SPECTRA: EXACT COMPONENT TRANSPORT FOR TEST-TIME PRIOR ADAPTATION IN SIMULATION-BASED INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Simulation-based inference (SBI) has become a powerful approach to Bayesian inference in complex scientific models whose likelihoods are difficult or impossible to evaluate. Amortized SBI learns reusable inference models from simulated data, enabling rapid posterior inference for new observations, and modern generative models have made these models increasingly expressive. However, this reuse is limited to the prior distribution chosen during training, whereas scientific analyses often need revised priors as knowledge accumulates or alternative assumptions are tested. We introduce Spectra, a test-time adaptation method for diffusion-based SBI. Spectra uses an exact score-transport identity to obtain the adapted score from a frozen diffusion model in closed form for structured prior changes, without additional simulation or training. Across six SBI benchmarks, Spectra achieves accurate adaptation under strong prior shifts at low online sampling cost. This enables pretrained SBI models to incorporate updated prior information at test time.

1 INTRODUCTION

Bayesian inference estimates unknown model parameters θ from observed data x and quantifies their uncertainty through the posterior $p(\theta \mid x)$. For complex scientific models, however, the likelihood $p(x \mid \theta)$ is intractable or too costly to evaluate, even though synthetic data can be generated readily from a simulator (Cranmer et al., 2020). This makes standard likelihood-based Markov chain Monte Carlo impractical or unavailable (Robert & Casella, 2004). Simulation-based inference (SBI) instead learns from simulated parameter–observation pairs (Cranmer et al., 2020; Lueckmann et al., 2021). Amortized SBI makes this posterior inference reusable across observations by training an inference model once and applying it to new observations without repeating simulation and training (Greenberg et al., 2019; Gloeckler et al., 2024). This reuse has enabled applications ranging from real-time gravitational-wave inference (Dax et al., 2021; 2025) to cosmological constraints from galaxy clustering (Hahn et al., 2024) and identification of neural circuit configurations from electrophysiological recordings (Gonçalves et al., 2020).

Posterior distributions in scientific applications are often multimodal or strongly correlated, motivating flexible generative posterior models. Diffusion and score-based models generate samples by learning to reverse a gradual noising process (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021), and have been adapted to posterior sampling in SBI (Geffner et al., 2023; Sharrock et al., 2024; Gloeckler et al., 2024). Flow matching and consistency models have also been developed for SBI: flow matching learns time-dependent velocity fields for posterior sampling, while consistency models reduce generation to only a few inference steps (Wildberger et al., 2023; Schmitt et al., 2024). Transformer-based approaches can amortize across datasets by conditioning on a dataset provided as context at inference time, allowing a single model to be reused across datasets and tasks (Müller et al., 2022; Chang et al., 2025; Mittal et al., 2025; Whittle et al., 2026). In the standard fixed-prior setting considered here, training pairs are generated by drawing θ from a chosen prior π_{tr} and then simulating $x \sim p(x \mid \theta)$. This amortization therefore applies across observations, but only under the training prior π_{tr}.

However, the prior fixed at training time is rarely the only one needed in a scientific analysis. In gravitational-wave astronomy, population-level observations can inform priors for individual events (Moore & Gerosa, 2021), while assumptions about black-hole masses and spins can change the

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

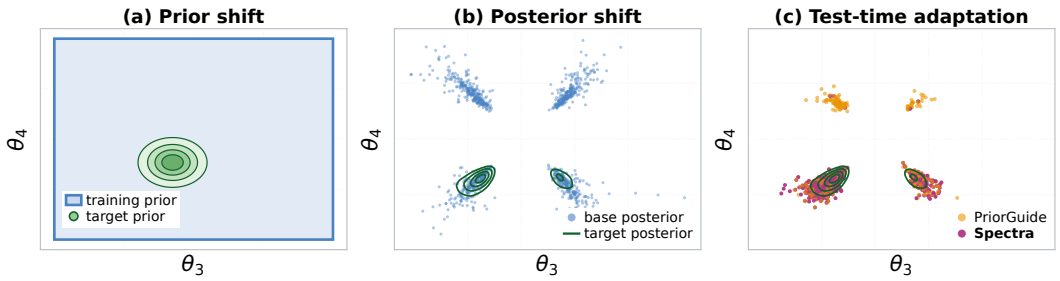


Figure 1: **Prior shift and test-time adaptation on the simple likelihood complex posterior (SLCP) task. (a)** A concentrated target prior replaces the broad training prior. **(b)** The frozen model represents the base posterior under the training prior, while the target prior concentrates posterior mass on two of its four modes. **(c)** Both PriorGuide and Spectra recover the target modes. However, PriorGuide retains visible mass outside the target contours, whereas Spectra follows the target geometry more closely. The posterior panels show the (θ_3, θ_4) projection. Full results across tasks and observations are reported in Table 1 and Figure 3.

inferred properties of a source (Vitale et al., 2017). The same issue appears in other fields: fossil-calibration priors affect estimates of species divergence times in evolutionary biology (Warnock et al., 2015), and prior-sensitivity analyses in cognitive science examine how evidence for competing models varies with the prior (Elsemüller et al., 2024). Priors may therefore need to change both as new information becomes available and as researchers test the robustness of their conclusions to alternative assumptions (Gelman et al., 2020). A single study may consequently require posterior inference under several plausible priors.

Each of these posteriors could be computed separately using a non-amortized inference procedure or by retraining the amortized model, but both options forgo much of the benefit of amortization. Retraining, in particular, repeats costly simulation and optimization for every new prior. Importance sampling can instead reweight samples from the original posterior, but becomes inefficient when the new posterior concentrates in regions poorly covered by the original samples (Agapiou et al., 2017). Another way to avoid per-prior retraining is to train across a family of priors, but test-time flexibility is constrained by the prior representation built into the model during training (Elsemüller et al., 2024; Chang et al., 2025; Whittle et al., 2026). In-context SBI with tabular foundation models avoids task-specific retraining, but still requires suitable simulated examples as context at inference time (Hollmann et al., 2025; Vetter et al., 2025). For an already trained diffusion-SBI model, recent methods adapt the prior at test time, either through an approximate sampling correction (Yang et al., 2026) or by training an auxiliary correction model for each target prior (Zang et al., 2026). For this setting, an effective adapter must remain accurate under substantial prior shifts, add little inference-time cost, and require neither additional simulation nor retraining.

We introduce Spectra, a test-time method for adapting pretrained diffusion-SBI models while leaving both the simulator and the trained model unchanged. Because diffusion sampling repeatedly queries the score network across noise levels, prior adaptation requires the corresponding score under the revised prior. Spectra obtains this score by exploiting analytic structure in the known prior change, using a single transformed query to the frozen model. For isotropic exponential–quadratic prior-ratio factors, this transformation is exact. Positive mixtures of such factors extend the construction to richer prior changes. Figure 1 illustrates the adaptation problem and a representative result on one benchmark task. Our main contributions are:

- **Exact score transport and characterization.** For prior changes that reweight the training prior by an isotropic exponential–quadratic factor, the adapted diffusion score follows exactly from one frozen-model query at a transformed state and noise level, without a score Jacobian or any Gaussian assumption on the base posterior. We further prove that, within this affine Gaussian transport construction, isotropic exponential–quadratic factors are the only positive reweightings that admit such an exact mapping.
- **Composition of richer prior changes.** Richer prior changes can be represented as positive mixtures of transportable factors, yielding mixtures of exactly transported posterior components. For a fixed observation, Spectra estimates the component weights once and

samples one component per reverse trajectory. Online sampling therefore still requires only one frozen-score evaluation per reverse step, independent of the number of components.

- **Accuracy and efficiency.** Across six SBI benchmarks, Spectra remains accurate under strong prior shifts and for mixture priors while keeping one frozen-score query per reverse step. Compared with a strengthened version of PriorGuide, the closest test-time method, its accuracy gains are largest on nonlinear problems, and its online sampling is 2.4–3.2× faster.

2 BACKGROUND

2.1 SIMULATION-BASED INFERENCE

Let $\theta \in \mathbb{R}^d$ denote the unknown parameters of a scientific model and x the observed data. A prior describes plausible parameter values before observing x, while the likelihood $p(x \mid \theta)$ describes how observations are generated from a given θ. We write π_{tr} for the prior used to train the inference model. Bayes' rule gives

$$p(\theta \mid x) = \frac{p(x \mid \theta)\pi_{\mathrm{tr}}(\theta)}{\int p(x \mid \vartheta)\pi_{\mathrm{tr}}(\vartheta)\, d\vartheta}. \tag{1}$$

The posterior $p(\theta \mid x)$ describes which parameter values remain plausible after observing x. In SBI, this posterior is learned from simulated parameter–observation pairs: training pairs are generated by drawing $\theta \sim \pi_{\mathrm{tr}}$ and then $x \sim p(x \mid \theta)$ (Cranmer et al., 2020). In amortized SBI, these pairs train a reusable model of $p(\theta \mid x)$, which can then be applied to new observations without further simulation or retraining (Greenberg et al., 2019; Gloeckler et al., 2024).

2.2 DIFFUSION MODELS

Diffusion models provide a flexible way to represent such posteriors by gradually corrupting samples with Gaussian noise and learning how to reverse this corruption (Sohl-Dickstein et al., 2015; Ho et al., 2020). Let $z_0 \sim p_0$ denote a clean sample and z_t its noisy counterpart at diffusion time $t \in [0, 1]$. We use the variance-exploding (VE) formulation, with increasing noise standard deviation $\sigma(t)$ and $\sigma(0) = 0$. The forward transition and resulting marginal density are

$$p_{t|0}(z_t \mid z_0) = \mathcal{N}\big(z_t; z_0, \sigma(t)^2 I\big), \qquad p_t(z_t) = \int p_{t|0}(z_t \mid z_0) p_0(z_0)\, dz_0. \tag{2}$$

The reverse process depends on the score $s(z_t, t) = \nabla_{z_t} \log p_t(z_t)$, the gradient of the log density at the current noisy state. Intuitively, the score points toward regions of higher probability and provides the direction used for denoising. For the VE process, the score determines the reverse-time stochastic differential equation

$$dz_t = -2\sigma(t)\dot{\sigma}(t)\nabla_{z_t} \log p_t(z_t)\, dt + \sqrt{2\sigma(t)\dot{\sigma}(t)}\, dW_t, \tag{3}$$

integrated from $t = 1$ toward $t = 0$ (Song et al., 2021). Here $\dot{\sigma}(t) = d\sigma(t)/dt$, and W_t denotes a Wiener process in reverse time. Because the marginal score is unknown, a neural network $s_\psi(z_t, t)$ is trained using denoising score matching (DSM) (Hyvärinen, 2005; Vincent, 2011):

$$\mathcal{L}_{\mathrm{DSM}}(\psi) = \mathbb{E}_{t, z_0, z_t}\Big[\lambda(t)\, \big\| s_\psi(z_t, t) - \nabla_{z_t} \log p_{t|0}(z_t \mid z_0) \big\|_2^2\Big], \tag{4}$$

where $z_0 \sim p_0$, $z_t \sim p_{t|0}(\cdot \mid z_0)$, t is sampled across noise levels, and $\lambda(t) > 0$ weights those levels. For Gaussian corruption, the conditional score is available analytically as $\nabla_{z_t} \log p_{t|0}(z_t \mid z_0) = -(z_t - z_0)/\sigma(t)^2$. Minimizing this objective learns the marginal score $s(z_t, t)$ in expectation (Vincent, 2011), providing the score needed by the reverse process. Sampling starts from a Gaussian approximation at the highest noise level and integrates the reverse dynamics using the learned score (Song et al., 2021).

