufficient distance measure (Definition A.1). Since $\mathcal{E}$ carries the units of x, $\frac{1}{\epsilon}$ makes $\mathcal{D}_\epsilon^{\mathrm{ED}}$ unit-less. It does not change the minimiser and only weights the discrepancy against the prior. By default ϵ is adaptive. It is the median pairwise distance among simulations at 256 prior draws (counted toward the simulation budget) and updated at every energy evaluation by an exponential moving average (decay 0.95) of the minibatch-averaged median pairwise distance among $\{x_k\}_{k=1}^K \cup \{x_o\}$, with no gradient flowing through ϵ. Because ϵ is a single detached scale, updated with weight 0.05 from a minibatch average, it depends only weakly on the replicates of any one θ. Within each evaluation we treat it as fixed, so unbiasedness holds conditionally on ϵ, while across training ϵ drifts and the Phase-1 target with it. With a single replicate ($K = 1$, used on Two Moons) the second term in (18) is absent and $\mathcal{D}_\epsilon^{\mathrm{ED}}$ reduces to $2\|x_1 - x_o\|/\epsilon$, whose expectation exceeds $\mathcal{E}^2/\epsilon$ by $\mathbb{E}\|X - X'\|/\epsilon$. For Two Moons the simulator noise is additive, so this offset does not depend on θ and, by Theorem 4.1, the gradient is that of the energy distance itself. The median over $\{x_1, x_o\}$ is then simply $\|x_1 - x_o\|$, so ϵ tracks a moving average of the minibatch-mean distance between the simulations and x_o. It starts at the scale of the prior predictive and shrinks as Phase 1 concentrates on the posterior, which anneals the surrogate from smooth to sharp during training (Appendix C.7 compares this with fixed length scales).

C.3 Maximum mean discrepancy (MMD)

We also consider the squared maximum mean discrepancy (MMD) [38] with the Gaussian (RBF) kernel $k_\epsilon(u, v) = \exp\!\big(-\|u - v\|_2^2/(2\epsilon^2)\big)$ of bandwidth $\epsilon > 0$, which is characteristic, so that $\mathrm{MMD}_\epsilon^2(P, Q) = 0$ iff $P = Q$. As for the energy distance, we identify P with the simulator distribution $p(x \mid \theta)$ and Q with the (point-mass) data distribution at x_o, and

$$\mathrm{MMD}_\epsilon^2(P, Q) \;=\; \mathbb{E}_{X,X'\sim P}\,k_\epsilon(X, X') \;+\; \mathbb{E}_{Y,Y'\sim Q}\,k_\epsilon(Y, Y') \;-\; 2\,\mathbb{E}_{X\sim P,\,Y\sim Q}\,k_\epsilon(X, Y). \tag{19}$$

For each θ in the training minibatch we draw K i.i.d. simulator replicates $\{x_k\}_{k=1}^K = \{s(\theta;\, \omega_k)\}_{k=1}^K$ and treat the N observations $\{y_n\}_{n=1}^N$ as samples from Q (here $N = 1$, $y_1 = x_o$). Since the unbiased

U-statistic of the Q–Q term is undefined for $N = 1$, we use the plug-in (V-statistic) estimator

$$\mathcal{D}_\epsilon^{\mathrm{MMD}}\big(s(\theta), x_o\big) = \frac{1}{K^2}\sum_{k=1}^{K}\sum_{k'=1}^{K} k_\epsilon(x_k, x_{k'}) \;+\; \frac{1}{N^2}\sum_{n=1}^{N}\sum_{n'=1}^{N} k_\epsilon(y_n, y_{n'}) \\ -\; \frac{2}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N} k_\epsilon(x_k, y_n), \tag{20}$$

diagonals included, so that for $N = 1$ the second term equals 1. As $\mathcal{D}_\epsilon^{\mathrm{MMD}} \in [0, 2]$ is unitless, no length-scale normalisation is applied. The bandwidth ϵ is either fixed or, by default, set by the median heuristic separately for each θ: $\epsilon(\theta)$ is the median pairwise Euclidean distance among $\{x_k\}_{k=1}^K \cup \{x_o\}$, recomputed at every energy evaluation from the current replicates (no pilot simulations, no smoothing), and gradients are propagated through $\epsilon(\theta)$. The V-statistic lies in $[0, 2]$, so it is a sufficient distance measure (Definition A.1), but unlike the energy distance its expectation is biased, $\mathbb{E}\,\mathcal{D}_\epsilon^{\mathrm{MMD}} = \mathrm{MMD}_\epsilon^2 + \frac{1}{K}\big(1 - \mathbb{E}\,k_\epsilon(X, X')\big)$ for fixed ϵ, an additional penalty on simulator spread that vanishes as $K \to \infty$. With the median heuristic, $\epsilon(\theta)$ depends on the replicates and is differentiated through, as for the adaptive variance of Appendix C.1. MMD is used only in the surrogate ablations (Appendices C.6 and C.7).

C.4 Bandwidth and Metric Trade-offs

The impact of $\tilde{\mathcal{D}}_\epsilon$ and ϵ on $\tilde{p}_\epsilon(x_o \mid \theta)$ has been studied extensively in the ABC literature [21, 32, 56, 60]. Under mild regularity, $\tilde{p}_\epsilon \to p$ as $\epsilon \to 0$, but small ϵ degrades acceptance and inflates Monte Carlo variance, while large ϵ over-smooths the likelihood and biases the posterior [69]. The choice of $\tilde{\mathcal{D}}_\epsilon$ involves similar considerations, and kernel-based discrepancies such as MMD and the energy distance can improve the bias-variance trade-off relative to a fixed summary statistic.

In RSBI the trade-off takes a different form, because Phase 1 samples the surrogate with gradients rather than by acceptance. A small ϵ gives a sharp energy whose gradient is dominated by Monte Carlo noise near x_o and vanishes far from it. A large ϵ flattens the data term until the drift is set by the prior alone. Since Phase 2 NRE-ASBS refinement targets the true posterior, the surrogate need not approximate $p(\theta \mid x_o)$ precisely. It only needs to place mass in the region that Phase 2 then refines, so larger ϵ values can be tolerated than in standalone ABC. The surrogates we use are not guaranteed to reproduce the mode, shape or variance of the true likelihood, and Phase 2 corrects the resulting bias.

Our ablations show that both choices depend on the simulator and the data, and that no single surrogate or bandwidth rule is best across tasks. Among surrogates (Appendix C.6), the energy distance gives the best Phase-1 proposal on Two Moons and the Bernoulli GLM, where MMD fails to concentrate at all. On SLCP all three surrogates are within one standard deviation of each other in C2ST, and MMD places much of its mass on the edges of the prior box. Whether the bandwidth should be adaptive or fixed is equally task dependent (Appendix C.7). On SLCP an adaptive per-θ variance is needed for the Gaussian NLL, because θ_3, θ_4 and θ_5 enter the simulator only through the noise covariance, and no fixed variance discovers the modes better than prior samples do. On the Bernoulli GLM the opposite holds. Per-θ variances estimated from a few binary replicates are too noisy, adaptive bandwidths fail (C2ST ≥ 0.995) and a fixed bandwidth is required. Sensitivity to ϵ also varies. Misspecifying the best bandwidth by an order of magnitude costs 0.13 to 0.18 in Phase-1 C2ST on Two Moons and raises it to 0.98 or more on the Bernoulli GLM, whereas on SLCP the C2ST changes by less than 0.03 over four orders of magnitude. The pilot landscape heuristic of Sec. C.5 chooses bandwidths without the reference posterior that lie within half a decade of the best value on Two Moons and coincide with it on the Bernoulli GLM. It remains a heuristic, however, and it does not decide between surrogates or between adaptive and fixed bandwidths. Making these choices in a principled way requires a better understanding of the gradient landscape that a differentiable simulator and a distance induce over the prior, which we leave to future work.

C.5 Bandwidth and temperature hyperparameter guesses via energy and gradient landscape

We suggest a cheap heuristic to aid in choosing the surrogate bandwidth ϵ and the Phase-1 temperature T. This method leverages a cheap pilot run of $\mathcal{O}(10^2)$ draws from the prior in order to observe the landscape of gradients and energy throughout the prior volume. The assumption is that bandwidth ϵ and temperature T configurations that provide informative gradients for the Phase-1 zoom should

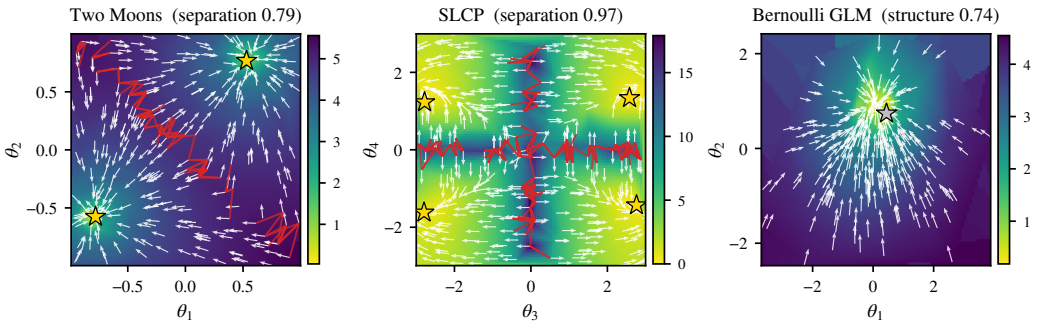

Figure 9: Pilot energy landscapes for Two Moons (left), SLCP (centre) and the Bernoulli GLM (right), computed with the Phase-1 surrogate, bandwidth and temperature used for the benchmark main results at the default prior on observation 1 (see Sec. G for Phase-1 hyperparamters and surrogates used). Colour is $\log(1 + E/T - \min E/T)$; white arrows show the direction of $-\nabla_\theta E$; stars are basin sinks (gold: significant); red edges are ridges. SLCP and the Bernoulli GLM are shown on 2-D slices through (θ_3, θ_4) and (θ_1, θ_2). The number in each panel title is the score used to rank bandwidths for that posterior: separation for the multimodal Two Moons and SLCP, structure for the unimodal Bernoulli GLM. Scores for these settings: Two Moons: separation 0.79, structure 0.79; SLCP: separation 0.97, structure 0.21; Bernoulli GLM: separation 0.00, structure 0.74.

produce a structured landscape with gradients that cohesively point to one or more energy wells, no matter where they are within the prior. We detail this process below, noting that exploring a fully fledged hyperparameter optimization regime for WT-ASBS is generally interesting and could constitute future work.

Before any training, we evaluate the surrogate energy $E(\theta) = \mathcal{D}_\epsilon\big(s(\theta), x_o\big) - \log p(\theta)$ on a small pilot batch of prior draws with a few simulator replicates $\omega_{1:K}$ each, look at its landscape, and score candidate bandwidths on the same simulations. The heuristic needs no training and no reference posterior, and its only cost is the pilot simulations.

At each pilot draw we record the energy and the drift $-\nabla_\theta E$, obtained by back-propagating through the differentiable simulator. Splitting the replicates into two halves gives two independent estimates E_1, E_2, and the Monte Carlo noise of the energy is $\sigma_E = \frac{1}{2}\,\mathrm{median}\,|E_1 - E_2|$. All quantities below are computed on the k-nearest-neighbour graph of the pilot draws, so every candidate ϵ, T is evaluated on the same simulations without new simulator calls. In more than two dimensions a projection of the draws mixes points that differ in the hidden parameters, so the maps are drawn on 2-D slices of two nominal parameters only.

Each draw steps to the neighbour that is lower in energy and best aligned with its drift. Following these steps downhill ends in a sink, and the draws that end in the same sink form one basin, or well; its share of the draws is the share of prior mass it attracts. Two neighbouring draws in different basins whose drifts point away from each other, each drift having a negative projection onto the direction to the other draw, lie on a *ridge*. The barrier ΔE of a basin is the lowest energy on its boundary (the higher endpoint of a cross-basin edge) minus the basin's minimum. Adjacent basins are merged unless the barrier between them is at least $3\sigma_E$ deep and at least 30% of their boundary pairs diverge, so that only boundaries where the drift genuinely splits between two wells remain.