2.3 DIFFUSION-BASED AMORTIZED SBI

In diffusion-based SBI, the clean variable is the parameter vector, $z_0 = \theta$, and the observation x conditions the denoising model. Applying the forward process from Section 2.2 to the posterior $p(\theta \mid x)$ gives the noisy posterior and its score

$$p_t(z \mid x) = \int p_{t|0}(z \mid \theta) p(\theta \mid x)\, d\theta, \qquad s_p(z, t, x) = \nabla_z \log p_t(z \mid x). \tag{5}$$

We refer to $p(\theta \mid x)$ as the base posterior and s_p as its noisy score. The subscript p distinguishes these quantities from the adapted posterior introduced below. A conditional network $s_\psi(z_t, t, x)$ is trained with the same denoising score-matching objective as in Section 2.2, using simulated pairs $\theta \sim \pi_{\mathrm{tr}}$ and $x \sim p(x \mid \theta)$, with x supplied as an additional condition (Geffner et al., 2023; Gloeckler et al., 2024). After training, we freeze the network and denote its learned score estimate by $\widehat{s}_p$. At inference time, the observation x is fixed and $\widehat{s}_p$ drives the reverse diffusion process, producing approximate samples from the base posterior. The same model can be reused for new observations, but its learned score still corresponds to the training prior π_{tr}.

3 SPECTRA: EXACT COMPONENT TRANSPORT

3.1 PRIOR ADAPTATION AS SCORE CORRECTION

Suppose the same observation x is analyzed under a new prior π_{new}, while the simulator remains unchanged. We assume π_{new} places no mass outside the support of π_{tr} and define the prior ratio $r(\theta) = \pi_{\mathrm{new}}(\theta)/\pi_{\mathrm{tr}}(\theta)$. Since the likelihood is unchanged, Bayes' rule gives

$$q(\theta \mid x) = \frac{p(\theta \mid x) r(\theta)}{Z_r(x)}, \qquad Z_r(x) = \mathbb{E}_{p(\theta|x)}[r(\theta)]. \tag{6}$$

In the diffusion formulation, Eq. (6) determines the target distribution at the clean endpoint. However, reverse sampling requires the corresponding target score at every noise level. A Bayes-rule calculation (Appendix C) gives

$$s_q(z, t, x) = s_p(z, t, x) + \nabla_z \log \mathbb{E}_{p(\theta|z,x)}[r(\theta)]. \tag{7}$$

Here $p(\theta \mid z, x)$ is the conditional distribution of the clean parameter given the noisy state z under the base posterior. The frozen model provides s_p, but not $p(\theta \mid z, x)$, so the correction in Eq. (7) is not directly available. PriorGuide makes this correction tractable by approximating $p(\theta \mid z, x)$ as Gaussian (Yang et al., 2026), while density-ratio adaptation learns an auxiliary correction for each target prior (Zang et al., 2026). Spectra instead shows that, for structured prior ratios, the adapted score can be obtained exactly from a transformed query to the frozen base score.

3.2 EXACT SINGLE-FACTOR TRANSPORT

Write $\tau = \sigma(t)^2$ for the noise variance, and index scores by τ in what follows. Consider the isotropic exponential–quadratic factor

$$\phi_{a,\kappa,c}(\theta) = \exp\Big(c + a^\top \theta - \frac{\kappa}{2}\|\theta\|^2\Big), \tag{8}$$

where $a \in \mathbb{R}^d$ and $\kappa, c \in \mathbb{R}$. If $r(\theta) \propto \phi(\theta)$, then $q^\phi(\theta \mid x) \propto p(\theta \mid x)\phi(\theta)$ is the target posterior. The quadratic term is isotropic: a single scalar κ controls its strength in every direction, while a controls an arbitrary linear tilt. For $\kappa > 0$, ϕ is proportional to an isotropic Gaussian with mean a/κ and covariance $\kappa^{-1} I$. For $\kappa = 0$, it reduces to a linear exponential tilt. The constant c only rescales the factor and cancels after posterior normalization. This restriction is on the prior-ratio factor while the base posterior itself can be arbitrary. For Gaussian training and target priors, their means may differ arbitrarily, while their precision matrices may differ only by an isotropic term.

The key observation is that multiplying the Gaussian corruption kernel by ϕ produces another Gaussian kernel at a transformed state and noise level:

$$\phi(\theta)\mathcal{N}(z; \theta, \tau I) = C_\tau(z)\mathcal{N}(m_\tau(z); \theta, \rho_\tau I), \tag{9}$$

where $C_\tau(z) > 0$ is an analytic prefactor and

$$D_\tau = 1 + \kappa\tau, \qquad \rho_\tau = \frac{\tau}{D_\tau}, \qquad m_\tau(z) = \frac{z + \tau a}{D_\tau}. \tag{10}$$

We require $D_\tau > 0$ at every noise level used by the sampler. This holds automatically for $\kappa \geq 0$; for $\kappa < 0$, it restricts the schedule to $\tau < -1/\kappa$. Substituting this identity into the integral defining the noisy posterior and differentiating with respect to z gives the target score in closed form (Appendix D):

$$\boxed{s_{q^\phi}(z, \tau, x) = \frac{a - \kappa z}{D_\tau} + \frac{1}{D_\tau} s_p(m_\tau(z), \rho_\tau, x).} \tag{11}$$

Proposition 1 (Exact score transport)**.** *Let $p(\theta \mid x)$ be any base posterior density with a differentiable Gaussian smoothing, and let $q^\phi \propto p\phi$ be normalizable. For $\tau > 0$ with $D_\tau > 0$, Eq.* (11) *gives the exact noisy score of q^ϕ.*

Each target-score evaluation therefore requires one base-score query at $(m_\tau(z), \rho_\tau)$ and a closed-form correction. The result applies to arbitrary base-posterior geometry under the stated regularity conditions. Appendix D gives the full derivation, learned-score error identity, and special cases.

3.3 Characterization of exact transport

The transport above follows from a kernel identity that is independent of the base posterior. This raises a natural question: which positive factors admit the same form of exact transport?

At a fixed noise level $\tau > 0$, consider

$$\phi(\theta)\mathcal{N}(z; \theta, \tau I) = C_\tau(z)\mathcal{N}(A_\tau z + u_\tau; \theta, \rho_\tau I), \tag{12}$$

where $\rho_\tau > 0$, $A_\tau \in \mathbb{R}^{d\times d}$, and $u_\tau \in \mathbb{R}^d$ depend only on τ, and $C_\tau(z) > 0$.

Theorem 2 (Characterization of exact transport)**.** *Fix $\tau > 0$. Suppose a positive factor ϕ admits the kernel transport in Eq.* (12) *for some affine state transform $A_\tau z + u_\tau$ and scalar variance $\rho_\tau > 0$. Then ϕ must have the isotropic exponential–quadratic form in Eq.* (8)*, and necessarily $A_\tau = I/D_\tau$, $u_\tau = \rho_\tau a$, and $\rho_\tau = \tau/D_\tau$. Conversely, every factor of the form in Eq.* (8) *satisfies Eq.* (12) *whenever $D_\tau > 0$.*

Thus this family is not chosen merely for convenience: within affine state transforms and scalar Gaussian variances, it is the complete class of positive factors admitting exact transport. Appendix E gives the proof. For the Gaussian-factor special case, closely related exact transformed-query identities arise in Gaussian conditioning (Holderrieth et al., 2026; Mammadov et al., 2026). Appendix A discusses the relation.

3.4 Combining posterior components

These exact single-factor transports can be used as building blocks for richer prior ratios. Specifically, consider a positive weighted sum of K factors

$$r(\theta) = \sum_{k=1}^{K} b_k \phi_k(\theta), \qquad b_k > 0, \tag{13}$$

where each ϕ_k has the form in Eq. (8). Each factor defines a posterior component $q_k(\theta \mid x) = p(\theta \mid x)\phi_k(\theta)/Z_k(x)$ with $Z_k(x) = \mathbb{E}_{p(\theta|x)}[\phi_k(\theta)]$, assuming $0 < Z_k(x) < \infty$. The score of each component is given exactly by Eq. (11). Substituting Eq. (13) into Eq. (6) and normalizing gives

$$q(\theta \mid x) = \sum_{k=1}^{K} \alpha_k(x) q_k(\theta \mid x), \qquad \alpha_k(x) = \frac{b_k Z_k(x)}{\sum_j b_j Z_j(x)}. \tag{14}$$

Thus the coefficients b_k belong to the chosen representation of the known prior ratio and are fixed independently of x. In contrast, the posterior mixture weights $\alpha_k(x)$ depend on the observation through $Z_k(x)$. Appendix F gives the mixture algebra. If Eq. (13) approximates r rather than representing it exactly, the same algebra gives the posterior induced by the approximating ratio. Appendix G quantifies its discrepancy from the exact target posterior.

3.5 Sampling one component per trajectory

Eq. (14) expresses the target posterior as a mixture of the component posteriors q_k. For a fixed observation x, $\alpha_k(x)$ are the static weights of this posterior mixture. Sampling from the mixture is straightforward: for each output sample, first select component k with probability $\alpha_k(x)$, then draw from $q_k(\theta \mid x)$. Let J denote the selected component:

$$\Pr(J = k \mid x) = \alpha_k(x), \qquad \theta \mid J = k, x \sim q_k(\theta \mid x). \tag{15}$$

Marginalizing over J recovers the target mixture in Eq. (14). Spectra realizes the conditional draw by reverse diffusion, using component J's transported score throughout the trajectory. Each reverse step therefore requires one base-score query, independent of the number of components K.

A natural alternative is to follow the score of the full mixture, which requires state-dependent component responsibilities at every reverse step. These responsibilities depend on normalized component densities and are not directly available from local queries to the frozen score model. Sampling the component once avoids this per-step recombination. Appendix F gives the formal argument and derives the full-mixture score. Implementation details are given in Appendix J.3.

3.6 Estimating component weights

For $K > 1$, the only remaining quantities to estimate in Eq. (14) are the observation-specific normalizers $Z_k(x) = \mathbb{E}_{p(\theta|x)}[\phi_k(\theta)]$. In practice, the estimator operates on the finite reverse chain of the base sampler. Let $\widetilde{Z}_k(x)$ denote the corresponding finite-chain counterpart of $Z_k(x)$. A direct estimate of $\widetilde{Z}_k(x)$ averages ϕ_k over endpoint samples from the base chain. However, this can suffer from poor overlap when component k concentrates in a region rarely visited by those samples. We therefore use the corresponding transported component process as an importance-sampling proposal.

Let $\omega = (z_R, \dots, z_0)$ denote a reverse trajectory, and let $P(\omega)$ and $Q_k(\omega)$ be the path densities of the base and component-k chains used by the estimator. Their finite-step transition densities are known, so the trajectory likelihood ratio $P(\omega)/Q_k(\omega)$ is tractable even when the endpoint densities are not. Under common path support, change of measure for the finite reverse chain gives

$$\widetilde{Z}_k(x) = \mathbb{E}_{\omega\sim P}[\phi_k(z_0)] = \mathbb{E}_{\omega\sim Q_k}\left[\phi_k(z_0)\frac{P(\omega)}{Q_k(\omega)}\right]. \tag{16}$$

Monte Carlo averaging of the right-hand side gives $\widehat{Z}_k$, which is used in Eq. (14) to form the component weights. Appendix H gives the trajectory likelihood ratio, overlap diagnostics, the direct estimator, and the full procedure. Appendix G separates finite-sampler and Monte Carlo weight-estimation errors. This is an upfront, observation-specific calculation, with proposal trajectories generated separately for each component. For a fixed observation and factor set, the estimated normalizers can be reused when only the coefficients b_k change.

4 Experiments

We evaluate Spectra on six SBI benchmarks under both single-factor and mixture prior shifts, adapting frozen diffusion-based Simformer models (Gloeckler et al., 2024) without retraining or additional simulator calls. Across these experiments, compared with a strengthened version of PriorGuide, Spectra improves posterior accuracy most clearly on nonlinear tasks under strong shifts, remains comparable on OUP and the Gaussian Linear controls, improves mixture-posterior agreement on all six tasks, and reduces steady-state single-factor online sampling time by 2.4–$3.2\times$ at the sampler settings used for accuracy.

4.1 Experimental setup

We consider six benchmark tasks: Two Moons and simple likelihood complex posterior (SLCP) (Lueckmann et al., 2021), the Ornstein–Uhlenbeck process (OUP) (Uhlenbeck & Ornstein, 1930), Bayesian causal inference (BCI) in multisensory perception (Körding et al., 2007), and Gaussian Linear problems (Lueckmann et al., 2021) in 10 and 20 dimensions (GL-10D and GL-20D), included as Gaussian controls. Following Yang et al. (2026), single-factor target priors are isotropic Gaussians with 0.5 (mild) or 0.2 (strong) times the training-prior marginal standard deviation. Mixture shifts use two strong Gaussian components. Full prior construction, posterior references, and support checks are given in Appendix J. These benchmark shifts lie in the exactly transportable component class. Appendix N studies anisotropic and correlated Gaussian targets outside the exact single-factor class using positive-mixture approximations, where Spectra remains more accurate than PriorGuide.

Baselines include PriorGuide (Yang et al., 2026), the unadapted frozen model (Base), and sampling-importance-resampling (SIR) (Rubin, 1988). PriorGuide receives the exact analytic prior ratio and uses the same posterior-sampling settings as Spectra, while SIR reweights samples from Base by the prior ratio. We write a sampler configuration as (N, L), where N is the number of reverse time-grid points and L the number of Langevin corrector updates per interval. Sampling and weight-estimation details are given in Appendix J.3 and Appendix H.

Table 1: **Posterior accuracy under single-factor and mixture prior shifts.** Mean (standard deviation) across ten observations or 30 prior–observation pairs. Ideal values are 0.5 for C2ST and 0 for MMTV. Bold marks the mean closest to the ideal value within each task and setting.

		Single-factor, mild		Single-factor, strong		Mixture	
Task	Method	C2ST	MMTV	C2ST	MMTV	C2ST	MMTV
Two Moons	Base	0.568(0.036)	0.219(0.087)	0.691(0.095)	0.426(0.192)	0.736(0.074)	0.510(0.147)
	SIR	0.525(0.005)	**0.075**(0.011)	0.544(0.018)	**0.070**(0.012)	0.543(0.018)	**0.062**(0.009)
	PriorGuide	0.528(0.012)	0.123(0.046)	0.556(0.034)	0.165(0.086)	0.561(0.036)	0.176(0.086)
	Spectra	**0.521**(0.005)	0.093(0.016)	**0.532**(0.015)	0.102(0.030)	**0.521**(0.009)	0.073(0.014)
SLCP	Base	0.765(0.057)	0.255(0.041)	0.917(0.019)	0.466(0.072)	0.946(0.029)	0.498(0.075)
	SIR	0.640(0.086)	**0.123**(0.054)	0.713(0.067)	**0.101**(0.043)	0.796(0.086)	0.205(0.269)
	PriorGuide	0.632(0.108)	0.128(0.061)	0.637(0.061)	0.122(0.034)	0.618(0.054)	0.150(0.057)
	Spectra	**0.631**(0.108)	0.128(0.060)	**0.608**(0.073)	0.107(0.044)	**0.570**(0.056)	**0.081**(0.031)
OUP	Base	0.587(0.064)	0.186(0.075)	0.742(0.088)	0.397(0.079)	0.703(0.063)	0.322(0.085)
	SIR	0.524(0.009)	0.054(0.009)	0.548(0.050)	**0.045**(0.011)	0.529(0.011)	**0.046**(0.012)
	PriorGuide	**0.507**(0.003)	**0.053**(0.006)	**0.505**(0.009)	0.047(0.011)	0.509(0.008)	0.049(0.017)
	Spectra	0.508(0.002)	**0.053**(0.006)	**0.505**(0.009)	**0.045**(0.010)	**0.507**(0.006)	**0.046**(0.012)
GL-10D	Base	0.898(0.024)	0.308(0.028)	0.997(0.002)	0.664(0.038)	0.996(0.003)	0.636(0.044)
	SIR	0.760(0.056)	0.225(0.260)	0.998(0.002)	0.965(0.064)	0.999(0.001)	0.965(0.103)†
	PriorGuide	0.541(0.020)	**0.044**(0.007)	0.526(0.019)	**0.038**(0.006)	0.557(0.029)	0.080(0.028)
	Spectra	**0.540**(0.019)	**0.044**(0.007)	**0.524**(0.017)	**0.038**(0.005)	**0.543**(0.030)	**0.052**(0.018)
GL-20D	Base	0.947(0.016)	0.300(0.017)	1.000(0.000)	0.645(0.017)	0.999(0.001)	0.640(0.030)
	SIR	0.927(0.029)	0.888(0.061)	1.000(0.000)	0.852(0.139)†	1.000(0.000)	0.900(0.125)†
	PriorGuide	**0.555**(0.022)	**0.042**(0.003)	**0.529**(0.021)	0.035(0.004)	0.529(0.011)	0.051(0.021)
	Spectra	**0.555**(0.023)	**0.042**(0.003)	**0.529**(0.020)	**0.034**(0.004)	**0.521**(0.012)	**0.036**(0.016)
BCI	Base	0.758(0.030)	0.254(0.049)	0.948(0.016)	0.497(0.045)	0.946(0.015)	0.485(0.039)
	SIR	0.591(0.026)	0.086(0.031)	0.799(0.066)	0.224(0.312)	0.789(0.061)	0.183(0.206)
	PriorGuide	0.569(0.028)	0.087(0.028)	0.575(0.013)	0.081(0.011)	0.548(0.019)	0.072(0.020)
	Spectra	**0.567**(0.028)	**0.085**(0.028)	**0.551**(0.015)	**0.066**(0.010)	**0.544**(0.017)	**0.061**(0.011)

PriorGuide receives the exact prior ratio. Sampling configurations are $(N, L) = (25, 8)$ for PriorGuide and Spectra, except single-factor BCI, where both use $(250, 0)$ following Yang et al. (2026); Base and SIR use $(100, 0)$. At the matched $(100, 0)$ configuration, Spectra attains a lower MMTV than SIR on Two Moons and SLCP (Appendices K.1 and L.1).
† SIR MMTV is undefined for some Gaussian Linear runs. Valid counts are reported in Appendices K and L.

Posterior agreement is assessed using two complementary metrics. The classifier two-sample test (C2ST; Lopez-Paz & Oquab, 2017) assesses joint agreement, including parameter dependencies, with an ideal value of 0.5 corresponding to chance-level classification. Mean marginal total variation (MMTV) measures average discrepancy across one-dimensional marginals, with an ideal value of zero.

4.2 Single-factor prior adaptation

Under strong prior shifts, Spectra shows better agreement with the reference posterior than PriorGuide on Two Moons, SLCP, and BCI, while the two methods perform similarly on OUP and the Gaussian Linear controls (Table 1).

To separate shift strength from changes in prior center and observation, we additionally vary the target-prior width while holding the prior center, observations, backbone, and sampler fixed. The gap between PriorGuide and Spectra grows systematically on Two Moons as the target prior narrows, but remains essentially flat on OUP (Figure 2; Appendix K.4). The effect of shift severity therefore depends on posterior geometry rather than shift strength alone.

Figure 3 visualizes representative strong-shift posteriors. Spectra follows the target geometry closely across the displayed regimes, whereas PriorGuide exhibits off-target mass on the nonlinear examples. SIR illustrates a complementary trade-off: reweighting can be effective when the Base posterior already has substantial overlap with the target, but becomes inefficient as that overlap deteriorates. Additional controlled comparisons and diagnostics are reported in Appendix K.

4.3 Mixture prior shifts

Across 180 prior–observation pairs on six tasks, Spectra with path-space weight estimation shows better agreement with the reference posterior than PriorGuide on every task, with the clearest gains

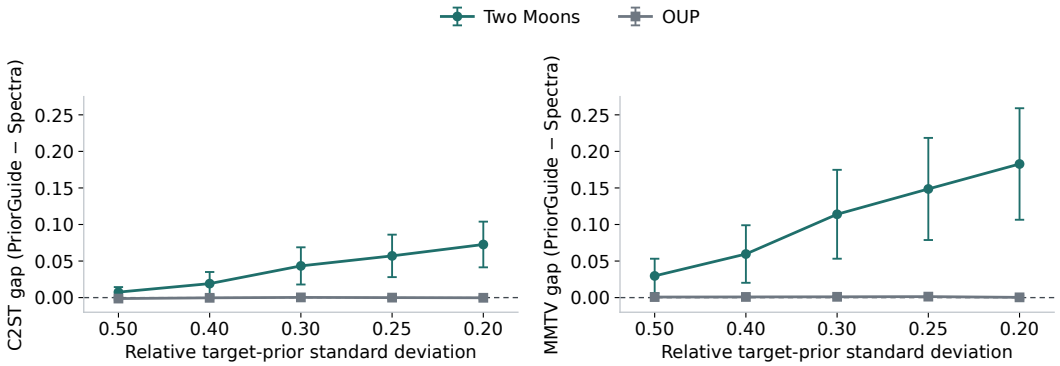


Figure 2: **Accuracy gap as the target prior narrows.** Mean paired PriorGuide–Spectra difference with 95% paired t confidence intervals across ten observations. All experimental factors except target-prior width are held fixed. Positive gaps indicate better performance by Spectra.