We introduce three metrics that can be used to score a given surrogate bandwidth ϵ. First, the *cohesion*: the median cosine between a draw's drift and the direction to its basin's sink, which measures whether the drift points into its own well. Second, the *ridge* fraction: the share of cross-basin neighbour pairs whose drifts diverge, which measures whether the boundaries between wells are genuine ridges. Third, the *depth*: the mean over basins of $\min(1, \Delta E/10\sigma_E)$, which measures whether the barriers stand above the Monte Carlo noise. For a single well the ridge term is 1 and the depth uses the energy contrast $(E_{95} - E_{\min})/\sigma_E$ in place of a barrier. They combine into

$$\text{structure} = \text{cohesion} \times \text{ridge} \times \text{depth}, \qquad \text{separation} = \begin{cases} \text{ridge} \times \text{depth}, & \geq 2 \text{ wells}, \\ 0, & \text{one well}. \end{cases} \tag{21}$$

Task	wells	cohesion	ridge	depth	structure	separation
Two Moons	2	1.00	0.79	1.00	0.79	0.79
SLCP	4	0.22	0.97	1.00	0.21	0.97
Bernoulli GLM	1	0.74	–	1.00	0.74	0.00

Table 5: Landscape scores of the pilot batches in Fig. 9. *Wells*: significant basins. *Cohesion*: median cosine between a draw's drift and the direction to its basin's sink. *Ridge*: share of cross-basin neighbour pairs whose drifts point apart (– for a single well, where the ridge term is 1). *Depth*: mean over basins of $\min(1, \Delta E/10\sigma_E)$, or $\min(1, (E_{95} - E_{\min})/10\sigma_E)$ for a single well. Structure = cohesion × ridge × depth. Separation = ridge × depth for two or more wells and 0 otherwise.

Structure rewards a drift that points coherently into wells that are deep and cleanly separated, whereas separation rewards only the separation of several wells, regardless of the drift inside each. For a multimodal energy landscape we take the ϵ, or the adaptive variant, with the highest separation. For a unimodal energy landscape, we take the highest structure.For certain priors, a large ϵ leaves only the prior's own well, which every score rates as perfect, so only settings whose median data gradient is at least three times the prior gradient are considered.

The landscape can inform the temperature T, but only when WT is on. The target $\exp(-E/T)$ crosses a barrier of height ΔE with a probability that scales as $\exp(-\Delta E/T)$ (Sec. A.1), so what matters is the barrier in units of the temperature, $\Delta E/T$. Well-tempered metadynamics (WT) with bias factor γ fills the wells and lowers the effective barrier to roughly $\Delta E/\gamma$, leaving $\Delta E/(\gamma T)$. Let κ denote the barrier height, in units of T, that we want to remain after the bias. Setting $\Delta E/(\gamma T) = \kappa$, with ΔE the median barrier of the significant wells, gives

$$T = \begin{cases} 1, & \text{one well (no WT),} \\ \dfrac{\Delta E}{\gamma\,\kappa}, & \text{several wells.} \end{cases} \tag{22}$$

The value of κ trades mode separation against mixing. A remaining barrier of $\kappa \approx 3$ gives a crossing factor $e^{-3} \approx 5\%$. Much smaller values flatten the wells and blur the modes, and much larger ones let the sampler stay trapped in the first well it finds. We use $\kappa = 3$ as a heuristic default rather than a tuned value. A single well needs neither tempering nor a bias, so $T = 1$.

Figure 9 shows such landscapes for the Two moons, SLCP and Bernoulli GLM tasks. The colour shows the energy, with the white arrows depicting the drift/gradients from the average pilot. The stars depict the location of the basin sinks and the red edges show ridges such that the wells and barriers between energy basins can be inspected directly. Two Moons shows two wells separated by a ridge along $\theta_1 = -\theta_2$ whilst SLCP shows four wells, one for each sign combination of (θ_3, θ_4), separated by ridges along $\theta_3 = 0$. We note that this energy landscape was only seen for the Gaussian surrogate on SLCP. The Bernoulli GLM shows a single well with the drift pointing into it from everywhere. Table 5 lists the scores of these landscapes.

C.6 Surrogate likelihood ablations

We compare three surrogate distances $\mathcal{D}_\epsilon(s(\theta, \omega), x_o)$ as the Phase-1 energy: the Gaussian negative log-likelihood (Gaussian-NLL), the energy distance and the maximum mean discrepancy (MMD). We run the comparison on Two Moons, SLCP and the Bernoulli GLM, each at its baseline prior. Within a task all surrogates share the same Phase-1 backbone, and training is capped at the simulation budget $N_{\text{sim}} \in \{10^3, 10^4, 10^5\}$. All entries are the mean ± standard deviation over 10 observations. Tables 6–13 report the results; for the multimodal tasks, Figs. 10 and 11 show the Phase-1 samples of each surrogate on one observation.

These ablations are experiments in their own right, separate from the benchmark runs: rather than reusing the benchmark settings, each surrogate is given its own bandwidth ϵ and temperature T, so that we compare reasonably tuned surrogates rather than one surrogate's defaults. These settings are chosen using the heuristic method of Sec. C.5 from a pilot batch of prior-predictive simulations, which uses no reference posterior. On Two Moons, ϵ is fixed at the scale of the pilot simulations (a Gaussian-NLL variance of 1.0×10^{-3}; an energy-distance length scale and MMD bandwidth of 0.78, the median pairwise distance), which is also the value that best separates the two energy

wells of the pilot landscape. T is then set from the same landscape so that, after well-tempered metadynamics with bias factor 10, about $3\,kT$ of the median barrier between the wells remains: $T = 12.8$, 0.068 and 0.032, with one simulator replicate per θ. On SLCP the bandwidths are adaptive (as in the benchmark), and T is set from the energy landscape in the same way: $T = 11.1$ for the Gaussian-NLL and $T = 0.011$ for MMD, while the energy distance keeps $T = 10^{-3}$ from short pilot runs on observation 1, with three replicates per θ. On the Bernoulli GLM we keep $T = 1$ for all surrogates, because the prior is Gaussian and T would temper it as well; sharpness is instead set through a fixed ϵ from the pilot: a Gaussian-NLL variance of 0.05, an energy-distance length scale of 0.914 (as in the benchmark runs) and an MMD bandwidth of 54.2 (about the median pairwise distance of the pilot simulations), with one, two and two replicates per θ.

Phase 1 only has to seed Phase 2 with a proposal that places mass on every posterior mode; it does not have to match the posterior. We therefore report two metrics, and a third, mode discovery, for the multimodal Two Moons and SLCP. The first is the C2ST of the Phase-1 samples against the reference posterior (lower is better, 0.5 means indistinguishable). The second is a *mode fidelity*, which measures how mass is distribued across modes, something C2ST does not explicitly test. Let τ be the 95th percentile of the nearest-neighbour distances within the reference sample. The *precision* P is the fraction of Phase-1 samples that lie within τ of a reference sample. The *recall* R_m of mode m is the fraction of reference samples in mode m that have a Phase-1 sample within τ. The mode fidelity is

$$F = P \cdot \min_m R_m \in [0, 1], \tag{23}$$

with 1 ideal. $F \to 0$ if Phase 1 misses a mode ($\min_m R_m \to 0$) or spreads its mass too thinly over the prior ($P \to 0$); $F > 0$ means every mode is covered. For the unimodal Bernoulli GLM, F reduces to precision times recall. On SLCP we compute F on (θ_3, θ_4) only: these coordinates separate the four modes, whereas the posterior is so narrow in the full five dimensions that almost no Phase-1 sample falls within τ of a reference sample, even when Phase 1 has found all four modes. Because F rewards samples that land on the reference support rather than near a mode, on Two Moons and SLCP we also report a coarser *mode discovery*: each Phase-1 sample is assigned to the nearest reference-mode centre (in θ for Two Moons, in (θ_3, θ_4) for SLCP) if it lies within 1.5 times that mode's 95th-percentile reference radius (at most half the distance between mode centres), and discovery is K times the smallest per-mode share (K modes; 1 means all mass split equally over the modes, 0 a missed mode). As a reference point, samples drawn from the prior score 0.04 on Two Moons and 0.23 on SLCP. Because the Phase-1 proposal is much broader than the posterior, F is small, and we report it in units of 10^{-3}.

Two Moons. Table 6) shows that with larger budgets the C2ST improves monotonically, to 0.875 for the energy distance, 0.922 for MMD and 0.941 for the Gaussian-NLL at 10^5. It never reaches 0.5, as expected, because Phase 1 learns the posterior under the surrogate likelihood rather than the simulator likelihood. Mode fidelity (Table 7) is positive for every surrogate and budget, so both moons are always seeded. The energy distance gives by far the best coverage at 10^4 and 10^5 simulations, albeit with large variation across observations. Mode discovery (Table 8, Fig. 10) ranks the surrogates as C2ST does: at 10^3 simulations all three are barely above the prior (0.09–0.11 against 0.04), while at 10^5 the energy distance reaches 0.64, MMD 0.53 and the Gaussian-NLL 0.32.

Table 6: Phase-1 **C2ST** on Two Moons (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Lower is better (0.5 = indistinguishable from the reference).

Surrogate	10^3	10^4	10^5
Gaussian-NLL	0.981 ± 0.005	0.955 ± 0.017	0.941 ± 0.011
Energy distance	0.982 ± 0.008	0.891 ± 0.047	0.875 ± 0.026
MMD	0.981 ± 0.007	0.934 ± 0.026	0.922 ± 0.012

Table 7: Phase-1 **mode fidelity** on Two Moons (baseline prior) vs. the simulation budget (capped), mean ± std over 10 observations. Values $\times 1000$.

Surrogate	10^3	10^4	10^5
Gaussian-NLL	1.560 ± 1.092	10.560 ± 6.981	19.030 ± 11.230
Energy distance	1.270 ± 1.328	57.630 ± 24.147	101.430 ± 37.410
MMD	1.480 ± 1.282	21.500 ± 10.500	38.990 ± 12.119

Table 8: Phase-1 **mode discovery** on Two Moons (baseline prior) vs. the simulation budget (capped), mean ± std over 10 observations. Higher is better. Samples drawn from the prior score 0.04.

Surrogate	10^3	10^4	10^5
Gaussian-NLL	0.114 ± 0.048	0.260 ± 0.105	0.323 ± 0.181
Energy distance	0.093 ± 0.063	0.527 ± 0.114	0.639 ± 0.234
MMD	0.107 ± 0.055	0.350 ± 0.115	0.530 ± 0.132

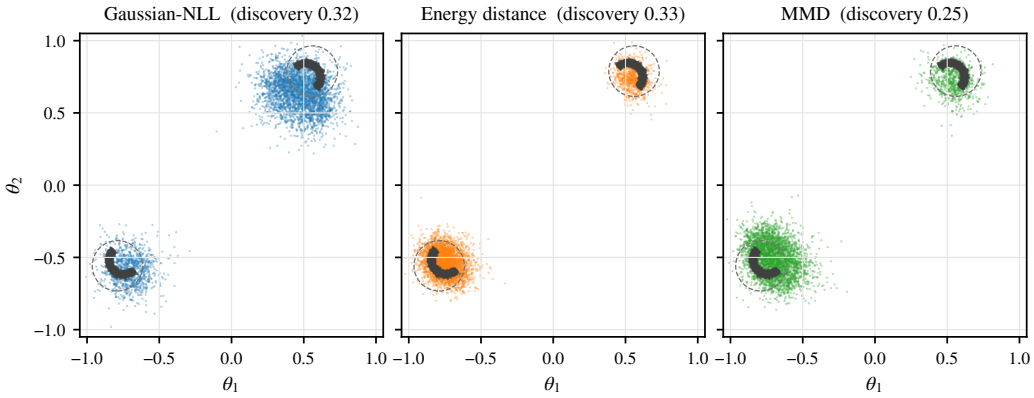

Figure 10: Phase-1 samples on Two Moons for each surrogate's profile (observation 1, 10^4 simulations; colour) over the reference posterior (grey). Dashed circles: the regions used for mode discovery ($1.5\times$ each moon's 95th-percentile reference radius). All three surrogates place a cluster on each moon; discovery is limited by how unevenly the mass is split between the moons and by clusters wider than the moons themselves.

SLCP. All three surrogates give a similar Phase-1 C2ST on SLCP, between 0.97 and 0.98 at every budget (Table 9); the differences are within one standard deviation. Mode fidelity on (θ_3, θ_4) (Table 10) is positive for every surrogate and budget, so all three seed all four modes. It is highest for the Gaussian-NLL (at 10^3 and 10^5 simulations) and the energy distance (at 10^4); at 10^4 and 10^5 it is three to four times lower for MMD, which places much of its mass on the edges of the prior box. Mode discovery (Table 11, Fig. 11) shows how weakly Phase 1 concentrates on SLCP at these budgets: the Gaussian-NLL and the energy distance are only moderately above the prior (0.34–0.39 at 10^4 and 10^5 simulations, against 0.23), with the Gaussian-NLL ahead at 10^3 (0.29), and MMD stays at the level of the prior (0.16–0.24); the variation across observations is large. The posterior itself is resolved in Phase 2.

Table 9: Phase-1 **C2ST** on SLCP (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Lower is better (0.5 = indistinguishable from the reference).

Surrogate	10^3	10^4	10^5
Gaussian-NLL	0.979 ± 0.011	0.976 ± 0.013	0.971 ± 0.016
Energy distance	0.983 ± 0.007	0.973 ± 0.015	0.970 ± 0.015
MMD	0.981 ± 0.011	0.981 ± 0.008	0.979 ± 0.014

Table 10: Phase-1 **mode fidelity** on SLCP (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Values $\times 1000$. Mode fidelity is computed on (θ_3, θ_4) only, the coordinates that separate the four SLCP modes (and carry the metadynamics bias).

Surrogate	10^3	10^4	10^5
Gaussian-NLL	68.642 ± 75.682	111.676 ± 133.676	138.893 ± 158.338
Energy distance	41.317 ± 46.915	130.302 ± 131.649	112.418 ± 109.132
MMD	41.959 ± 40.802	28.202 ± 28.300	36.760 ± 52.349

Table 11: Phase-1 **mode discovery** on SLCP (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Higher is better. Samples drawn from the prior score 0.23.

Surrogate	10^3	10^4	10^5
Gaussian-NLL	0.287 ± 0.241	0.338 ± 0.298	0.377 ± 0.332
Energy distance	0.173 ± 0.172	0.358 ± 0.266	0.389 ± 0.273
MMD	0.164 ± 0.163	0.224 ± 0.184	0.241 ± 0.183

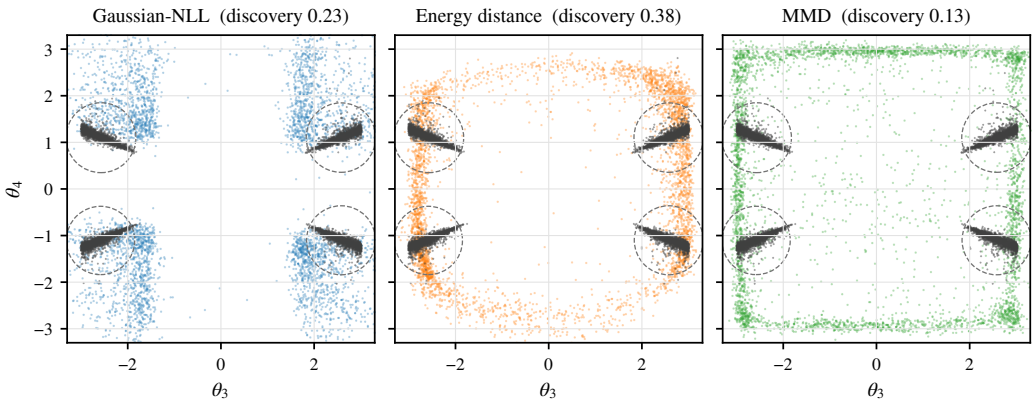

Figure 11: Phase-1 samples on SLCP in the mode-separating coordinates (θ_3, θ_4) for each surrogate's profile (observation 1, 10^4 simulations; colour) over the reference posterior (grey). Dashed circles: the regions used for mode discovery ($1.5\times$ each mode's 95th-percentile reference radius). The Gaussian-NLL puts its mass beside all four modes ($|\theta_3| \approx 2$), spread along θ_4; the energy distance spreads its mass in a ring around the prior box that passes through them; MMD concentrates on the box edges.

Bernoulli GLM. The surrogates differ most on the Bernoulli GLM (Table 12). The energy distance is the only surrogate whose C2ST improves steadily with the budget, reaching 0.687 at 10^5 simulations, and it also attains the highest mode fidelity, $F \approx 0.77$ (Table 13). The Gaussian-NLL stays at a C2ST of 0.97–0.99 with lower fidelity. MMD fails to concentrate on the posterior at all (C2ST $= 1.000$, $F = 0$).

Table 12: Phase-1 **C2ST** on Bernoulli GLM (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Lower is better (0.5 = indistinguishable from the reference).

Surrogate	10^3	10^4	10^5
Gaussian-NLL	0.969 ± 0.022	0.983 ± 0.011	0.988 ± 0.010
Energy distance	0.810 ± 0.066	0.725 ± 0.069	0.687 ± 0.059
MMD	1.000 ± 0.000	1.000 ± 0.000	1.000 ± 0.000

Table 13: Phase-1 **mode fidelity** on Bernoulli GLM (baseline prior) vs. the simulation budget (capped), mean $\pm$ std over 10 observations. Values $\times 1000$.

Surrogate	10^3	10^4	10^5
Gaussian-NLL	86.070 ± 84.689	231.940 ± 129.145	184.920 ± 175.774
Energy distance	430.270 ± 200.013	737.610 ± 71.645	767.610 ± 54.824
MMD	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000

C.7 Surrogate bandwidth ablations

Each surrogate has a bandwidth ϵ, namely the per-dimension variance of the Gaussian-NLL (Appendix C.1), the length scale of the energy distance (Appendix C.2) and the kernel bandwidth of the MMD (Appendix C.3). Each can be fixed or set adaptively from the simulations, as defined in those appendices. The benchmark runs use the adaptive variance for the Gaussian-NLL on SLCP and Lotka–Volterra and the adaptive length scale for the energy distance on Two Moons. They use a fixed energy-distance length scale of 0.914 on the Bernoulli-GLM, selected from the heuristic procedure of Appendix C.5, and $\epsilon = 1$ for the energy distance on `gw_ringdown`. MMD is not used in the benchmark, and its adaptive variant is the median heuristic.

To measure how sensitive Phase 1 is to ϵ, we rerun Phase 1 on observation 1 with each surrogate's ablation settings of Appendix C.6 (temperature, replicates and metadynamics) and change only ϵ, which we sweep over a log-spaced grid of fixed values. The temperature stays at the value chosen for the ablation bandwidth, so the sweep isolates ϵ. The budget is $N_{\text{sim}} = 10^4$ on SLCP and the Bernoulli GLM, while on Two Moons Phase 1 runs its total benchmark schedule (4,512 simulations). The grid contains the fixed ablation bandwidth of Appendix C.6 and, where the benchmark fixes one, the benchmark value (0.914 for the Bernoulli energy distance). On Two Moons it also contains 0.095, the value at which the benchmark's adaptive length scale settles once Phase 1 is near the posterior. For the energy distance and a fixed-variance Gaussian-NLL, ϵ only rescales the surrogate, so changing it is equivalent to changing the temperature of the likelihood term. For MMD it also changes the shape of the energy.

For each task we plot C2ST and mode fidelity F (Appendix C.6) against ϵ and summarise the sweep in two tables. The C2ST table compares the profile with the best fixed bandwidth $\epsilon^\star$, and shows how much C2ST degrades when $\epsilon^\star$ is off by a factor of ten in either direction. The fidelity table gives the band of ϵ over which $F > 0.01$, the bandwidth ϵ^{peak} that maximises F, with the precision P and worst-mode recall $\min_m R_m$ there, and F for the profile. Outside this band fidelity collapses in one of two ways. A bandwidth that is too small loses modes ($\min_m R_m \to 0$), and one that is too large spreads mass over the prior ($P \to 0$). On SLCP a third table (Table 18) reports mode discovery. We do not consider mode discovery for two moons since all surrogates recover the modes significantly.

Two Moons. Misspecifying $\epsilon^\star$ by an order of magnitude degrades the Phase-1 C2ST by 0.13–0.18, depending on the surrogate (Fig. 12, Table 14). The profile bandwidths, chosen without the reference, lie within half a decade of $\epsilon^\star$ and are within 0.09 of its C2ST (0.894 vs. 0.809 for the Gaussian-NLL, 0.812 vs. 0.788 for the energy distance and 0.843 vs. 0.810 for MMD). Mode fidelity is non-trivial only in a narrow band of ϵ (Table 15), yet every profile lies inside it ($F = 0.050$, 0.167 and 0.079, against peaks of 0.19–0.25).

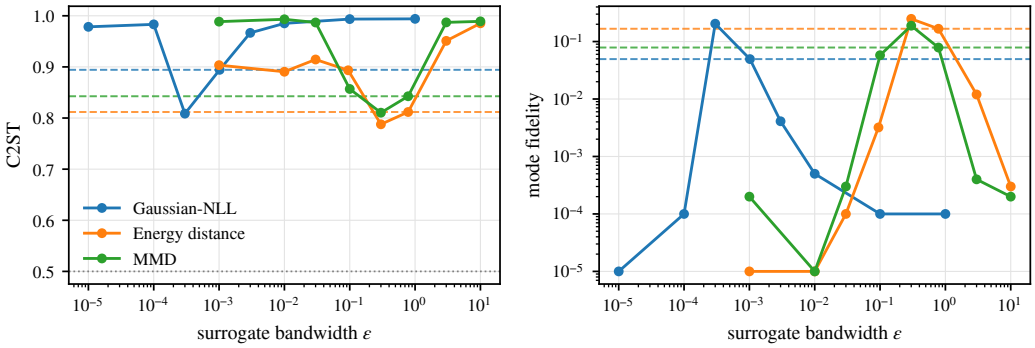

Figure 12: Phase-1 sensitivity to the surrogate bandwidth ϵ on Two Moons (4,512 simulations, one observation). Left: C2ST against the reference (lower is better; dotted: 0.5). Right: mode fidelity (= precision $\times$ worst-mode recall, log scale; zero drawn at 10^{-5}). Solid: each surrogate swept over its own bandwidth parameter (Gaussian-NLL variance, energy-distance length scale, MMD kernel bandwidth); dashed: the surrogate's profile setting.

Table 14: Phase-1 sensitivity to the surrogate bandwidth ϵ on Two Moons (4,512 simulations, one observation). *profile*: C2ST at each surrogate's bandwidth and temperature as chosen, without the reference, from the energy-landscape diagnostics. $\epsilon^\star$: the best fixed bandwidth on the sweep (identified using the reference), with its C2ST, and the C2ST when $\epsilon^\star$ is mis-set by one order of magnitude in either direction (change relative to $\epsilon^\star$ in brackets). Lower is better (0.5 = indistinguishable from the reference).

Surrogate	profile	$\epsilon^\star$	C2ST($\epsilon^\star$)	C2ST($0.1\,\epsilon^\star$)	C2ST($10\,\epsilon^\star$)
Gaussian-NLL	0.894	0.0003	0.809	0.978 (+0.170)	0.967 (+0.158)
Energy distance	0.812	0.3	0.788	0.914 (+0.127)	0.951 (+0.163)
MMD	0.843	0.3	0.810	0.987 (+0.176)	0.987 (+0.177)

Table 15: Phase-1 mode fidelity (= precision $\times$ worst-mode recall; higher is better) versus the surrogate bandwidth ϵ on Two Moons (4,512 simulations, one observation). Non-trivial band: the range of ϵ with mode fidelity above 0.01 (– if none); ϵ^{peak}: the bandwidth with the highest mode fidelity, with its precision and worst-mode recall; *profile*: mode fidelity at the surrogate's profile setting.

Surrogate	non-trvial band	ϵ^{peak}	fidelity	precision	min-recall	profile
Gaussian-NLL	[0.0003, 0.00102]	0.0003	0.204	0.308	0.664	0.050
Energy distance	[0.3, 3]	0.3	0.248	0.411	0.604	0.167
MMD	[0.1, 0.783]	0.3	0.189	0.371	0.509	0.079

SLCP. Phase-1 C2ST on SLCP is almost insensitive to ϵ: all values lie between 0.978 and 1.000 over four orders of magnitude, and $\epsilon^\star$ falls at the edge of the grid for the Gaussian-NLL and the energy distance (Fig. 13, Table 16). The profiles, which use adaptive bandwidths on SLCP, are within 0.005 of the best fixed value, and better than it for the Gaussian-NLL. For the Gaussian-NLL a fixed variance discards information: θ_3–θ_5 enter SLCP only through the noise covariance, which the adaptive per-θ variance captures and a fixed one does not. Accordingly, its mode fidelity on (θ_3, θ_4) stays at or below 0.003 for every fixed variance, against 0.007 for the adaptive profile (Table 17). The energy distance keeps $F > 0.01$ over the whole grid (peak 0.038) and MMD for $\epsilon \geq 3$ (peak 0.032). Mode discovery is shown in Table 18 and shows the cost of fixing the Gaussian-NLL variance: the adaptive profile reaches 0.23, whereas no fixed variance exceeds the prior (≤ 0.14), and variances up to 0.1 miss modes entirely. For the energy distance the adaptive profile (0.38) matches the best fixed length scale (0.37); for MMD the adaptive bandwidth (0.13) is below every fixed one (0.14–0.26). We note from the results of Fig. 11 that the mode discovery and fidelity metrics do not consider distinct isolation of the Phase 1 modes present with the GNLL.

Table 16: Phase-1 sensitivity to the surrogate bandwidth ϵ on SLCP (10,000 simulations, one observation). *profile*: C2ST at each surrogate's bandwidth and temperature as chosen, without the reference, from the energy-landscape diagnostics. $\epsilon^\star$: the best fixed bandwidth on the sweep (identified using the reference), with its C2ST, and the C2ST when $\epsilon^\star$ is mis-set by one order of magnitude in either direction (change relative to $\epsilon^\star$ in brackets). Lower is better (0.5 = indistinguishable from the reference).