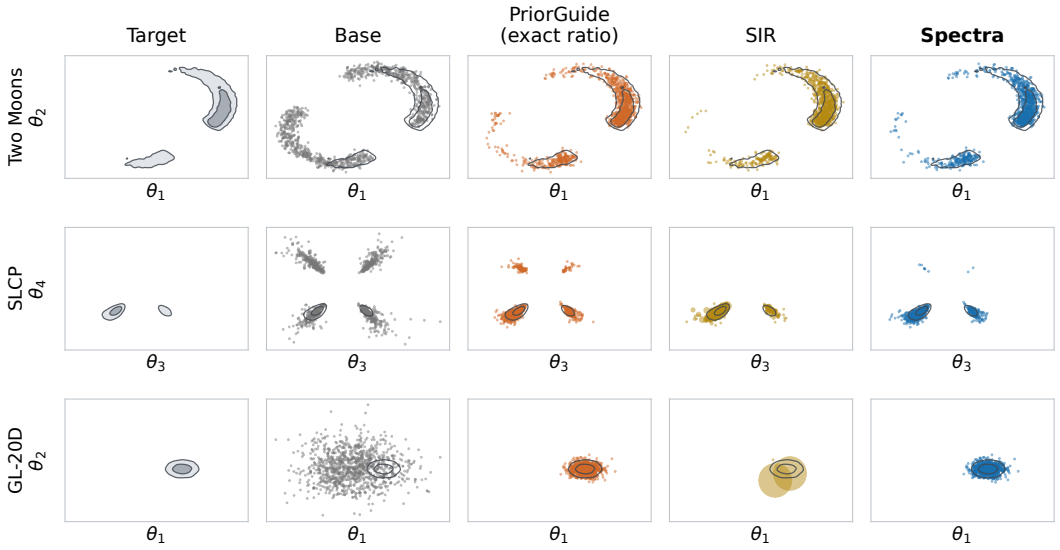


Figure 3: **Posterior geometry under strong single-factor prior shifts.** Reference contours enclose 50% and 90% posterior mass. Markers show 1,000 samples from each method for one fixed observation, backbone, and sampling seed per task. Larger SIR markers indicate repeated resampling of the same Base sample. SLCP and GL-20D show two-dimensional projections. Table 1 reports full-dimensional accuracy.

on Two Moons and SLCP (Table 1). For 162 of the 180 pairs, one component accounts for more than 95% of the posterior mass after observing x. In the remaining 18 pairs, both components retain at least 5% posterior mass, and the aggregate advantage over PriorGuide persists (Appendix L.2).

We also evaluate mixtures with more than two components on Two Moons (Appendix L.3). Posterior accuracy remains stable up to $K = 16$. At $K = 16$, five components retain at least 5% posterior mass for each of the three observations, so the posterior remains genuinely multi-component. For these larger mixtures, path-space weight estimates remain close to the corresponding reference weights. Full mixture results and weight-estimation diagnostics are reported in Appendix L.

4.4 Accuracy and Computational Cost

Spectra requires one transformed base-score evaluation per reverse step, whereas PriorGuide additionally differentiates through the frozen score model to compute its guidance. For timing, we implement PriorGuide's guidance with a vector–Jacobian product (VJP) instead of the full Jacobian used by the benchmarked implementation. Even with this faster implementation, Spectra reduces

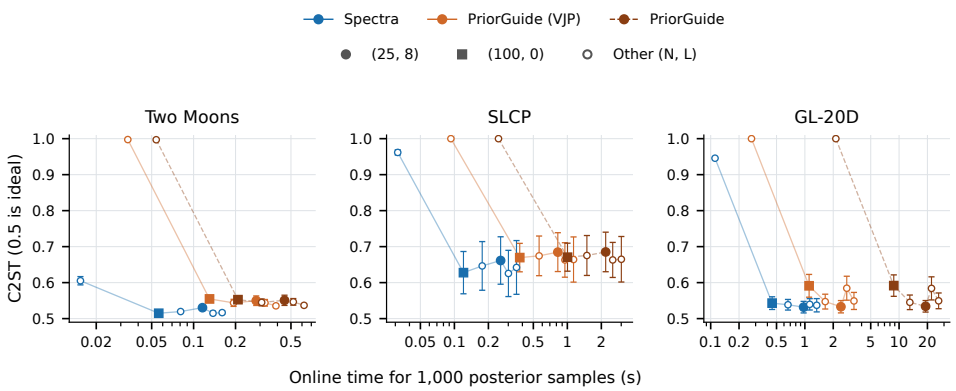


Figure 4: **Single-factor accuracy versus online sampling time under strong prior shifts.** PriorGuide (VJP) uses a vector–Jacobian product for the same guidance update and preserves PriorGuide's accuracy (Appendix M). Timing covers the reverse sampling stage for 1,000 posterior samples after compilation. One-time setup costs are excluded.

steady-state online sampling time by 2.4–3.2× across five timing tasks at the sampler settings of Table 1. Across the sampler budgets in Figure 4, increasing PriorGuide's sampling effort does not close the accuracy–cost gap. Detailed timing results and implementation comparisons are reported in Appendix M.

For mixture priors, fixing one component per trajectory keeps the subsequent sampling cost independent of the number of components. On Two Moons, the measured Spectra sampling time is unchanged from $K = 1$ to $K = 64$. The K-dependent computation is instead concentrated in observation-specific component-weight estimation, whose path-space cost grows linearly with K. Weight estimation can dominate the initial cost for a new observation, while additional samples for the same observation remain inexpensive. Detailed timing and scaling results are reported in Appendices M.2 and M.3.

5 Discussion

Spectra allows amortized diffusion-based SBI models to be reused when prior assumptions change, without repeating expensive simulation or training. It exploits structure in the known prior change and propagates its effect through the frozen score model, so a single pretrained model can support revised priors and prior-sensitivity analyses at test time. Composition extends the same principle to richer prior changes while keeping the per-step cost at one frozen-score evaluation.

Limitations. Spectra derives its exactness and low online sampling cost from structure in the prior change rather than offering unrestricted prior flexibility. The present theory assumes isotropic Gaussian corruption, and adaptation is limited to the support of the training prior. Broader prior changes, such as anisotropic or correlated Gaussians, require structured representations like the positive mixtures of Appendix N. Exactness concerns the transported score rather than the complete numerical sampler, which still inherits error from the learned score, terminal initialization, and finite integration. For multi-component representations, greater flexibility also introduces observation-specific weight estimation and an additional setup cost.

Conclusion. Spectra shows that changing prior assumptions need not require changing the pretrained inference model itself. When the prior update has suitable structure, its effect can be propagated analytically through a frozen diffusion-SBI model, with composition extending the same idea to richer prior changes. More broadly, this suggests that structured test-time transformations can extend the useful life of expensive amortized inference models as the scientific assumptions they encode are revised.

REPRODUCIBILITY STATEMENT

Derivations, proofs, and error analyses are in Appendices B–I; experimental protocols, reference validation, and full results are in Appendices J–N. The accompanying anonymized code package includes the scripts, fixed random seeds, model and task configurations, and experiment settings needed to reproduce the reported tables and figures. Unless explicitly stated otherwise, headline comparisons use the exact analytic prior ratio when available and the stated matched posterior-sampling protocols.

AI USE DISCLOSURE

Generative AI tools were used as interactive assistants under author direction for research brainstorming and methodological feedback; drafting and checking parts of mathematical derivations and proofs; supporting implementation, debugging, and experimental analysis; and manuscript drafting, editing, and LaTeX preparation. The scientific direction, methodological and experimental decisions, validation of mathematical and empirical results, and final presentation were determined by the authors. AI-assisted outputs were treated as provisional and reviewed by the authors, who take responsibility for all claims, results, and conclusions in this work.

REFERENCES

Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M. Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, 32(3):405–431, 2017. doi: 10.1214/17-STS611. URL https://doi.org/10.1214/17-STS611.

Benjamin Boys, Mark Girolami, Jakiw Pidstrigach, Sebastian Reich, Alan Mosca, and O. Deniz Akyildiz. Tweedie moment projected diffusions for inverse problems. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=4unJi0qrTE.

Gabriel Cardoso, Yazid Janati el idrissi, Sylvain Le Corff, and Eric Moulines. Monte Carlo guided denoising diffusion models for Bayesian linear inverse problems. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=nHESwXvxWK.

Paul Edmund Chang, Nasrulloh Ratu Bagus Satrio Loka, Daolang Huang, Ulpu Remes, Samuel Kaski, and Luigi Acerbi. Amortized probabilistic conditioning for optimization, simulation and inference. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 703–711, 2025. URL https://proceedings.mlr.press/v258/chang25a.html.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OnD9zGAGT0k.

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://doi.org/10.1073/pnas.1912789117.

Maximilian Dax, Stephen R. Green, Jonathan Gair, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Real-time gravitational wave science with neural posterior estimation. *Physical Review Letters*, 127:241103, 2021. doi: 10.1103/PhysRevLett.127.241103. URL https://doi.org/10.1103/PhysRevLett.127.241103.

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(8053):49–53, 2025. doi: 10.1038/s41586-025-08593-z. URL https://www.nature.com/articles/s41586-025-08593-z.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006. doi: 10.1111/j.1467-9868.2006.00553.x. URL https://doi.org/10.1111/j.1467-9868.2006.00553.x.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B. Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle:

Compositional generation with energy-based diffusion models and MCMC. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8489–8510, 2023. URL https://proceedings.mlr.press/v202/du23a.html.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. doi: 10.1198/jasa.2011.tm11181. URL https://doi.org/10.1198/jasa.2011.tm11181.

Lasse Elsemüller, Hans Olischläger, Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=Kxtpa9rvM0.

Tomas Geffner, George Papamakarios, and Andriy Mnih. Compositional score modeling for simulation-based inference. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11098–11116, 2023. URL https://proceedings.mlr.press/v202/geffner23a.html.

Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C. Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, and Martin Modrák. Bayesian workflow. *arXiv preprint arXiv:2011.01808*, 2020. doi: 10.48550/arXiv.2011.01808. URL https://arxiv.org/abs/2011.01808.

Manuel Gloeckler, Michael Deistler, Christian Dietrich Weilbach, Frank Wood, and Jakob H. Macke. All-in-one simulation-based inference. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15735–15766, 2024. URL https://proceedings.mlr.press/v235/gloeckler24a.html.