Surrogate	profile	$\epsilon^\star$	C2ST($\epsilon^\star$)	C2ST($0.1\,\epsilon^\star$)	C2ST($10\,\epsilon^\star$)
Gaussian-NLL	0.984	20	0.988	0.991 (+0.003)	0.988 (+0.000)
Energy distance	0.983	0.001	0.978	0.978 (+0.000)	0.984 (+0.006)
MMD	0.986	10	0.983	0.988 (+0.005)	0.986 (+0.003)

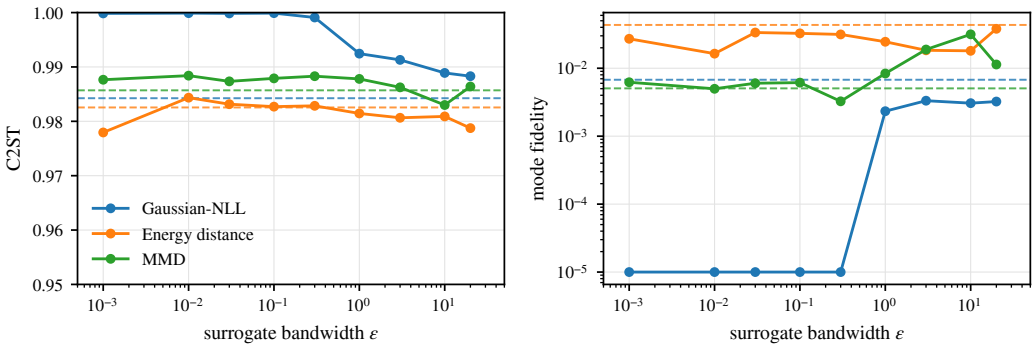

Figure 13: Phase-1 sensitivity to the surrogate bandwidth ϵ on SLCP (10,000 simulations, one observation). Left: C2ST against the reference (lower is better; dotted: 0.5). Right: mode fidelity (= precision $\times$ worst-mode recall, log scale; zero drawn at 10^{-5}). Solid: each surrogate swept over its own bandwidth parameter (Gaussian-NLL variance, energy-distance length scale, MMD kernel bandwidth); dashed: the surrogate's profile setting. Mode fidelity is computed on (θ_3, θ_4) only, the coordinates that separate the four SLCP modes (and carry the metadynamics bias).

Table 17: Phase-1 mode fidelity (= precision $\times$ worst-mode recall; higher is better) versus the surrogate bandwidth ϵ on SLCP (10,000 simulations, one observation). Non-trivial band: the range of ϵ with mode fidelity above 0.01 (– if none); ϵ^{peak}: the bandwidth with the highest mode fidelity, with its precision and worst-mode recall; *profile*: mode fidelity at the surrogate's profile setting. Mode fidelity is computed on (θ_3, θ_4) only, the coordinates that separate the four SLCP modes (and carry the metadynamics bias).

Surrogate	non-trivial	ϵ^{peak}	fidelity	precision	min-recall	profile
Gaussian-NLL	–	3	0.003	0.027	0.124	0.007
Energy distance	[0.001, 20]	20	0.038	0.132	0.288	0.043
MMD	[3, 20]	10	0.032	0.083	0.378	0.005

Table 18: Phase-1 mode discovery versus the surrogate bandwidth ϵ on SLCP (10,000 simulations, one observation). *profile*: discovery at the surrogate's profile setting (adaptive or fixed bandwidth); best fixed ϵ: the swept bandwidth with the highest discovery; range: lowest and highest discovery ov er the fixed bandwidths. Mode discovery: each Phase-1 sample is assigned to the nearest reference-mode centre in (θ_3, θ_4) if it lies within $1.5\times$ that mode's 95th-percentile reference radius (at most half the distance between mode centres); discovery $= 4\times$ the smallest per-mode share (1 = all mass split equally over the 4 modes, 0 = a mode missed). Samples drawn from the prior score 0.16.

Surrogate	profile	best fixed ϵ	discovery(ϵ)	range over fixed ϵ
Gaussian-NLL	0.23 (adaptive)	10	0.14	[0.00, 0.14]
Energy distance	0.38 (adaptive)	0.03	0.37	[0.11, 0.37]
MMD	0.13 (adaptive)	10	0.26	[0.14, 0.26]

Bernoulli GLM. On the Bernoulli GLM adaptive bandwidths fail (C2ST ≥ 0.995, $F = 0$ on observation 1): for the Gaussian-NLL the per-θ variance estimated from few replicates of the Bernoulli observations is too noisy. A fixed bandwidth is therefore required, and it matters (Fig. 14, Table 19): the profile bandwidths coincide with $\epsilon^\star$ for both the energy distance (C2ST 0.770) and the Gaussian-NLL (0.957), but the C2ST rises to 0.98–1.00 when they are off by a factor of 10. Both have a broad band of bandwidths with non-trivial fidelity, peaking at $F = 0.81$ for the energy distance (Table 20). MMD fails at every bandwidth in the grid. The fixed bandwidth that was used for the benchmarking in the main results was derived from the profile value and corresponds to the best performing bandwidth in this ablation.

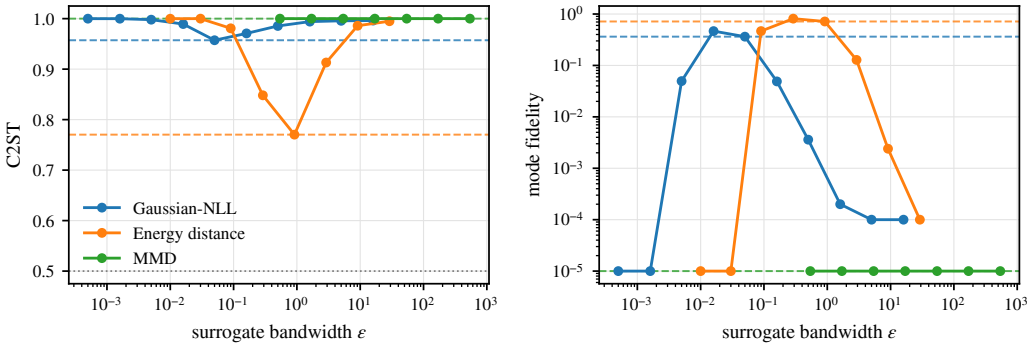

Figure 14: Phase-1 sensitivity to the surrogate bandwidth ϵ on Bernoulli GLM (10,000 simulations, one observation). Left: C2ST against the reference (lower is better; dotted: 0.5). Right: mode fidelity (= precision $\times$ worst-mode recall, log scale; zero drawn at 10^{-5}). Solid: each surrogate swept over its own bandwidth parameter (Gaussian-NLL variance, energy-distance length scale, MMD kernel bandwidth); dashed: the surrogate's profile setting.

Table 19: Phase-1 sensitivity to the surrogate bandwidth ϵ on Bernoulli GLM (10,000 simulations, one observation). *profile*: C2ST at each surrogate's bandwidth and temperature as chosen, without the reference, from the energy-landscape diagnostics. $\epsilon^\star$: the best fixed bandwidth on the sweep (identified using the reference), with its C2ST, and the C2ST when $\epsilon^\star$ is mis-set by one order of magnitude in either direction (change relative to $\epsilon^\star$ in brackets). Lower is better (0.5 = indistinguishable from the reference).

Surrogate	profile	$\epsilon^\star$	C2ST($\epsilon^\star$)	C2ST($0.1\,\epsilon^\star$)	C2ST($10\,\epsilon^\star$)
Gaussian-NLL	0.957	0.05	0.957	0.998 (+0.041)	0.985 (+0.028)
Energy distance	0.770	0.914	0.770	0.981 (+0.211)	0.986 (+0.216)
MMD	1.000	0.54	1.000	1.000 (+0.000)	1.000 (+0.000)

Table 20: Phase-1 mode fidelity (= precision $\times$ worst-mode recall; higher is better) versus the surrogate bandwidth ϵ on Bernoulli GLM (10,000 simulations, one observation). Non-trivial band: the range of ϵ with mode fidelity above 0.01 (– if none); ϵ^{peak}: the bandwidth with the highest mode fidelity, with its precision and worst-mode recall; *profile*: mode fidelity at the surrogate's profile setting.

Surrogate	non-trivial band	ϵ^{peak}	fidelity	precision	min-recall	profile
Gaussian-NLL	[0.005, 0.16]	0.016	0.466	0.984	0.473	0.363
Energy distance	[0.09, 2.9]	0.29	0.813	0.972	0.837	0.716
MMD	–	0.54	0.000	0.000	0.000	0.000

D Task details

This appendix comprises details on computational expense as well as descriptive summaries of each experiemnt conducted in this paper. For the experiments, RSBI required differentiable versions

of each simulator taken from the `sbibm` package. At the true parameters of observations 1 and 5, two-sample Kolmogorov–Smirnov tests on every output dimension, at the 5% level with Bonferroni correction, find no difference between the original `sbibm` simulators and our re-implementations for any task. Means agree to within 2.3 standard errors and standard deviations to within 6%, consistent with sampling noise at the sample sizes used (1.5×10^3 to 4×10^4 draws).

D.1 Compute Usage

Nearly all experiments in Section 5 are run on a single NVIDIA A100 40G GPU, with some run instead on an NVIDIA GTX 1660 Ti laptop GPU. The time taken for each RSBI run varies depending on the number of simulations and specific hyperparameters. In general, each RSBI run on the Two Moons simulator using the NVIDIA A100 40G GPU in the 10^5 simulation budget regime takes ~ 25 minutes to run. Total compute time on this specific experiment therefore approaches ~ 4 hours.

D.2 GW ringdown

Gravitational-wave ringdown is the final stage of a binary black-hole merger [70], occurring after the two black holes merge and the newly formed remnant settles into a single spinning black hole. During this phase the signal is dominated by a superposition of damped sinusoids, the quasi-normal modes, whose frequencies and decay times are determined entirely by the remnant's mass and spin according to general relativity. This makes ringdown a uniquely clean probe of strong-field gravity: measuring these modes constrains the remnant properties and tests general relativity in the high-curvature regime [71, 72].

Our differentiable ringdown simulator combines two quasi-normal modes ($\ell = m = 2, n \in \{0, 1\}$) with the H1 antenna pattern at the LIGO Hanford detector and additive Gaussian noise drawn from the H1 power spectral density estimated from LIGO first-observing-run data around GW150914. The strain is whitened via the Cholesky factor of the Toeplitz noise covariance built from the autocorrelation implied by this PSD, giving a unit-variance representation that sets the natural surrogate length scale $\epsilon = 1$. The analysis window is $T = 0.2\,\mathrm{s}$ at a sampling rate of $4096\,\mathrm{Hz}$, giving $d_x = 820$ samples per detector. The full forward map, from waveform generation through antenna projection, noise injection and whitening, is implemented in PyTorch so that $\nabla_\theta s(\theta, \omega)$ is available end-to-end, which is what makes Phase 1's surrogate gradient and Phase 2's adjoint matching applicable. The noise enters as Lz with $z \sim \mathcal{N}(0, I)$ and L the Cholesky factor, so the simulator is reparameterised exactly. Unlike the other tasks, this simulator is not a re-implementation. The same class generates the observations, the reference posteriors, the simulations of the baselines and those of RSBI, with the same detector, window and sampling settings.

We infer $\theta = (M, \chi, A_{220}, \varphi_{220}, A_{221}, \varphi_{221}, \cos\iota, \alpha, \cos\delta, \psi)$ with $d_\theta = 10$ in total. For the experiments reported in the main text the four extrinsic dimensions $(\cos\iota, \alpha, \cos\delta, \psi)$ are pinned to truth, the natural conditional setup for a single-detector analysis with known sky position and distance, reducing the inferred dimension to $d_\theta = 6$. Two prior scenarios are considered. The *default* scenario with $M \in [40, 100]\,M_\odot$, $\chi \in [0, 0.99]$, $A_{220}, A_{221} \in [0, 5]$, $\varphi_{220}, \varphi_{221} \in [0, 2\pi]$, $\cos\iota, \cos\delta \in [-1, 1]$, $\alpha \in [0, 2\pi]$, $\psi \in [0, \pi]$ covers astrophysically plausible stellar-mass remnants. The *wide* scenario with $M \in [10, 200]\,M_\odot$ and $A_{220}, A_{221} \in [0, 15]$, with the remaining ranges unchanged, relaxes the astrophysical priors to those a fundamental-physics analysis might use when remaining agnostic about the source population. The two scenarios give $\log_{10}\mathcal{R}_p \approx 2.6$ and 2.8 respectively (Table 21).

Because the noise model is exact in whitened coordinates, the likelihood is analytic given θ and we can sample a reference posterior directly. Ten injections are drawn uniformly from a sub-box of the default prior, with $M \in [60, 80]\,M_\odot$, $A_{220} \in [3, 5]$ and $A_{221} \in [2, 4]$ and the remaining parameters over their full default ranges, and one noise realisation is added per injection to produce x_o. The same ten observations are used across both prior scenarios, since widening the prior does not regenerate the data, only the reference posterior. Figure 3 shows observation 10, whose injected source is $M = 70.18\,M_\odot$, $\chi = 0.558$, $A_{220} = 4.249$, $\varphi_{220} = 5.873\,\mathrm{rad}$, $A_{221} = 2.724$ and $\varphi_{221} = 1.099\,\mathrm{rad}$. We draw $N_{\mathrm{ref}} = 10^4$ reference samples per observation. The `emcee` runs are computed once per (prior, observation) pair and reused across method runs.

We run RSBI at a single simulation budget of $N_{\mathrm{sim}} = 10^5$ with $R = 2$ Phase-2 rounds. Phase 1 uses 5.99×10^4 simulations, which seed the first NRE (topped up to 6×10^4), and the second round

adds 4×10^4 simulations drawn from the round-0 Phase 2 sampler. Phase 1's energy is the unbiased energy distance of Appendix C.2 with a fixed length scale $\epsilon = 1$ and $K = 4$ replicates. The source is the prior box with a soft boundary penalty ($\lambda_{\text{soft}} = 10$), which keeps the bridge from spending capacity on the prior corners. Well-tempered metadynamics acts in Phase 1 along $\xi(\theta) = (M, \chi)$ to cover the mass and spin basins, and Phase 2 starts from the Phase 1 samples reweighted by the final bias. Phase 2 runs without a bias, in coordinates whitened by the Phase 1 samples (Appendix A.5). Both prior scenarios share every setting, and the sampler operates on the prior box rescaled to $[0, 10]^6$. Inference is on the 6D free subspace, with all MLP input dimensions remapped from the on-disk $d_\theta = 10$ to 6. The full configuration is given in Sec. G.

D.3 SLCP

We evaluate RSBI on the SLCP task of Lueckmann et al. [18] ($\theta \in \mathbb{R}^5$, $x \in \mathbb{R}^8$) across the three prior boxes DEFAULT $= [-3, 3]^5$, WIDE5X $= [-5, 5]^5$, and WIDE10X $= [-10, 10]^5$ (see Tab. 21 for the corresponding prior-to-posterior volume ratios), each at simulator-call budgets $N_{\text{sim}} \in \{10^3, 10^4, 10^5\}$, on observations $1-10$ from the `sbibm` reference set. Posterior quality is reported via the C2ST against the reference posterior of each prior, evaluated on 10^4 posterior samples.

The simulator draws four i.i.d. points $x_j \sim \mathcal{N}(m_\theta, S_\theta)$, with $m_\theta = (\theta_1, \theta_2)$ and S_θ built from $s_1 = \theta_3^2$, $s_2 = \theta_4^2$ and $\rho = \tanh \theta_5$ as in `sbibm`. We re-implement it in PyTorch through the reparameterisation $x_j = m_\theta + L_\theta z_j$, $z_j \sim \mathcal{N}(0, I_2)$, where L_θ is the Cholesky factor of S_θ, computed in closed form for the 2×2 case with a diagonal jitter of 10^{-6}, so that $\nabla_\theta x$ is available by automatic differentiation. The baselines use the original `sbibm` simulator. The two implementations agree everywhere except where $\tanh \theta_5$ saturates in single precision ($|\theta_5| \gtrsim 8.3$, reachable only under WIDE10X). There `sbibm` clamps ρ to ± 0.9999, whereas we floor the diagonal of L_θ at 10^{-3}, and both only serve to keep the covariance positive definite.

The controlled-SDE drift $u_\phi(\theta, t)$ and the corrector head are Fourier-feature MLPs with sinusoidal time embedding and GELU activations, 6×64 (depth $\times$ width) in Phase 1 and 4×128 in Phase 2. The NRE classifier projects θ and x separately to 32 features each, concatenates them and applies a 6×64 MLP with SiLU activations. Both phases integrate the controlled SDE on $[0, 1]$ with an EDM-style noise schedule [73] ($\rho = 7$), using 64 Euler–Maruyama steps in Phase 1 and 256 in Phase 2.

Phase 1 fits a controlled SDE that approximately samples the surrogate posterior under the Gaussian-NLL surrogate distance $\mathcal{D}_\epsilon^{\text{NLL}}(s(\theta), x_o)$ of Section C.1, with the per-dimension variance estimated adaptively from K replicates of the differentiable simulator at each evaluation.

Phase 2 runs R rounds, $R = 1$ at $N_{\text{sim}} = 10^3$ and $R = 2$ at 10^4 and 10^5. The first NRE is trained on the Phase 1 simulations, topped up with draws from the Phase 1 sampler (10^3, 5×10^3 and 1.4×10^4 simulations in total), and the second round adds the remaining budget (5×10^3 and 8.6×10^4) from the previous round's sampler. Each round refines the controlled SDE, warm-started from the previous round and started from the Phase 1 sampler, against $E_r(\theta) = -\log p(\theta) - \log r_{\psi_r}(\theta, x_o)$. Well-tempered metadynamics [16, 17] biases the SDE along the collective variable $\xi(\theta) = (\theta_3, \theta_4)$ to encourage exploration of the four SLCP modes. The bias is reset to zero on the final round, so that the sampled distribution is unbiased up to importance reweighting (Section 4.2). Hyperparameters are listed in Sec. G.

D.4 Two Moons

We evaluate RSBI on the Two Moons task of Lueckmann et al. [18] ($\theta \in \mathbb{R}^2$, $x \in \mathbb{R}^2$) across the three prior boxes DEFAULT $= [-1, 1]^2$, WIDE10X $= [-10, 10]^2$, and WIDE50X $= [-50, 50]^2$ (see Table 21 for the corresponding prior-to-posterior volume ratios), each at simulator-call budgets $N_{\text{sim}} \in \{10^3, 10^4, 10^5\}$, on observations $1-10$ from the sbibm reference set. Posterior quality is reported via the C2ST against the closed-form sbibm reference posterior, sampled by rejection and evaluated on 10^4 posterior samples.

The controlled-SDE drift $u_\phi(\theta, t)$ and the corrector head are Fourier-feature MLPs with sinusoidal time embedding and GELU activations, 4×128 (depth $\times$ width) in Phase 1 and 6×64 in Phase 2. The NRE classifier projects θ and x separately to half its width each, concatenates them and applies

a 6-layer MLP with SiLU activations, of width 64 to 256 depending on the cell. Phase 1 integrates its controlled SDE on $[0, 1]$ with a Brownian schedule on the DEFAULT prior and an EDM-style schedule [73] on the wide priors, over 128 Euler–Maruyama steps (256 for WIDE50X). Phase 2 uses an EDM-style schedule over 256 steps throughout.

The Two Moons simulator

$$s(\theta;\, a, r) \;=\; \Big(r\cos a + 0.25 - \tfrac{|\theta_1+\theta_2|}{\sqrt{2}},\;\; r\sin a + \tfrac{-\theta_1+\theta_2}{\sqrt{2}} \Big)$$

is driven by the random angle a and radius r. We re-implement it in PyTorch with the noise drawn independently of θ, $a = \pi(u - \frac{1}{2})$ with $u \sim \mathcal{U}(0, 1)$ and $r = 0.1 + 0.01\, z$ with $z \sim \mathcal{N}(0, 1)$, as in `sbibm`, so that s is differentiable in θ away from the line $\theta_1 = -\theta_2$, where automatic differentiation returns a subgradient of $|\theta_1 + \theta_2|$. This implementation replaces the `sbibm` simulator for all methods, so RSBI and the baselines use identical code. Phase 1 uses the energy-distance surrogate $\mathcal{D}_\epsilon^{\mathrm{ED}}$ of Section C.2 with an adaptive length scale and $K = 1$ replicate per parameter.

Phase 2 runs $R = 2$ rounds in every cell. The first NRE is trained on the Phase 1 simulations, topped up with draws from the Phase 1 sampler, and the second round simulates the rest of the budget from the round-1 sampler, with $600 + 400$ simulations at $N_{\mathrm{sim}} = 10^3$, and $5{,}000 + 5{,}000$ and $55{,}000 + 45{,}000$ on the DEFAULT prior at 10^4 and 10^5 ($1{,}000 + 9{,}000$ and $5{,}000 + 95{,}000$ on the wide priors). Each round refines the warm-started controlled SDE against $E_r(\theta) = -\log p(\theta) - \log r_{\psi_r}(\theta, x_o)$. Well-tempered metadynamics is used in Phase 2 on the wide priors and on the DEFAULT prior at 10^3 simulations, and in Phase 1 on the DEFAULT prior at 10^4 and 10^5 simulations. It is off elsewhere. All Phase 1, Phase 2 and NRE hyperparameters, which differ between prior boxes and budgets, are listed in Sec. G.

D.5 Bernoulli GLM

We evaluate RSBI on the Bernoulli GLM task of Lueckmann et al. [18] ($\theta \in \mathbb{R}^{10}$, $x \in \mathbb{R}^{10}$), a generalised linear model in which the spike train of a neuron, driven by a white-noise stimulus over 100 time bins, is reduced to ten sufficient statistics. The prior is Gaussian, $\theta \sim \mathcal{N}(0, B^{-1})$, with the smoothness-inducing precision B of `sbibm`. We widen it by scaling the precision by 1, 0.04 and 0.01, which multiplies the prior standard deviations by 1, 5 and 10 (DEFAULT, MEDIUM and LARGE; see Tab. 21 for the corresponding prior-to-posterior volume ratios). Each prior is run at simulator-call budgets $N_{\mathrm{sim}} \in \{10^3, 10^4, 10^5\}$ on observations $1{-}10$ from the `sbibm` reference set, and posterior quality is reported via the C2ST against the reference posterior of each prior on 10^4 posterior samples.

We re-implement the `sbibm` simulator in PyTorch, batched over θ, with the same design matrix $X \in \mathbb{R}^{100\times 10}$ and stimulus. Each spike is drawn as $y_t \sim \mathrm{Bernoulli}(\sigma(\psi_t))$ with $\psi = X\theta$, and the summary statistics, the spike count and a nine-bin spike-triggered average, are computed exactly as in `sbibm`, which the baselines use. Binary spikes have no reparameterisation gradient. For Phase 1 we therefore differentiate through a straight-through Gumbel-softmax relaxation [30] of the spike draws (temperature 0.5). The forward pass returns the hard Gumbel-max sample with logits $(0, \psi_t)$, which is an exact Bernoulli draw, and the backward pass uses the gradient of the continuous relaxation, which is biased. All simulations, including those reused from Phase 1, are therefore exact, and the NRE is trained on the exact simulator.

The controlled-SDE drift $u_\phi(\theta, t)$ and the corrector head are Fourier-feature MLPs with sinusoidal time embedding and GELU activations, 6×128 (depth $\times$ width) in both phases. The NRE classifier projects θ and x separately to 64 features each, concatenates them and applies a 6×128 MLP with SiLU activations. Both phases integrate the controlled SDE on $[0, 1]$ with 128 Euler–Maruyama steps and an EDM-style noise schedule [73], with $\rho = 5$ in Phase 1 and $\rho = 7$ in Phase 2.

Phase 1 fits a controlled SDE that approximately samples the surrogate posterior under the energy-distance surrogate of Section C.2, with a fixed length scale $\epsilon = 0.914$ and $K = 2$ replicates of the differentiable simulator per parameter. Because the posterior is unimodal, no well-tempered bias is used in either phase. Phase 2 runs $R = 2$ rounds. The first NRE is trained on 25% of the budget, the Phase 1 simulations (topped up with draws from the Phase 1 sampler at $N_{\mathrm{sim}} = 10^5$), and the remaining 75% is simulated in the second round from the round-1 sampler. Each round refines the warm-started controlled SDE against $E_r(\theta) = -\log p(\theta) - \log r_{\psi_r}(\theta, x_o)$. The reference noise scale $\sigma_{\max}$ grows with the prior width. All hyperparameters are listed in Sec. G.

D.6 Lotka–Volterra

Lotka–Volterra is the classic predator–prey model and a standard SBI benchmark task [18]. Four rate parameters $\theta = (\alpha, \beta, \gamma, \delta)$ govern the oscillating populations of a prey and a predator species, and the goal is to infer them from a short, noisy record of both populations.

The simulator integrates $\dot{u} = f(u, \theta)$, $f(u, \theta) = (\alpha u_1 - \beta u_1 u_2,\ -\gamma u_2 + \delta u_1 u_2)$, from $u(0) = (30, 1)$ and records both species at $K = 10$ times $t_k = k\Delta$, $\Delta = 2.1$, with multiplicative noise $x_k = u(t_k) \odot \exp(\sigma \xi_k)$, $\xi_k \sim \mathcal{N}(0, I_2)$, $\sigma = 0.1$, giving $x \in \mathbb{R}^{2K}$. A differentiable fourth-order Runge–Kutta (RK4) scheme (step 0.05) provides $\nabla_\theta x$ by backpropagating through the unrolled RK4 steps, rather than by solving an adjoint ODE. `sbibm` instead solves the same ODE with Julia's adaptive Tsit5 solver, which the baselines use and which generated the observations for the references. Our PyTorch re-implementation keeps the same initial state, observation times, output clipping to $[10^{-10}, 10^4]$ and noise model. To check the fixed step, we compared noise-free outputs with an RK4 solve at step 0.001, at the true parameters and 200 reference posterior draws of every observation. The largest deviation is 2×10^{-4} in $\log x$, three orders of magnitude below the observation noise $\sigma = 0.1$. For rates far in the prior tails the solve can overflow, so populations are clamped to $[10^{-10}, 10^6]$ after every step, simulator gradients that are non-finite for a sample are set to zero and the rest are capped at norm 10^{12}. The prior term of the energy is differentiated separately, so such draws are still pulled back towards the support. RSBI samples $\phi = \log \theta$, under which the LogNormal prior becomes $\mathcal{N}(\mu_0, s^2 I)$, with $s = 0.5$, 1 and 2 for the DEFAULT, MEDIUM and LARGE priors. This is the same reparameterisation SNVI applies through its automatic support transform. Samples are mapped back with $\theta = e^\phi$ and evaluated in θ-space against the same reference posteriors as all baselines.