Pedro J. Gonçalves, Jan-Matthis Lueckmann, Michael Deistler, Marcel Nonnenmacher, Kaan Öcal, Giacomo Bassetto, Chaitanya Chintaluri, William F. Podlaski, Sara A. Haddad, Tim P. Vogels, David S. Greenberg, and Jakob H. Macke. Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife*, 9:e56261, 2020. doi: 10.7554/eLife.56261. URL https://elifesciences.org/articles/56261.

David S. Greenberg, Marcel Nonnenmacher, and Jakob H. Macke. Automatic posterior transformation for likelihood-free inference. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2404–2414, 2019. URL https://proceedings.mlr.press/v97/greenberg19a.html.

ChangHoon Hahn, Pablo Lemos, Liam Parker, Bruno Régaldo-Saint Blancard, Michael Eickenberg, Shirley Ho, Jiamin Hou, Elena Massara, Chirag Modi, Azadeh Moradinezhad Dizgah, and David Spergel. Cosmological constraints from non-Gaussian and nonlinear galaxy clustering using the SimBIG inference framework. *Nature Astronomy*, 8:1457–1467, 2024. doi: 10.1038/s41550-024-02344-2. URL https://www.nature.com/articles/s41550-024-02344-2.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Peter Holderrieth, Uriel Singer, Tommi Jaakkola, Ricky T. Q. Chen, Yaron Lipman, and Brian Karrer. GLASS flows: Efficient inference for reward alignment of flow and diffusion models. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/ad6ac52e02e14e0f2ddc0d4587ab47f1-Abstract-Conference.html.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025. doi: 10.1038/s41586-024-08328-6. URL https://www.nature.com/articles/s41586-024-08328-6.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005. URL https://www.jmlr.org/papers/v6/hyvarinen05a.html.

Konrad P. Körding, Ulrik Beierholm, Wei Ji Ma, Steven Quartz, Joshua B. Tenenbaum, and Ladan Shams. Causal inference in multisensory perception. *PLoS ONE*, 2(9):e943, 2007. doi: 10.1371/journal.pone.0000943. URL https://doi.org/10.1371/journal.pone.0000943.

Julia Linhart, Gabriel Victorino Cardoso, Alexandre Gramfort, Sylvain Le Corff, and Pedro L. C. Rodrigues. Diffusion posterior sampling for simulation-based inference in tall data settings. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=cdhfoS6Gyo.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJkXfE5xx.

Jan-Matthis Lueckmann, Jan Boelts, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. Benchmarking simulation-based inference. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 343–351, 2021. URL https://proceedings.mlr.press/v130/lueckmann21a.html.

Abbas Mammadov, Ozgur Kara, Kaan Oktay, Iskander Azangulov, Adil Kaan Akan, Hyungjin Chung, James Matthew Rehg, and Yee Whye Teh. Exact posterior score estimation for solving linear inverse problems. *arXiv preprint arXiv:2606.17048*, 2026. doi: 10.48550/arXiv.2606.17048. URL https://arxiv.org/abs/2606.17048.

Aayush Mishra, Daniel Habermann, Marvin Schmitt, Stefan T. Radev, and Paul-Christian Bürkner. Robust amortized Bayesian inference with self-consistency losses on unlabeled data. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0fd3d8093b3ba73d19b393a1326fdba7-Abstract-Conference.html.

Sarthak Mittal, Niels Leif Bracher, Guillaume Lajoie, Priyank Jaini, and Marcus Brubaker. Amortized in-context Bayesian posterior estimation. *arXiv preprint arXiv:2502.06601*, 2025. doi: 10.48550/arXiv.2502.06601. URL https://arxiv.org/abs/2502.06601.

Christopher J. Moore and Davide Gerosa. Population-informed priors in gravitational-wave astronomy. *Physical Review D*, 104(8):083008, 2021. doi: 10.1103/PhysRevD.104.083008. URL https://doi.org/10.1103/PhysRevD.104.083008.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do Bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=KSugKcbNf9.

George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 837–848, 2019. URL https://proceedings.mlr.press/v89/papamakarios19a.html.

Severi Rissanen, Markus Heinonen, and Arno Solin. Free hunch: Denoiser covariance estimation for diffusion models without extra costs. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/e0bc6dbcbcc957b2aeadb20c39ba7f05-Abstract-Conference.html.

Christian P. Robert and George Casella. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer, New York, NY, second edition, 2004. ISBN 978-0-387-21239-5. doi: 10.1007/978-1-4757-4145-2. URL https://doi.org/10.1007/978-1-4757-4145-2.

Donald B. Rubin. Using the SIR algorithm to simulate posterior distributions. In José M. Bernardo, Morris H. DeGroot, Dennis V. Lindley, and Adrian F. M. Smith (eds.), *Bayesian Statistics 3: Proceedings of the Third Valencia International Meeting*, pp. 395–402. Oxford University Press, Oxford, UK, 1988. ISBN 978-0-19-852220-1.

Marvin Schmitt, Valentin Pratz, Ullrich Köthe, Paul-Christian Bürkner, and Stefan T. Radev. Consistency models for scalable and fast simulation-based inference. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/e58026e2b2929108e1bd24cbfa1c8e4b-Abstract-Conference.html.

Louis Sharrock, Jack Simons, Song Liu, and Mark Beaumont. Sequential neural score estimation: Likelihood-free inference with conditional score based diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 44565–44602, 2024. URL https://proceedings.mlr.press/v235/sharrock24a.html.

Marta Skreta, Tara Akhound-Sadegh, Viktor Ohanesian, Roberto Bondesan, Alan Aspuru-Guzik, Arnaud Doucet, Rob Brekelmans, Alexander Tong, and Kirill Neklyudov. Feynman–Kac correctors in diffusion: Annealing, guidance, and product of experts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 55906–55949, 2025. URL https://proceedings.mlr.press/v267/skreta25a.html.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 2256–2265, 2015. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

G. E. Uhlenbeck and L. S. Ornstein. On the theory of the Brownian motion. *Physical Review*, 36(5):823–841, 1930. doi: 10.1103/PhysRev.36.823. URL https://doi.org/10.1103/PhysRev.36.823.

Julius Vetter, Manuel Gloeckler, Daniel Gedon, and Jakob H. Macke. Effortless, simulation-efficient Bayesian inference using tabular foundation models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 92164–92203, 2025. doi: 10.52202/085713-2770. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/770cabd044c4eacb6dc5924d9a686dce-Abstract-Conference.html.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142. URL https://doi.org/10.1162/NECO_a_00142.

Salvatore Vitale, Davide Gerosa, Carl-Johan Haster, Katerina Chatziioannou, and Aaron Zimmerman. Impact of Bayesian priors on the characterization of binary black hole coalescences. *Physical Review Letters*, 119:251103, 2017. doi: 10.1103/PhysRevLett.119.251103. URL https://doi.org/10.1103/PhysRevLett.119.251103.

Rachel C. M. Warnock, James F. Parham, Walter G. Joyce, Tyler R. Lyson, and Philip C. J. Donoghue. Calibration uncertainty in molecular dating analyses: there is no substitute for the prior evaluation of time priors. *Proceedings of the Royal Society B: Biological Sciences*, 282(1798):20141013, 2015. doi: 10.1098/rspb.2014.1013. URL https://doi.org/10.1098/rspb.2014.1013.

George Whittle, Juliusz Ziomek, Jacob Rawling, and Maike A. Osborne. Distribution transformers: Fast approximate Bayesian inference with on-the-fly prior adaptation. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2502.02463.

Jonas Wildberger, Maximilian Dax, Simon Buchholz, Stephen Green, Jakob H. Macke, and Bernhard Schölkopf. Flow matching for scalable simulation-based inference. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/3663ae53ec078860bb0b9c6606e092a0-Abstract-Conference.html.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-1363. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/63e8bc7bbf1cfea36d1d1b6538aecce5-Abstract-Conference.html.

Yang Yang, Severi Rissanen, Paul Edmund Chang, Nasrulloh Ratu Bagus Satrio Loka, Daolang Huang, Arno Solin, Markus Heinonen, and Luigi Acerbi. PriorGuide: Test-time prior adaptation for simulation-based inference. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=G4I23g5Ugh.

Yichen Zang, Song Liu, and Jiun-Yi Lin. Guidance for prior change via density ratio estimation. *arXiv preprint arXiv:2608.21729*, 2026. doi: 10.48550/arXiv.2608.21729. URL https://arxiv.org/abs/2608.21729.

APPENDIX

The appendix is organized as follows.

- Appendix A provides extended related work.
- Appendices B–F collect notation, derivations, and proofs.
- Appendices G–H cover error bounds and normalizer estimation, and Appendix I approximates correlated Gaussian priors by positive factors.
- Appendices J–N show experimental protocols and full results.

A EXTENDED RELATED WORK

Prior adaptation in amortized inference. PriorGuide is the closest method to our setting. It adapts a frozen diffusion-SBI model from a clean prior ratio using a Gaussian approximation of the reverse conditional (Yang et al., 2026), whereas Spectra exploits prior-ratio structure to obtain the adapted score directly from the base score. A contemporaneous approach instead learns the noisy ratio correction with a target-prior-specific auxiliary network, avoiding new simulator calls but requiring additional training (Zang et al., 2026). Other methods obtain prior flexibility by exposing the inference model to a family of priors or model alternatives during training (Chang et al., 2025; Whittle et al., 2026; Elsemüller et al., 2024). Automatic posterior transformation addresses proposal changes during sequential simulation and retraining (Greenberg et al., 2019), while likelihood-based SBI permits a new prior to be imposed after training at the cost of per-observation posterior sampling (Papamakarios et al., 2019). Related work also studies robustness of amortized inference through self-consistency objectives on unlabeled observations (Mishra et al., 2026). Our setting instead starts from a posterior model trained under a single prior and keeps it frozen during adaptation.

Conditioning and composition in diffusion models. Several diffusion methods construct approximate conditional scores for inverse problems, including pseudoinverse guidance, diffusion posterior sampling, and Tweedie moment-projected methods (Song et al., 2023; Chung et al., 2023; Boys et al., 2024). Free Hunch estimates local denoiser covariance using information from training data and the sampling trajectory (Rissanen et al., 2025). Exact Gaussian-conditioning identities can also lead to transformed model queries: GLASS combines Gaussian observations through a transformed input and noise level (Holderrieth et al., 2026), while EPS derives an exact posterior-score identity for linear Gaussian measurements (Mammadov et al., 2026). For a Gaussian prior-ratio factor, Spectra's transformed query is an identity of this kind. Spectra applies it to the prior ratio of a frozen amortized posterior model, extends it to linear tilts, factors that broaden the prior and positive mixtures of such factors, and shows that, among affine query maps with a scalar noise variance, isotropic exponential–quadratic factors are the only ones that admit exact transport.

Diffusion scores can be composed across observations or factors, although naive score addition may require corrected sampling (Geffner et al., 2023; Linhart et al., 2026; Du et al., 2023). Particle-based methods provide another route to conditional diffusion sampling (Wu et al., 2023; Skreta et al., 2025; Cardoso et al., 2024). In our mixture construction, the adapted posterior is instead represented as a mixture induced by the prior ratio, with component weights estimated once and one component kept fixed along each reverse trajectory.