The Phase 1 surrogate is the Gaussian negative log-likelihood with adaptive variance (Sec. C.1) at temperature $T = 1$. Because each feature is scaled by its own variance, no additional feature standardisation (such as the per-feature z-scoring of the `sbi` toolkit, 53) is used.

The controlled-SDE drift $u_\phi(\phi, t)$ and the corrector head are Fourier-feature MLPs with sinusoidal time embedding, 6×128 (depth $\times$ width) with GELU activations in both phases. The NRE classifier projects ϕ and x separately to 64 features each, concatenates them and applies a 6×128 MLP with SiLU activations, on inputs standardised by the mean and standard deviation of the first round's training set. Both phases integrate the controlled SDE on $[0, 1]$ with an EDM-style noise schedule [73], using 64 Euler–Maruyama steps in Phase 1 and 128 in Phase 2, with $\rho = 5$ in Phase 1 ($\rho = 5$ in Phase 2 at $N_{\text{sim}} = 10^5$ and 7 otherwise). The reference noise scale is $\sigma_{\max} = 0.3$ in Phase 1 (0.5 for the wider priors at 10^5) and $\sigma_{\max} = 1$ in Phase 2, where it acts in the whitened coordinates of Appendix A.5. Phase 1 uses two simulator replicates per parameter (three at 10^5) and energy gradients clipped at norm 256.

At $N_{\text{sim}} = 10^3$ and 10^4, Phase 1 receives 70% of the budget and Phase 2 runs a single round ($R = 1$). Its NRE is trained on the Phase 1 simulations drawn after the horizon and shooting curriculum has ended, so it never sees masked or shooting-based data, topped up to the full budget with draws from the Phase 1 sampler. At 10^5, Phase 1 is capped at 5×10^4 simulations, the first NRE is seeded with its post-curriculum simulations, and two sequential rounds of 2.5×10^4 simulations follow ($R = 3$). For the LARGE prior at 10^5, Phase 1 instead receives 8×10^4 simulations and a single round ($R = 1$) is trained on its post-curriculum simulations topped up to the full budget. At every budget, Phase 2 is refined against the truncated ratio with $k = 3$ and $q = 0.9$, and k = 2 and q = 0.9 for the LARGE prior at 10^5, without a well-tempered bias since the posterior is unimodal. All hyperparameters are listed in Sec. G.

Surrogate likelihood choice Away from the posterior, a parameter error changes the period of the predator–prey cycle, so the phase error accumulates and late observations fall at essentially arbitrary points of a mismatched cycle. Any discrepancy on the full trajectory then becomes a rippled plateau in ϕ whose gradient carries little information about the direction of the posterior. We therefore use two modifications of the Phase 1 surrogate. Neither changes the data, the prior, the reference posterior or the evaluation metric. *(i) Horizon schedule.* For a horizon $H \le K$, the later features of each simulation are replaced by those of the observation, $[\Pi_H(x)]_k = x_k$ for $k < H$ and x_k^o otherwise, so they contribute exactly zero to the discrepancy while the data dimension is unchanged. Over a short prefix the phase error cannot wrap around the cycle, so E is close to convex over a much larger region. With ν the fraction of the Phase 1 budget consumed, $H = 3$ for $\nu < 0.3$, $H = 6$ for $0.3 \le \nu < 0.6$ and $H = K$ afterwards (at $N_{\text{sim}} = 10^5$ the switches occur at fixed counts of 1.5×10^4

and 3×10^4 simulations), so the end of Phase 1 targets the full trajectory. *(ii) Multiple shooting.* With $\Phi_\Delta(u, \theta)$ the RK4 flow over one observation interval, each observation is predicted from the *observed* state one interval earlier, $x_k^{\mathrm{sh}} = \Phi_\Delta(x_{k-1}^o, \theta) \odot e^{\sigma \xi_k}$ for $k \geq 1$ ($x_0^{\mathrm{sh}} = u(0) \odot e^{\sigma \xi_0}$). Phase errors cannot accumulate beyond one interval, and each segment needs 42 RK4 steps instead of up to 378. Conditioning on noisy states shifts the minimiser slightly (an errors-in-variables effect), so shooting is used only for $\nu < 0.5$, and the remaining bias is removed by the ratio-corrected Phase 2.

Both of the above are classical remedies for long-horizon gradients in differential-equation models, whose sensitivity to θ grows polynomially for oscillatory and exponentially for chaotic dynamics [74]. Growing the fitted window is a continuation (curriculum) strategy [75]. Multiple shooting splits the trajectory into short segments started from estimated or observed states [76, 77] and was recently shown to rescue neural ODE fitting on oscillatory data [78]. Smoothing-based estimators avoid long-horizon integration altogether [79]. Both modifications apply only to the Phase 1 surrogate, the energy that steers the variational sampler towards the posterior region before Phase 2 corrects it, and they cost no simulations beyond those Phase 1 already uses. They cannot be applied in the same way to SNPE or SNVI. These methods train a density or ratio estimator on a corpus of simulated pairs (θ, x), and that estimator is their final posterior. Masking the late observations of the corpus, or replacing its simulations with shooting segments, would change the conditional they estimate to $p(\theta \mid x_{<H})$ or to that of a different simulator, with no later phase to correct it. A horizon curriculum could at most supply an intermediate proposal, at the cost of training additional estimators on part of the budget, which neither method does. In RSBI, by contrast, the NRE and the Phase 2 target only ever see full, unmodified simulations. The horizon schedule needs only black-box access to $\theta \mapsto x$. Multiple shooting additionally assumes that the simulator can be restarted from an arbitrary state and that the full state is observed, both of which hold for Lotka–Volterra. It uses the simulator's latent process in the spirit of Brehmer et al. [29], the same white-box access RSBI already relies on through $\nabla_\theta s(\theta)$. As a rule, if the pilot gradient field in Sec. C.5 is incoherent away from the posterior, use the horizon schedule. If the state is fully observed and restartable, multiple shooting is a further option. If the field is already coherent, keep the full-trajectory surrogate.

D.7 Reference posteriors

We require a reference posterior $p(\theta \,|\, x_o)$ for every $(\text{task}, \text{scenario}, \text{observation})$ triple in the benchmark grid, since C2ST in Section 5 is computed against it. The sampler used to obtain these samples differs by task; we always draw $N_{\text{ref}} = 10^4$ samples per observation and reuse them across method runs.

For Two Moons the likelihood is fully analytic on the polar variables recovered by `sbibm`'s `TwoMoons._likelihood` [18]. The simulator can be therefore inverted in closed form. `sbibm`'s reference sampler draws the simulator noise, maps it to the parameter that reproduces x_o (choosing between the two branches at random), and rejects only draws outside the prior box. Each prior scenario is redrawn independently with its own box. For eight of the ten observations the posterior lies inside $[-1, 1]^2$, so all scenarios share the same reference. For observations 5 and 10 the default box truncates one moon, removing about 57% and 12% of the wide-prior posterior mass, and the wider scenarios recover the full moon.

For SLCP the analytic likelihood induces a four-mode posterior across which Markov chains mix poorly, so we use `sbibm`'s shipped SIR+NSF reference sampler for all three scenarios (DEFAULT, WIDE5X, WIDE10X). On the $20\times$-larger WIDE10X box this sampler fails for observations 4 and 10: the posterior occupies a tiny fraction of the box, the spline-flow proposal misses parts of it, and rejection sampling stalls at an acceptance rate of about 10^{-6}. For these two cases we exploit the closed-form SLCP likelihood directly. Because it depends on θ_3 and θ_4 only through θ_3^2 and θ_4^2, the posterior is exactly symmetric under sign flips of these two coordinates, so we sample the folded posterior ($\theta_3, \theta_4 \geq 0$) and attach independent random signs afterwards, which gives the four modes exactly equal mass. A short ensemble-MCMC pilot run [42] on the folded posterior locates the posterior mass; a mixture of Student-t distributions fitted to it, plus a 5% uniform component on the box, serves as an importance proposal against the exact posterior, from which we resample 10^4 points. As a check, the same procedure applied to observation 4 under the DEFAULT prior reproduces the `sbibm` reference (C2ST $= 0.50$).

For `gw_ringdown` the whitened-strain log-likelihood is the analytic Gaussian $-\frac{1}{2}\|L^{-1}(x_o - s(\theta))\|_2^2 +$ const, but the posterior has pronounced structure—periodic identifications on the phase dimensions

and a near-degeneracy between A_{220} and A_{221} at low signal-to-noise—so we sample via affine-invariant ensemble MCMC with `emcee` [42]: 256 walkers initialized in a 0.01-ball around the injected truth,10^5 steps per walker, 10^3 steps of burn-in, thinning factor 2, and an equal mixture of differential-evolution and stretch moves. Convergence is diagnosed via per-chain effective sample size (ESS $\gtrsim 100$ per parameter) and trace-plot inspection, rerunning with longer chains when either flag fires.

For Bernoulli GLM, we use the reference-posterior sampler that ships with `sbibm` [18] unchanged, and replace only the prior with the scenario prior (Table 21). The scenarios scale the precision matrix of the `sbibm` smoothness prior, so the prior remains Gaussian and the `sbibm` sampler applies directly. That sampler is an exact Gibbs sampler based on Pólya–Gamma data augmentation [80]. It is initialised at the true parameters, discards $25{,}000$ warm-up iterations and keeps every 25th draw, giving 10^4 samples.

For Lotka–Volterra, the default-prior reference is the `sbibm` reference posterior, obtained with its ODE solver and its log-normal observation model (noise scale 0.1) as the likelihood. A slice-sampling MCMC chain is started at the true parameters and run for 10^4 warm-up steps. A neural spline flow is then fitted to its samples and mixed with the prior (weight 0.1) to form a proposal, and rejection sampling from this proposal yields 10^4 exact posterior samples. The wider scenarios only widen the scale of the log-normal prior, with its location unchanged (Table 21). Their references are obtained by importance-resampling the default reference with weights $p_s(\theta)/p_{0.5}(\theta)$, the ratio of the scenario prior to the default prior, drawing 10^4 samples with replacement. Because every observation's posterior lies well inside the default prior, the weights are nearly uniform. The effective sample size is at least 86% of N for the medium prior and 78% for the large prior across the ten observations, and the resampled sets contain about $6{,}000$ to $6{,}400$ distinct draws.

D.8 Prior scenarios and volume ratios

Table 21: Prior scenarios and mean log prior-to-posterior volume ratios, averaged over the ten observations. Posterior volumes are estimated from the reference samples as the product of per-dimension 95% marginal widths and as the volume of the sample-covariance ellipsoid. For Gaussian priors, the prior volume is measured in the same way from its covariance (marginal widths $2\times 1.96\,\sigma_d$). Lotka–Volterra volumes are computed in $\log\theta$. Σ_0 denotes the default `sbibm` Bernoulli GLM prior covariance and $\mu_0 = (-0.125, -3, -0.125, -3)$.

Task	Scenario	Prior	$\log_{10}\mathcal{R}_p^{\text{marg.}}$	$\log_{10}\mathcal{R}_p^{\text{ell.}}$
Two moons	default	$\mathcal{U}([-1,1]^2)$	0.89	1.69
	wide10x	$\mathcal{U}([-10,10]^2)$	2.89	3.65
	wide50x	$\mathcal{U}([-50,50]^2)$	4.28	5.04
SLCP	default	$\mathcal{U}([-3,3]^5)$	1.57	3.46
	wide5x	$\mathcal{U}([-5,5]^5)$	1.93	4.05
	wide10x	$\mathcal{U}([-10,10]^5)$	2.69	4.99
Bernoulli GLM	default	$\mathcal{N}(0,\Sigma_0)$	5.38	3.55
	medium	$\mathcal{N}(0,25\,\Sigma_0)$	9.86	8.19
	large	$\mathcal{N}(0,100\,\Sigma_0)$	12.18	10.68
Lotka–Volterra	default	$\log\theta \sim \mathcal{N}(\mu_0, 0.5^2 I)$	5.05	6.26
	medium	$\log\theta \sim \mathcal{N}(\mu_0, I)$	6.24	7.46
	large	$\log\theta \sim \mathcal{N}(\mu_0, 2^2 I)$	7.44	8.67
`gw_ringdown`	default	$\mathcal{U}([40,100],[0,0.99],[0,5],[0,2\pi],[0,5],[0,2\pi])$	2.62	6.41
	wide	$\mathcal{U}([10,200],[0,0.99],[0,15],[0,2\pi],[0,15],[0,2\pi])$	2.84	6.73

For each task we consider a default prior and two wider ones (one for `gw_ringdown`): uniform boxes for Two Moons, SLCP and `gw_ringdown`, and Gaussian priors with inflated covariance for Bernoulli GLM and Lotka–Volterra (Table 21). For each (task, scenario, obs) triple we estimate $\log_{10}\mathcal{R}_p$ from the reference samples using a per-dimension 95%-marginal box estimator and a sample-covariance ellipsoid estimator (see Appendix F). For Gaussian priors the prior volume is measured with the same two estimators, from its covariance. The marginal estimator is the one we report throughout the main text, including the legends of the C2ST-versus-budget figures, because it is insensitive to

multimodality. The ellipsoid estimator, which accounts for correlations but is inflated when the posterior has several modes (as for SLCP), is given in the table for comparison only. Across tasks the scenarios span marginal $\log_{10} \mathcal{R}_p$ from about 0.9 to 12.

The `gw_ringdown` ratio changes little between the two scenarios (2.62 and 2.84), even though the wide box has about 1.4 orders of magnitude more prior volume, because the posterior widens with the prior. The overtone mode is only weakly constrained by single-detector data. Its phase ϕ_{221} is essentially unconstrained, and the 95% width of its amplitude A_{221} grows from about 4.8 to 14, nearly filling the wider amplitude range.

E Baselines

In this work we compare RSBI against two established neural-SBI methods. SNPE is configured as in `sbibm` [18]: a neural spline flow (`nsf`) density estimator with 50 hidden features, but with $R = 2$ with 50% of the budget allocated per round, and the `sbibm` training schedule.

SNVI+fKL is configured as in Glöckler et al. [7]: a `resnet` ratio estimator paired with an autoregressive spline flow proposal, but with $R = 5$ sequential rounds with 256 variational particles per round, and the β-divergence schedule released with the original implementation.

Software versions. All baselines of the main benchmark run on `sbibm` 1.1.1 (`https://github.com/sbi-benchmark/sbibm`), extended with our prior scenarios and the differentiable gravitational-wave task. SNPE runs on the `sbi` 0.16.0 fork released with the SNVI paper, and SNVI+fKL runs on that fork together with its `sbivibm` runners, both from the original implementation (`https://github.com/mackelab/snvi_repo`). These run under Python 3.10 and PyTorch 2.11. The Phase 2 ablations of Appendix B.3 use the current `sbi` release, 0.26.1 (`https://github.com/sbi-dev/sbi`), under Python 3.12 and PyTorch 2.10.

The careful reader will notice that we use fewer sequential rounds for SNPE and SNVI+fKL than mentioned in Refs. [7, 18]. Indeed, the results in Fig. 4 corresponding to the `sbibm` default slightly differ from these publications. This was done due to computational limitations. We also restrict the Phase-2 sequential posterior refinement of RSBI to a maximum of two rounds (matching SNPE) except for Lotka-Volterra for which we use three - in this case we checked the SNPE method with 3 rounds, yielding negligible difference to those recorded in the main results in Sec. 5.2. Furthermore, for SNPE and SNVI we use the default training procedures of the sbi package. The neural density estimator (SNPE) and the classifier-based ratio estimator underlying SNVI are trained with Adam, holding out 10% of the simulations for validation and stopping once the validation loss has not improved for 20 consecutive epochs; the weights with the best validation loss are kept. In addition, each round of SNVI fits a normalizing-flow variational posterior to the learned ratio by minimizing the forward KL divergence for at most 1,000 iterations, stopping early once the average slope of the running-mean loss over the last 50 iterations falls below 5×10^{-5} in absolute value. We do note however that the authors of the SNVI paper mention in the appendix, that the benchmark convergence criteria were "chosen conservative to avoid early stopping". This could indicate using as many as 1000 iterations in each VI stage, therefore, differences with the published scores could potentially be a result of this.

Lastly, both the SNVI release and `sbi` 0.26.1 estimate the forward KL with self-normalised importance weights $w_i \propto \exp(\log \tilde{p}(\theta_i) - \log q(\theta_i))$, where $\tilde{p}$ is the prior times the learned ratio. They exponentiate the log-weights before normalising, so a single log-weight above about 88 overflows in single precision and the loss becomes undefined. On such a step the optimiser restores the last finite state of q and refits it to the prior before continuing. On wide priors, a sharp ratio estimator easily produces such log-weights, and the step can recur until the variational posterior is spread back over the prior. We found this in the Phase 2 ablation (Appendix B.3), where it affected 9 of 10 observations at 10^4 simulations. For that ablation we compute the weights in log space with a softmax, which gives the same estimator without overflow, and on a non-finite step we only restore the last finite state of q. For the main benchmark we use the released implementation unmodified. We checked that starting from the prior, the variational posterior is broad and so the log-weights stay far below the overflow threshold. Rerunning three observations at 10^4 simulations on the widest Two Moons prior with the stabilised weights fix gave no non-finite steps, and recovered essentially the same C2ST for the released implementation that was used to generate the main results of this paper.

F Volume estimator for $\mathcal{R}_p$

For each task and prior scenario we estimate the posterior support volume non-parametrically from the $N_{\text{ref}} = 10{,}000$ reference samples $\{\theta^{(n)}\}_{n=1}^{N_{\text{ref}}} \subset \mathbb{R}^{d_\theta}$. Writing $\widehat{\mu}$ and $\widehat{\Sigma}$ for the sample mean and covariance, the unit-Mahalanobis ellipsoid $\mathcal{E} = \{\theta : (\theta - \widehat{\mu})^\top \widehat{\Sigma}^{-1} (\theta - \widehat{\mu}) \le 1\}$ has Lebesgue volume

$$V_{\text{ell}} \;=\; c_{d_\theta} \sqrt{\det \widehat{\Sigma}}, \qquad c_{d_\theta} \;=\; \frac{\pi^{d_\theta/2}}{\Gamma\!\left(\frac{d_\theta}{2} + 1\right)}, \tag{24}$$

where c_{d_θ} is the volume of the unit d_θ-ball and $\sqrt{\det \widehat{\Sigma}}$ maps it to the covariance ellipsoid. Working in log-space for numerical stability,

$$\log V_{\text{ell}} \;=\; \tfrac{1}{2} \log\det \widehat{\Sigma} \;+\; \tfrac{d_\theta}{2} \log \pi \;-\; \log \Gamma\!\left(\tfrac{d_\theta}{2} + 1\right), \tag{25}$$

with $\log\det \widehat{\Sigma}$ computed via `slogdet`. For a rectangular prior with uncorrelated distributions, $\theta \sim \mathcal{U}(\prod_i [a_i, b_i])$, the ellipsoid log-ratio is

$$\log_{10} \mathcal{R}_p^{\text{ell.}} \;=\; \frac{1}{\ln 10} \big(\log V_{\text{prior}} - \log V_{\text{ell}}\big), \qquad V_{\text{prior}} \;=\; \prod_{i=1}^{d_\theta} (b_i - a_i). \tag{26}$$

The ellipsoid is the one-σ contour of the best-fit Gaussian to the reference samples and therefore captures elongation and orientation that the per-dimension marginal-box estimator $V_{\text{marg.}} = \prod_i (q^{(i)}_{0.975} - q^{(i)}_{0.025})$ ignores. For uniform priors it is also much smaller than the marginal box, which spans about $\pm 2\sigma$ per dimension, so $\log_{10} \mathcal{R}_p^{\text{ell.}}$ exceeds its marginal counterpart in Table 21. For Gaussian priors, where prior and posterior are measured with the same estimator, these constants cancel and the two log-ratios differ by $\frac{1}{2}(\log\det C_{\text{prior}} - \log\det \widehat{C})$, with C the correlation matrices. This is positive for the isotropic Lotka–Volterra prior and negative for the strongly correlated smoothness prior of the Bernoulli GLM, whose ellipsoid ratio is therefore the smaller of the two. If $\widehat{\Sigma}$ were rank-deficient we would floor its eigenvalues at 10^{-300} before taking the log-determinant. This was not needed for any cell in Table 21.

For the Bernoulli GLM, whose prior is Gaussian, $\theta \sim \mathcal{N}(0, \Sigma_{\text{prior}})$ with $\Sigma_{\text{prior}} = (m\, B^{-1})^{-1}$ (B^{-1} the smoothness precision of Lueckmann et al. [18] and m the scenario's precision multiplier), the prior has no finite support volume, so we measure it with the same estimator as the posterior. The prior's unit-Mahalanobis ellipsoid has volume $c_{d_\theta} \sqrt{\det \Sigma_{\text{prior}}}$, and the ball constant cancels in the ratio,

$$\log_{10} \mathcal{R}_p^{\text{ell.}} \;=\; \frac{1}{2 \ln 10} \big(\log\det \Sigma_{\text{prior}} - \log\det \widehat{\Sigma}\big), \tag{27}$$

while the marginal-box counterpart replaces the box widths $b_i - a_i$ by the prior's central 95% widths $2 z_{0.975}\, \sigma_i$, with $\sigma_i^2 = (\Sigma_{\text{prior}})_{ii}$. The Lotka–Volterra prior is log-normal, $\log\theta \sim \mathcal{N}(\mu_0, s^2 I)$, so we apply the same Gaussian formulae in $\log\theta$, the coordinates in which the prior is Gaussian (and in which RSBI samples), with $\Sigma_{\text{prior}} = s^2 I$ and $\widehat{\Sigma}$ the covariance of the log-transformed reference samples.

G RSBI Hyperparameters

In this section we report hyperparameters used throughout the paper's various experiments.

G.1 Main Benchmarks

Tables 22–25 list the RSBI hyperparameters of every prior × budget cell in Figure 4. The prior scenarios are defined in Table 21.

Table 22: RSBI hyperparameters for the Two Moons results in Figure 4. One column per prior × budget cell. A value spanning several columns applies to all of them, and — marks a setting that is not used in that cell.

	default			wide10x			wide50x		
	10^3	10^4	10^5	10^3	10^4	10^5	10^3	10^4	10^5
Simulation budget									
Phase-1 simulations	536	4768	54 272		524			524	
NRE sims added, first round	64	232	728	76	476	4476	76	476	4476
NRE sims added, later rounds	400	5000	45 000	400	9000	95 000	400	9000	95 000
NRE buffer size	1000	10 000	100 000	1000	10 000	100 000	1000	10 000	100 000
Phase 1									
Surrogate	energy distance (ϵ from a 256-sample pilot, EMA 0.95)								
Temperature T	0.1	0.1	0.3		0.1			0.05	
MC replicates K	1								
Prior boundary softness λ	10	1	1		10			10	
SDE schedule		Brownian			EDM			EDM	
σ	0.2	0.2	0.3		—			—	
$\sigma_{\min}$ / $\sigma_{\max}$		—			0.3 / 5			0.3 / 10	
ρ		—			2			2	
Integration steps		128			128			256	
Buffer size	128	1024	8192		32			32	
Resample size	8	32	256		4			4	
Batch size	8	32	256		4			4	
ASBS stages	3	3	5		5			5	
Epochs per stage	10	55	60		20			20	
Iterations per epoch	50	100	100		15			15	
Learning rate / weight decay	5×10^{-4} / 10^{-7}								
LR warm-up steps	100								
Clip grad norm	1000	1000	100		1000			1000	
Clip energy grad norm	1000	1000	100		1000			1000	
Controller MLP	4×128, GELU								
WT height h_{wt}	—	0.05	5×10^{-4}		—			—	
WT width σ_{cv}	—	0.07	0.07		—			—	
WT bias factor γ	—	30	30		—			—	
WT grid bins	—	32	32		—			—	
WT grid range	—	± 1.1	± 1.1		—			—	
WT collective variable	—	all of θ	all of θ		—			—	
Phase 2									
Stages R (NRE rounds)	2								
Prior boundary λ	10	500 000	500 000		10			10	
SDE schedule	EDM								
$\sigma_{\min}$	0.001	10^{-6}	10^{-6}		0.001			0.001	
$\sigma_{\max}$	0.2	0.2	0.1	0.2	0.3	0.3	0.2	0.3	0.3
ρ	4	7	7		4			4	
Integration steps	256								
Buffer size	16 384	8096	8096	16 384	16 384	8192		16 384	
Resample size	512	1024	512		512			512	
Batch size	4096	1024	512	4096	4096	512		4096	
ASBS stages	5	3	3		5			5	
Epochs per stage	60	400	100		60			60	
Iterations per epoch	10	20	100	10	10	100		10	
Learning rate	0.01	0.001	0.001	0.01	0.01	0.001		0.01	
Weight decay	10^{-7}								
LR warm-up steps	1000	400	400		1000			1000	
Clip grad / energy grad norm	100 / 100	100 / 500	100 / 800		100 / 100			100 / 100	
Controller MLP	6×64, GELU								
WT height h_{wt}	5×10^{-4}	—	—		5×10^{-4}			5×10^{-4}	
WT width σ_{cv}	0.02	—	—		0.02			0.02	
WT bias factor γ	10	—	—		10		10	20	20
WT grid bins	150	—	—		350			350	
WT grid range	± 1.1	—	—		± 5.1			± 5.1	
WT collective variable	all of θ	—	—		all of θ			all of θ	
NRE									
NRE MLP	6×64	6×256	6×128	6×64	6×64	6×256	6×64	6×64	6×128
NRE epochs, first round	1500	500	50	1500	1500	25	1500	1500	50
NRE epochs, later rounds	2500	200	300	2500	500	100	2500	500	100
NRE batch size	128	200	200		128			128	
NRE learning rate	5×10^{-4}	0.001	0.001		5×10^{-4}			5×10^{-4}	
NRE contrast classes K_{class}	9	100	100		9			9	

Table 23: RSBI hyperparameters for the Lotka–Volterra results in Figure 4. One column per prior × budget cell. A value spanning several columns applies to all of them, and — marks a setting that is not used in that cell. † the Phase-1 simulations are discarded and the first NRE round trains on fresh draws from the Phase-1 sampler.