B NOTATION AND ASSUMPTIONS

Density notation. In the main text, p and q denote exact Bayesian posteriors. In Appendix G, exact target and component posterior laws carry a star when they are compared with implemented laws. Although p is a Bayesian posterior in our application, the algebraic transport results only require it to be a base density satisfying the conditions below.

Support. We assume $\pi_{\mathrm{new}} \ll \pi_{\mathrm{tr}}$ and define $r = d\pi_{\mathrm{new}}/d\pi_{\mathrm{tr}}$. For priors with densities, this reduces to $r(\theta) = \pi_{\mathrm{new}}(\theta)/\pi_{\mathrm{tr}}(\theta)$ wherever $\pi_{\mathrm{tr}}(\theta) > 0$. If the target posterior instead assigns mass outside $S = \operatorname{supp} \pi_{\mathrm{tr}}$, any output law satisfying $\widehat{q}(S) = 1$ incurs $\mathrm{TV}(q, \widehat{q}) \geq q(S^c)$, by evaluating total variation on S^c.

Table 2: Principal notation. Symbols specific to error analysis and sampling protocols are defined where they are used.

Symbol	Meaning
$\pi_{\mathrm{tr}}, \pi_{\mathrm{new}}, r$	training prior, target prior, and their ratio
p, q	base and target posteriors
q^{ϕ}	posterior induced by a single prior-ratio factor
$t, \tau = \sigma(t)^2$	diffusion time and corresponding noise variance
$p_\tau, s_p, \widehat{s}_p$	Gaussian-smoothed base posterior, its exact score, and frozen score estimate
ϕ_k, b_k, q_k	prior-ratio factor, its coefficient, and posterior component
$Z_k, \widetilde{Z}_k$	normalizing constants under the exact posterior and finite base sampler
$\alpha_k, \widehat{\alpha}_k$	exact and estimated component weights
$D_\tau, m_\tau, \rho_\tau$	transport denominator, transformed state, and transformed noise variance

Regularity. The base p is a probability density on $\mathbb{R}^d$. The Gaussian convolutions used below are positive and differentiable, differentiation under the integral is valid, and each required normalizing constant is finite and nonzero. Kernel equalities hold almost everywhere with respect to Lebesgue measure.

Exactness. Exactness refers to the score identity obtained from the exact base score. Exact posterior sampling additionally requires the correct target terminal distribution and exact reverse dynamics reaching the clean endpoint. The implementation uses a learned score, a finite solver, and an approximate terminal distribution.

C Derivation of the noisy prior-reweighting identity

Fix an observation x and diffusion time t. Under the training prior, the noisy posterior is

$$p_t(z \mid x) = \int p_{t|0}(z \mid \theta) p(\theta \mid x)\, d\theta. \tag{17}$$

For a new prior π_{new}, let

$$r(\theta) = \frac{\pi_{\mathrm{new}}(\theta)}{\pi_{\mathrm{tr}}(\theta)}, \qquad q(\theta \mid x) = \frac{p(\theta \mid x) r(\theta)}{Z_r(x)},$$

where

$$Z_r(x) = \int p(\theta \mid x) r(\theta)\, d\theta. \tag{18}$$

Applying the same forward diffusion to the new posterior gives

$$\begin{aligned} q_t(z \mid x) &= \int p_{t|0}(z \mid \theta) q(\theta \mid x)\, d\theta \\ &= \frac{1}{Z_r(x)} \int p_{t|0}(z \mid \theta) p(\theta \mid x) r(\theta)\, d\theta. \end{aligned} \tag{19}$$

The reverse conditional under the base posterior follows from Bayes' rule,

$$p(\theta \mid z, x) = \frac{p_{t|0}(z \mid \theta) p(\theta \mid x)}{p_t(z \mid x)}. \tag{20}$$

Equivalently,

$$p_{t|0}(z \mid \theta) p(\theta \mid x) = p_t(z \mid x) p(\theta \mid z, x). \tag{21}$$

Substituting Eq. (21) into Eq. (19) yields

$$\begin{aligned} q_t(z \mid x) &= \frac{p_t(z \mid x)}{Z_r(x)} \int p(\theta \mid z, x) r(\theta)\, d\theta \\ &= \frac{p_t(z \mid x)}{Z_r(x)} \mathbb{E}_{p(\theta \mid z, x)}[r(\theta)]. \end{aligned} \tag{22}$$

Defining $H_t(z, x) = \mathbb{E}_{p(\theta \mid z, x)}[r(\theta)]$ therefore gives

$$q_t(z \mid x) = \frac{p_t(z \mid x) H_t(z, x)}{Z_r(x)}. \tag{23}$$

Since $Z_r(x)$ does not depend on z,

$$\begin{aligned} s_q(z,t,x) &= \nabla_z \log q_t(z \mid x) \\ &= \nabla_z \log p_t(z \mid x) + \nabla_z \log H_t(z,x) \\ &= s_p(z,t,x) + \nabla_z \log H_t(z,x), \end{aligned} \tag{24}$$

which gives Eq. (7).

D Exact single-factor transport

D.1 Proof of Proposition 1

Fix an observation x and a noise level $\tau > 0$ with $D_\tau = 1 + \kappa\tau > 0$. Let ϕ have the form in Eq. (8), and define

$$Z_\phi(x) = \int p(\theta \mid x)\phi(\theta)\, d\theta,$$

assuming $0 < Z_\phi(x) < \infty$. A constant scale in $r \propto \phi$ cancels in posterior normalization, so the Gaussian-smoothed target posterior is

$$q_\tau^\phi(z \mid x) = \frac{1}{Z_\phi(x)} \int p(\theta \mid x)\phi(\theta)\mathcal{N}(z;\theta,\tau I)\, d\theta. \tag{25}$$

We first rewrite the factor-kernel product inside the integral:

$$\begin{aligned} \phi(\theta)\mathcal{N}(z;\theta,\tau I) &= (2\pi\tau)^{-d/2} \exp\left(c + a^\top\theta - \frac{\kappa}{2}\|\theta\|^2 - \frac{\|z-\theta\|^2}{2\tau} \right) \\ &= (2\pi\tau)^{-d/2} \exp\left(c - \frac{\|z\|^2}{2\tau} - \frac{1}{2}\left(\tau^{-1}+\kappa\right)\|\theta\|^2 + \left(z/\tau + a\right)^\top \theta \right). \end{aligned} \tag{26}$$

To write the terms depending on θ as a Gaussian, set

$$\rho_\tau^{-1} = \tau^{-1} + \kappa, \qquad \rho_\tau^{-1} m_\tau(z) = z/\tau + a. \tag{27}$$

Since $D_\tau = 1 + \kappa\tau$, this gives

$$\rho_\tau = \frac{\tau}{D_\tau}, \qquad m_\tau(z) = \frac{z + \tau a}{D_\tau}. \tag{28}$$

Completing the square in θ gives

$$\phi(\theta)\mathcal{N}(z;\theta,\tau I) = (2\pi\tau)^{-d/2} \exp\left(-\frac{\|\theta - m_\tau(z)\|^2}{2\rho_\tau} + c + \frac{\|z+\tau a\|^2}{2\tau D_\tau} - \frac{\|z\|^2}{2\tau} \right). \tag{29}$$

Because

$$(2\pi\tau)^{-d/2} = D_\tau^{-d/2}(2\pi\rho_\tau)^{-d/2},$$

we obtain

$$\phi(\theta)\mathcal{N}(z;\theta,\tau I) = C_\tau(z)\mathcal{N}(m_\tau(z);\theta,\rho_\tau I), \tag{30}$$

$$\log C_\tau(z) = c - \frac{d}{2}\log D_\tau + \frac{\|z+\tau a\|^2}{2\tau D_\tau} - \frac{\|z\|^2}{2\tau}. \tag{31}$$

The Gaussian kernel is written with argument $m_\tau(z)$ and mean θ to match the base smoothing. Since $C_\tau(z)$ is independent of θ, substituting Eq. (30) into Eq. (25) gives

$$\begin{aligned} q_\tau^\phi(z \mid x) &= \frac{C_\tau(z)}{Z_\phi(x)} \int p(\theta \mid x)\mathcal{N}(m_\tau(z);\theta,\rho_\tau I)\, d\theta \\ &= \frac{C_\tau(z)}{Z_\phi(x)} p_{\rho_\tau}(m_\tau(z) \mid x). \end{aligned} \tag{32}$$

At fixed τ, ρ_τ is independent of z, while

$$\nabla_z m_\tau(z) = \frac{1}{D_\tau} I.$$

Moreover,

$$\nabla_z \log C_\tau(z) = \frac{z + \tau a}{\tau D_\tau} - \frac{z}{\tau} = \frac{a - \kappa z}{D_\tau}, \tag{33}$$

$$\nabla_z \log p_{\rho_\tau}(m_\tau(z) \mid x) = \frac{1}{D_\tau} s_p(m_\tau(z), \rho_\tau, x). \tag{34}$$

Taking the log gradient of Eq. (32) and using $\nabla_z \log Z_\phi(x) = 0$ therefore yields

$$s_{q^\phi}(z, \tau, x) = \frac{a - \kappa z}{D_\tau} + \frac{1}{D_\tau} s_p(m_\tau(z), \rho_\tau, x), \tag{35}$$

which proves Proposition 1. The derivation requires no Gaussian assumption on p and no derivative of its score.

D.2 Learned-score error and admissible noise levels

Define $e(z, \tau, x) = \widehat{s}_p(z, \tau, x) - s_p(z, \tau, x)$. Let $\widehat{s}_{q^\phi}$ denote the transported score obtained by replacing s_p with $\widehat{s}_p$ in Proposition 1. Subtracting the exact transported score cancels the shared closed-form term, giving

$$\widehat{s}_{q^\phi}(z, \tau, x) - s_{q^\phi}(z, \tau, x) = D_\tau^{-1} e(m_\tau(z), \rho_\tau, x). \tag{36}$$

For $\kappa \geq 0$, the multiplier D_τ^{-1} is at most one. For $\kappa < 0$, it exceeds one on the admissible range. A multiplier below one does not by itself imply smaller score error, because the base-score error is evaluated at the transformed query $(m_\tau(z), \rho_\tau)$, where its magnitude may differ.

The condition $D_\tau > 0$ is required to keep the transformed variance $\rho_\tau = \tau / D_\tau$ positive. For $\kappa \geq 0$, it holds at every positive noise level. For $\kappa < 0$, every noise variance used by the sampler must satisfy $\tau < -1/\kappa$. Finiteness of $Z_\phi(x)$ is a separate requirement. The transformed noise levels ρ_τ must also lie within the noise range supported by the learned score model, since clipping them back into that range changes the score identity.

D.3 Special cases and interpretation

For a linear exponential tilt, $\phi(\theta) = \exp(c + a^\top \theta)$, we have $\kappa = 0$, $D_\tau = 1$, $\rho_\tau = \tau$, and $m_\tau(z) = z + \tau a$. Hence

$$s_{q^\phi}(z, \tau, x) = a + s_p(z + \tau a, \tau, x). \tag{37}$$

The noise level is unchanged. The transport only shifts the base-score query by τa and adds the constant correction a.

For isotropic Gaussian localization, $\phi(\theta) = \mathcal{N}(\theta; \mu_\phi, \lambda I)$ with $\lambda > 0$, the terms depending on θ give $a = \mu_\phi / \lambda$ and $\kappa = 1/\lambda$. The remaining terms are absorbed into c. Substitution into Eq. (11) gives

$$s_{q^\phi}(z, \tau, x) = \frac{\mu_\phi - z}{\tau + \lambda} + \frac{\lambda}{\tau + \lambda} s_p\left(\frac{\lambda z + \tau \mu_\phi}{\tau + \lambda}, \frac{\tau \lambda}{\tau + \lambda}, x\right). \tag{38}$$

Here the transformed state lies between z and the localization center μ_ϕ, while the transformed noise variance $\tau\lambda/(\tau + \lambda)$ is smaller than τ.

E Formal characterization of exact transport

This section proves the characterization in Theorem 2. The question is which positive factors can be absorbed into an affine change of the Gaussian query state and a scalar change of its variance. All statements concern this explicit kernel identity.

E.1 Formal statement

Fix $\tau > 0$ and let $\phi : \mathbb{R}^d \to (0, \infty)$ be measurable. Suppose there exist a scalar $\rho > 0$, a matrix $A \in \mathbb{R}^{d \times d}$, a vector $u \in \mathbb{R}^d$, and a measurable function $C : \mathbb{R}^d \to (0, \infty)$ such that

$$\phi(\theta) \mathcal{N}(z; \theta, \tau I) = C(z) \mathcal{N}(Az + u; \theta, \rho I) \tag{39}$$

for almost every (z, θ) with respect to Lebesgue measure. At the fixed noise level τ, the quantities ρ, A, and u do not depend on z or θ. Then there exist constants $c \in \mathbb{R}$, $a \in \mathbb{R}^d$, and $\kappa \in \mathbb{R}$ with $1 + \kappa\tau > 0$ such that

$$\log \phi(\theta) = c + a^\top \theta - \frac{\kappa}{2}\|\theta\|^2 \tag{40}$$

for almost every θ. Moreover,

$$A = \frac{\rho}{\tau} I = \frac{1}{1+\kappa\tau} I, \qquad u = \rho a, \qquad \rho = \frac{\tau}{1+\kappa\tau}. \tag{41}$$

Conversely, every such ϕ satisfies Eq. (39) with these parameters and the prefactor in Eq. (31).

E.2 Necessity

Proof. Taking logarithms in Eq. (39) and isolating $\log \phi(\theta)$ gives

$$\log \phi(\theta) = \log C(z) + \frac{d}{2} \log \frac{\tau}{\rho} - \frac{1}{2\rho}\|\theta - Az - u\|^2 + \frac{1}{2\tau}\|\theta - z\|^2. \tag{42}$$

The term $d \log(\tau/\rho)/2$ comes from the two Gaussian normalizers. Expanding the squared norms gives

$$\begin{aligned} \log \phi(\theta) = & -\frac{1}{2}\left(\frac{1}{\rho} - \frac{1}{\tau}\right)\|\theta\|^2 + \theta^\top \left[\left(\frac{A}{\rho} - \frac{I}{\tau}\right) z + \frac{u}{\rho}\right] \\ & + \underbrace{\log C(z) + \frac{d}{2}\log\frac{\tau}{\rho} - \frac{\|Az+u\|^2}{2\rho} + \frac{\|z\|^2}{2\tau}}_{F(z)}. \end{aligned} \tag{43}$$

The left side is independent of z. To make the corresponding coefficient comparison rigorous under the almost-everywhere identity, Fubini's theorem gives a full-measure set of z values for which Eq. (43) holds for almost every θ. For any two such values of z, subtracting the two identities gives a polynomial in θ that vanishes almost everywhere. Hence all of its coefficients are zero. The coefficients of Eq. (43) must therefore be independent of z almost everywhere.

In particular, there is a constant vector $a \in \mathbb{R}^d$ such that

$$\left(\frac{A}{\rho} - \frac{I}{\tau}\right) z + \frac{u}{\rho} = a \qquad \text{for almost every } z. \tag{44}$$

An affine function of z can be constant almost everywhere only if its linear part vanishes. Hence

$$\frac{A}{\rho} - \frac{I}{\tau} = 0, \qquad A = \frac{\rho}{\tau} I, \qquad a = \frac{u}{\rho}. \tag{45}$$

The quadratic coefficient is already independent of z and has the required isotropic form. Define

$$\kappa = \frac{1}{\rho} - \frac{1}{\tau}. \tag{46}$$

Then the quadratic term is $-\kappa\|\theta\|^2/2$. The remaining term $F(z)$ must likewise equal a constant c almost everywhere. Therefore Eq. (43) reduces to

$$\log \phi(\theta) = c + a^\top \theta - \frac{\kappa}{2}\|\theta\|^2 \tag{47}$$

almost everywhere, proving the necessary form of ϕ. The definition of κ also determines the noise transform:

$$1 + \kappa\tau = \frac{\tau}{\rho} > 0, \qquad \rho = \frac{\tau}{1+\kappa\tau}, \qquad Az + u = \frac{z + \tau a}{1+\kappa\tau}. \tag{48}$$

Using $F(z) = c$ and solving for $C(z)$ gives

$$\begin{aligned} \log C(z) &= c - \frac{d}{2}\log\frac{\tau}{\rho} + \frac{\|Az+u\|^2}{2\rho} - \frac{\|z\|^2}{2\tau} \\ &= c - \frac{d}{2}\log(1+\kappa\tau) + \frac{\|z+\tau a\|^2}{2\tau(1+\kappa\tau)} - \frac{\|z\|^2}{2\tau}. \end{aligned} \tag{49}$$

This is exactly Eq. (31). The remaining dependence on z is independent of θ and is absorbed into the prefactor $C(z)$. □

E.3 Sufficiency and scope

Conversely, take $\phi(\theta) = \exp(c + a^\top\theta - \kappa\|\theta\|^2/2)$ with $D_\tau = 1 + \kappa\tau > 0$. Set $\rho = \tau/D_\tau$, $A = I/D_\tau$, $u = \rho a$, and $C = C_\tau$. Then

$$Az + u = \frac{z + \tau a}{D_\tau} = m_\tau(z),$$

so Eq. (30) is exactly Eq. (39). This proves sufficiency and completes Theorem 2.

The necessity result already holds at a single fixed $\tau > 0$: the kernel identity need not hold over an interval of noise levels to force the exponential-quadratic form. Across a diffusion schedule, the factor ϕ is fixed, so its coefficients (a, κ, c) do not depend on τ. Only the induced kernel transform changes with the noise level:

$$A_\tau = \frac{1}{1 + \kappa\tau} I, \qquad u_\tau = \frac{\tau a}{1 + \kappa\tau}, \qquad \rho_\tau = \frac{\tau}{1 + \kappa\tau}.$$

The identity therefore applies along a schedule provided $D_\tau = 1 + \kappa\tau > 0$ at every queried noise level. When $\kappa = 0$, these reduce to $\rho_\tau = \tau$, $A_\tau = I$, and $u_\tau = \tau a$.

Because the identity holds at the kernel level, integrating it against any base posterior that satisfies the regularity conditions of Appendix B gives the same score transport. The characterization concerns affine state maps with scalar isotropic variance as specified in Eq. (39). It makes no statement about algorithms that do not admit this kernel equality.

F Static mixtures and dynamic responsibilities

Fix x and a finite positive representation $r = \sum_{k=1}^K b_k\phi_k$ with $b_k > 0$ and $0 < Z_k(x) < \infty$. The following calculations distinguish the observation-specific component weights $\alpha_k(x)$ from the noisy-state-dependent component responsibilities $\gamma_{k,\tau}(z, x)$.

F.1 Normalizing the positive sum

Recall that $q_k(\theta \mid x) = p(\theta \mid x)\phi_k(\theta)/Z_k(x)$. Each component is normalized because

$$\int q_k(\theta \mid x)\, d\theta = \frac{1}{Z_k(x)} \int p(\theta \mid x)\phi_k(\theta)\, d\theta = 1. \tag{50}$$

Substitution into the unnormalized target gives

$$\begin{aligned} p(\theta \mid x)r(\theta) &= p(\theta \mid x) \sum_k b_k\phi_k(\theta) \\ &= \sum_k b_k Z_k(x) q_k(\theta \mid x). \end{aligned} \tag{51}$$

Integrating both sides with respect to θ determines the total normalizer:

$$Z_r(x) = \int p(\theta \mid x)r(\theta)\, d\theta = \sum_k b_k Z_k(x) \int q_k(\theta \mid x)\, d\theta = \sum_k b_k Z_k(x). \tag{52}$$

Dividing by $Z_r(x)$ gives

$$q(\theta \mid x) = \sum_k \alpha_k(x) q_k(\theta \mid x), \qquad \alpha_k(x) = \frac{b_k Z_k(x)}{Z_r(x)}. \tag{53}$$

The weights are positive and sum to one. Their dependence on x comes from the expectations $Z_k(x)$ under the base posterior. This proves Eq. (14).

F.2 Why the same weights persist under Gaussian smoothing

The noisy component density is $q_{k,\tau}(z \mid x) = \int \mathcal{N}(z; \theta, \tau I) q_k(\theta \mid x)\, d\theta$. Since the mixture is finite, we can interchange its sum and the smoothing integral:

$$\begin{aligned} q_\tau(z \mid x) &= \int \mathcal{N}(z; \theta, \tau I) \sum_k \alpha_k(x) q_k(\theta \mid x)\, d\theta \\ &= \sum_k \alpha_k(x) \int \mathcal{N}(z; \theta, \tau I) q_k(\theta \mid x)\, d\theta \\ &= \sum_k \alpha_k(x) q_{k,\tau}(z \mid x). \end{aligned} \tag{54}$$

Each $q_{k,\tau}$ is normalized, so this is a mixture with the same weights, not an unnormalized sum requiring new weights at each noise level.

Introduce a latent label J with $\Pr(J = k \mid x) = \alpha_k(x)$ and draw $\theta \mid J = k, x$ from q_k. After applying Gaussian corruption to θ, the joint density is

$$\Pr(J = k, z_\tau \in dz \mid x) = \alpha_k(x) q_{k,\tau}(z \mid x)\, dz. \tag{55}$$

Marginalizing z leaves $\Pr(J = k \mid x) = \alpha_k(x)$ at every noise level. Conditioning on a particular z, however, gives

$$\Pr(J = k \mid z_\tau = z, x) = \frac{\alpha_k(x) q_{k,\tau}(z \mid x)}{q_\tau(z \mid x)} = \gamma_{k,\tau}(z, x). \tag{56}$$

Here $\alpha_k(x)$ are the static component weights, whereas $\gamma_{k,\tau}(z, x)$ are the noisy-state-dependent component responsibilities.