	10^3			10^4			10^5		
	default	medium	large	default	medium	large	default	medium	large
Simulation budget									
Phase-1 simulations	700			7000			49 725	50 000	80 000
NRE sims added, first round	300			3000			275	0	20 000
NRE sims added, later rounds	—			—			25 000	25 000	—
NRE buffer size	—			—			100 000	100 000	—
Phase-1 simulation cap	700			7000			50 000	50 000	80 000
NRE top-up to	1000			10 000			50 000	50 000	80 000
Phase 1									
Surrogate	Gaussian NLL (adaptive variance)								
Temperature T	1								
MC replicates K	2			2			3		
ODE horizons	3, 6, 10								
Horizon switch points	0, 0.3, 0.6 × cap			0, 0.3, 0.6 × cap			0, 15 000, 30 000 sims		
Shooting until	0.5								
SDE schedule	EDM								
$\sigma_{\min}$	10^{-5}								
$\sigma_{\max}$	0.3			0.3			0.3	0.5	0.5
ρ	5								
Integration steps	64								
Buffer size	100			100			2000		
Resample size	50			50			25	50	50
Batch size	100			100			1000		
ASBS stages	5			7			7		
Epochs per stage	2			19			146	78	130
Iterations per epoch	200			200			20		
Learning rate / weight decay	5×10^{-4} / 10^{-7}								
LR warm-up steps	10								
Clip grad / energy grad norm	10^6 / 256								
Controller MLP	6 × 128, GELU								
Phase 2									
Stages R (NRE rounds)	1			1			3	3	1†
SDE schedule	EDM								
$\sigma_{\min}$ / $\sigma_{\max}$	10^{-5} / 1								
ρ	7			7			5		
Integration steps	128								
Buffer size	2000								
Resample size	512								
Batch size	128			128			500		
Stages × epochs × iterations	5 × 50 × 50			5 × 50 × 50			7 × 150 × 20		
Learning rate / weight decay	5×10^{-4} / 0.003								
LR warm-up steps	100								
Clip grad norm	10^8			10^8			10^8	100	100
Clip energy grad norm	10^8			10^8			10^6	1000	1000
Controller MLP	6 × 128, GELU								
NRE									
NRE MLP	6 × 128								
NRE epochs, first round	200			200			300		
NRE epochs, later rounds	—			—			150	150	—
NRE batch size / learning rate	1000 / 6×10^{-4}			1000 / 6×10^{-4}			350 / 6×10^{-4}		
NRE contrast classes K_{class}	10								
NRE input standardised	yes								
Run-time guards									
NRE $\log r$ cap quantile	0.999								
NRE $\log r$ cap margin	2								
NRE truncation k	3			3			3	3	2
NRE truncation trim quantile	0.99			0.99			0.99	0.99	0.9
NRE truncation strength	50								
Round draw filter (prior s.d.)	5								
Phase-2 whitening trim quantile	0.99			0.99			0.99	0.99	0.9

Table 24: RSBI hyperparameters for the Bernoulli GLM results in Figure 4. One column per prior × budget cell. A value spanning several columns applies to all of them, and — marks a setting that is not used in that cell.

	10^3			10^4			10^5		
	default	medium	large	default	medium	large	default	medium	large
Simulation budget									
Phase-1 simulations		250			2500		22 980	24 980	24 980
NRE sims added, first round		0			0		2020	20	20
NRE sims added, later rounds		750			7500			75 000	
NRE buffer size		1000			10 000			100 000	
Phase 1									
Surrogate					energy distance ($\epsilon = 0.914$)				
Temperature T					1				
MC replicates K					2				
Simulator temperature					0.5				
SDE schedule					EDM				
$\sigma_{\min}$					10^{-5}				
$\sigma_{\max}$	1.5	10	10	1.5	10	10	1.5	10	10
ρ					5				
Integration steps					128				
Buffer size	51	26	26	501	251	251		2500	
Resample size		1			1			10	
Batch size	51	26	26	501	251	251		512	
ASBS stages	5	3	3	5	3	3	7	3	3
Epochs per stage	25	50	50	250	500	500	225	500	500
Iterations per epoch		50			5			5	
Learning rate / weight decay					5×10^{-4} / 10^{-7}				
LR warm-up steps					10				
Clip grad / energy grad norm					10^6 / 10^6				
Controller MLP					6×128, GELU				
Phase 2									
Stages R (NRE rounds)					2				
SDE schedule					EDM				
$\sigma_{\min}$					10^{-5}				
$\sigma_{\max}$	3	10	15	3	10	15	2.5	5	15
ρ					7				
Integration steps					128				
Buffer size					2000				
Resample size					512				
Batch size					128				
Stages × epochs × iterations		$5 \times 50 \times 50$			$5 \times 50 \times 50$			$7 \times 50 \times 50$	
Learning rate / weight decay					5×10^{-4} / 0.003				
LR warm-up steps					100				
Clip grad / energy grad norm					10^8 / 10^8				
Controller MLP					6×128, GELU				
NRE									
NRE MLP					6×128				
NRE epochs, first round					150				
NRE epochs, later rounds		50			50		50	25	25
NRE batch size / learning rate		50 / 6×10^{-4}			300 / 6×10^{-4}			300 / 6×10^{-4}	
NRE contrast classes K_{class}		16			100			100	

Table 25: RSBI hyperparameters for the SLCP results in Figure 4. One column per prior × budget cell. A value spanning several columns applies to all of them, and — marks a setting that is not used in that cell.

	10^3			10^4			10^5		
	default	wide5x	wide10x	default	wide5x	wide10x	default	wide5x	wide10x
Simulation budget									
Phase-1 simulations	576			4896			13 248		
NRE sims added, first round	424			104			752		
NRE sims added, later rounds	—			5000			86 000		
NRE buffer size	—			8096			50 000		
Phase 1									
Surrogate	Gaussian NLL (adaptive variance)								
Temperature T	1								
MC replicates K	3			3			4		
Prior boundary softness λ	20								
SDE schedule	EDM								
$\sigma_{\min}$ / $\sigma_{\max}$	0.01 / 1								
ρ	7								
Integration steps	64								
Buffer size	128								
Resample size	16								
Batch size	32								
Stages × epochs × iterations	$1 \times 5 \times 30$			$1 \times 95 \times 30$			$1 \times 200 \times 30$		
Learning rate / weight decay	0.005 / 0.001								
LR warm-up steps	50								
Clip grad / energy grad norm	1 / 100								
Controller MLP	6×64, GELU								
Phase 2									
Stages R (NRE rounds)	1			2			2		
SDE schedule	EDM								
$\sigma_{\min}$	10^{-5}								
$\sigma_{\max}$	3			3	5	5	5		
ρ	7								
Integration steps	256								
Buffer size	2048								
Resample size	128			1024			128		
Batch size	128								
Stages × epochs × iterations	$5 \times 200 \times 100$			$5 \times 500 \times 100$			$5 \times 500 \times 100$		
Learning rate / weight decay	0.001 / 0.001								
LR warm-up steps	100								
Clip grad norm	100								
Clip energy grad norm	1	—	100	100	—	—	—		
Controller MLP	4×128, GELU								
WT height h_{wt}	0.005								
WT width σ_{cv}	0.1								
WT bias factor γ	10								
WT grid bins	128								
WT grid range	± 3.5	± 5.5	± 10.5	± 3.5	± 5.5	± 10.5	± 3.5	± 5.5	± 10.5
WT collective variable	(θ_3, θ_4)								
NRE									
NRE MLP	6×64								
NRE epochs, first round	300			100			500		
NRE epochs, later rounds	—			100			100		
NRE batch size / learning rate	10 / 5×10^{-4}			128 / 5×10^{-4}			1028 / 5×10^{-4}		
NRE contrast classes K_{class}	4								

G.2 `gw_ringdown`

Table 26 lists the RSBI hyperparameters of the `gw_ringdown` experiment (Table 1 and Figure 3) for the two prior scenarios of Table 21. All ten observations of a scenario use the same configuration; the sampler runs on the prior box rescaled to $[0, 10]^6$, so the schedule and well-tempered settings are given in those coordinates.

Table 26: RSBI hyperparameters for the `gw_ringdown` results in Table 1 and Figure 3 ($N_{\text{sim}} = 10^5$). The default and wide prior scenarios share every setting.

Simulation budget	
Phase-1 simulations	59 900
NRE sims added, first round	100
NRE sims added, later rounds	40 000
NRE buffer size	100 000
Phase 1	
Sampler coordinates	prior box $\mapsto [0, 10]^6$
Surrogate	energy distance ($\epsilon = 1$)
Temperature T	0.025
MC replicates K	4
Prior boundary softness λ	20
SDE schedule	EDM
$\sigma_{\min}$ / $\sigma_{\max}$	10^{-9} / 1
ρ	5
Integration steps	256
Buffer size	2000
Resample size	25
Batch size	1000
Stages × epochs × iterations	3 × 260 × 20
Learning rate / weight decay	0.01 / 10^{-7}
LR warm-up steps	975
Clip grad / energy grad norm	1000 / 1000
Controller MLP	6 × 128, GELU
WT height h_{wt}	0.05
WT width σ_{cv}	0.5
WT bias factor γ	10
WT grid bins	60
WT grid range	$[-0.5, 10.5]$
WT collective variable	(θ_1, θ_2)
Phase 2	
Stages R (NRE rounds)	2
SDE schedule	EDM
$\sigma_{\min}$ / $\sigma_{\max}$	10^{-5} / 1
ρ	5
Integration steps	256
Buffer size	5000
Resample size	256
Batch size	256
Stages × epochs × iterations	7 × 200 × 100
Learning rate / weight decay	0.001 / 0.003
LR warm-up steps	100
Clip grad / energy grad norm	100 / 1000
Controller MLP	4 × 128, GELU
Source reweighed by WT bias	yes
NRE	
NRE MLP	4 × 64
NRE epochs, first round	200
NRE epochs, later rounds	200
NRE batch size / learning rate	600 / 6×10^{-4}
NRE contrast classes K_{class}	20
Run-time guards	
Phase-2 whitening trim quantile	0.99
Phase-2 whitening fit samples	10 000

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: All methods are explained in Section 4 and various experiments in Section 5.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: Limitations are discussed in Section 6 and assumptions are stated throughout Section 4.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: the paper leans on theoretical results derived elsewhere. Appropriate references are given.

Guidelines:

- The answer [N/A] 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: Full hyperparameter method details with pseudocode are given in both the main text (Section 4, Algorithm 2, Algorithm 1) and throughout the Appendix (e.g., see Appendix A.1).

   Guidelines:

   - The answer [N/A] 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: All model and simulator code, as well as model checkpoints for reproducibility have been publicly released here: `https://github.com/Chromeilion/rsbi_supplementary`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: All necessary information is given in the Appendix material, with configurations, hyperparameter choices for each task (see Appendix D)

   Guidelines:

   - The answer [N/A] 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: All experiments in Section 5 follow standard practice, being reported with mean and standard deviation calculated over 10 samples.

   Guidelines:

   - The answer [N/A] 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: Compute details are presented in Appendix D.1.

   Guidelines:

   - The answer [N/A] 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: our research respects the NeurIPS code of ethics.

   Guidelines:

   - The answer [N/A] 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: [N/A]

    Justification: Although the method has potential to be used within scientific inference pipelines in the future, what is presented here is foundational with limited potential for misuse.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: See above

    Guidelines: the code does not pose risk of misuse.

    - The answer [N/A] 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: All assets used that are not created by us have permissive licenses and have been properly cited both in the manuscript and in the code.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: Not applicable for this research.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: Not applicable for this research.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: Not applicable for this research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.