F.3 Correctness of sampling the component once

For any measurable set $B \subseteq \mathbb{R}^d$, the two-stage construction in Eq. (15) satisfies

$$\begin{aligned} \Pr(\theta \in B \mid x) &= \sum_k \Pr(J = k \mid x) \Pr(\theta \in B \mid J = k, x) \\ &= \sum_k \alpha_k(x) \int_B q_k(\theta \mid x)\, d\theta = \int_B q(\theta \mid x)\, d\theta. \end{aligned} \tag{57}$$

Hence sampling one label followed by one draw from its component produces the target mixture.

Reverse diffusion supplies the conditional component draw. At a terminal noise level $\tau_T = \sigma(t_{\max})^2$ (Appendix J.3), the ideal construction draws J with probabilities $\alpha(x)$ and then initializes z_T from $q_{J,\tau_T}(\cdot \mid x)$. Conditional on $J = k$, exact reverse dynamics driven by s_{q_k} recover $q_{k,\tau}$ at each noise level and q_k at the clean endpoint. Since the component label is drawn with probability $\alpha_k(x)$ and then kept fixed, marginalizing over J recovers $q_\tau = \sum_k \alpha_k(x) q_{k,\tau}$ throughout the reverse process, and hence q at the endpoint. Proposition 1 supplies s_{q_k} from one transformed base-score query.

The exact mixture algebra also identifies what the implementation produces. Let $\widehat{q}_k$ be the endpoint law of the finite component sampler, with fresh sampling randomness independent of the weight estimates. If labels are drawn with estimated weights $\widehat{\alpha}$, then, conditional on those estimates, the output law is exactly

$$\widehat{q} = \sum_k \widehat{\alpha}_k \widehat{q}_k. \tag{58}$$

The approximation to the Bayesian target enters through the weights and component endpoint laws, not through the act of drawing a static label. Appendix G quantifies their contributions separately.

F.4 Deriving the marginal mixture score

By the definition of the score, $\nabla_z q_{k,\tau} = q_{k,\tau} s_{q_k}$. Differentiating the noisy density mixture therefore gives

$$\begin{aligned} s_q(z,\tau,x) &= \frac{\nabla_z q_\tau(z \mid x)}{q_\tau(z \mid x)} \\ &= \frac{\sum_k \alpha_k(x) \nabla_z q_{k,\tau}(z \mid x)}{q_\tau(z \mid x)} \\ &= \sum_k \frac{\alpha_k(x) q_{k,\tau}(z \mid x)}{q_\tau(z \mid x)} s_{q_k}(z,\tau,x) \\ &= \sum_k \gamma_{k,\tau}(z,x) s_{q_k}(z,\tau,x). \end{aligned} \tag{59}$$

Together with Eq. (56), this gives the marginal mixture score. It also shows why replacing $\gamma_{k,\tau}(z,x)$ by the static weights $\alpha_k(x)$ would generally give a different score.

To expose the density information required by γ, apply Eq. (32) to component k:

$$q_{k,\tau}(z \mid x) = \frac{C_{k,\tau}(z)}{Z_k(x)} p_{\rho_{k,\tau}}\big(m_{k,\tau}(z) \mid x\big). \tag{60}$$

Multiplying by $\alpha_k(x) = b_k Z_k(x)/Z_r(x)$ cancels $Z_k(x)$:

$$\alpha_k(x) q_{k,\tau}(z \mid x) = \frac{1}{Z_r(x)} \underbrace{b_k C_{k,\tau}(z) p_{\rho_{k,\tau}}\big(m_{k,\tau}(z) \mid x\big)}_{U_{k,\tau}(z,x)}. \tag{61}$$

The common $Z_r(x)$ also cancels when normalizing over components, so

$$\gamma_{k,\tau}(z,x) = \frac{U_{k,\tau}(z,x)}{\sum_j U_{j,\tau}(z,x)}. \tag{62}$$

Although the normalizing constants $Z_k(x)$ cancel, evaluating $\gamma_{k,\tau}$ still requires normalized base-density values at different transformed states and, in general, different noise levels. For two components,

$$\begin{aligned} \log \frac{U_{k,\tau}(z,x)}{U_{j,\tau}(z,x)} = \log \frac{b_k C_{k,\tau}(z)}{b_j C_{j,\tau}(z)} \\ + \log p_{\rho_{k,\tau}}\big(m_{k,\tau}(z) \mid x\big) - \log p_{\rho_{j,\tau}}\big(m_{j,\tau}(z) \mid x\big). \end{aligned} \tag{63}$$

The first term is available analytically from the factor parameters. The remaining terms require log-density values at the transformed queries. A score evaluation provides only the local gradient $\nabla \log p_\rho$, not the corresponding log-density value. Evaluating the dynamic responsibilities would therefore require additional machinery to recover or estimate the relevant relative densities. Static component sampling avoids this step: it requires only the observation-level weights $\alpha_k(x)$ and keeps the selected component fixed throughout the reverse trajectory.

F.5 Factor scaling and reuse across prior coefficients

A positive rescaling $\phi'_k = \zeta_k \phi_k$, accompanied by $b'_k = b_k/\zeta_k$, leaves the represented ratio unchanged. It gives $Z'_k = \zeta_k Z_k$ and

$$q'_k = \frac{p\,\zeta_k \phi_k}{\zeta_k Z_k} = q_k, \qquad b'_k Z'_k = b_k Z_k. \tag{64}$$

Neither the component distribution nor its component weight depends on the factor-scaling convention. Any fixed factor scale may be absorbed into b_k.

For a fixed observation and unchanged factors, $Z_k(x) = \mathbb{E}_{p(\theta|x)}[\phi_k(\theta)]$ does not depend on b_k. If only the coefficient vector changes, the new component weights are

$$\alpha'_k(x) = \frac{b'_k Z_k(x)}{\sum_j b'_j Z_j(x)}, \tag{65}$$

while the component score transforms remain unchanged. The same reuse holds for estimated normalizers when the observation, component set, and weight-estimation protocol are held fixed. A new observation generally requires new normalizers. If the supplied representation approximates r by $\widetilde{r}$, all mixture identities in this section apply to the posterior induced by $\widetilde{r}$. Its discrepancy from the intended target is handled in Appendix G.

F.6 Constructing factors and coefficients from a known prior mixture

The coefficients b_k belong to the supplied prior-ratio representation. They are distinct from the observation-dependent component weights $\alpha_k(x)$. Suppose the target prior has the known mixture representation

$$\pi_{\mathrm{new}}(\theta) = \sum_k \omega_k \nu_k(\theta), \qquad \omega_k > 0, \qquad \sum_k \omega_k = 1. \tag{66}$$

On the support of the training prior, division by π_{tr} gives

$$r(\theta) = \sum_k \omega_k \frac{\nu_k(\theta)}{\pi_{\mathrm{tr}}(\theta)}. \tag{67}$$

If each ratio ν_k/π_{tr} has the form in Eq. (8), we may take

$$\phi_k(\theta) = \frac{\nu_k(\theta)}{\pi_{\mathrm{tr}}(\theta)}, \qquad b_k = \omega_k. \tag{68}$$

This is an analytic construction, requiring no posterior samples or observation-specific estimation. A mixture of arbitrary prior densities need not satisfy this factor condition.

For two Gaussian priors, $\pi_{\mathrm{tr}} = \mathcal{N}(\mu_0, \Sigma_0)$ and $\nu = \mathcal{N}(\mu_1, \Sigma_1)$, their log ratio has the form

$$\begin{aligned} \log \frac{\nu(\theta)}{\pi_{\mathrm{tr}}(\theta)} &= c + \left(\Sigma_1^{-1}\mu_1 - \Sigma_0^{-1}\mu_0\right)^\top \theta \\ &\quad - \frac{1}{2}\theta^\top \left(\Sigma_1^{-1} - \Sigma_0^{-1}\right)\theta, \end{aligned} \tag{69}$$

where c is independent of θ. Hence this ratio has the single-factor form in Eq. (8) exactly when

$$\Sigma_1^{-1} - \Sigma_0^{-1} = \kappa I. \tag{70}$$

The means may therefore change arbitrarily. The restriction is that the change in quadratic curvature is isotropic. The isotropic Gaussian case below is a common special case.

Let $\pi_{\mathrm{tr}} = \mathcal{N}(\mu_0, v_0 I)$ and $\nu_k = \mathcal{N}(\mu_k, v_k I)$, with positive scalar variances. Expanding their log ratio gives

$$\begin{aligned} \log \frac{\nu_k(\theta)}{\pi_{\mathrm{tr}}(\theta)} &= \frac{d}{2}\log\frac{v_0}{v_k} - \frac{\|\theta-\mu_k\|^2}{2v_k} + \frac{\|\theta-\mu_0\|^2}{2v_0} \\ &= c_k + a_k^\top\theta - \frac{\kappa_k}{2}\|\theta\|^2, \end{aligned} \tag{71}$$

where

$$\kappa_k = v_k^{-1} - v_0^{-1}, \qquad a_k = \mu_k/v_k - \mu_0/v_0, \tag{72}$$

$$c_k = \frac{d}{2}\log\frac{v_0}{v_k} - \frac{\|\mu_k\|^2}{2v_k} + \frac{\|\mu_0\|^2}{2v_0}. \tag{73}$$

If the constant c_k is retained in ϕ_k, then $b_k = \omega_k$. The implementation instead uses a normalized Gaussian factor when $\kappa_k > 0$. Completing the square gives

$$\frac{\nu_k(\theta)}{\pi_{\mathrm{tr}}(\theta)} = \Gamma_k \mathcal{N}\left(\theta; \frac{a_k}{\kappa_k}, \kappa_k^{-1} I\right), \tag{74}$$

with

$$\Gamma_k = \left(\frac{2\pi}{\kappa_k}\right)^{d/2} \exp\left(c_k + \frac{\|a_k\|^2}{2\kappa_k}\right). \tag{75}$$

Under this convention, ϕ_k is the normalized Gaussian and $b_k = \omega_k\Gamma_k$. In general the coefficient differs from the prior mixture weight ω_k, but it is still determined analytically from the supplied

priors. For $\kappa_k \leq 0$, the exponential-quadratic form remains the relevant representation rather than a normalized Gaussian factor, subject to posterior normalizability and the sampling-schedule condition.

For a uniform training prior on a box B of volume $|B|$, $\pi_{\mathrm{tr}}(\theta) = |B|^{-1}$ inside B. Hence

$$r(\theta) = |B|\, \pi_{\mathrm{new}}(\theta), \qquad \theta \in B.$$

If the target prior is a mixture of isotropic Gaussians restricted to B and normalized as a whole, both $|B|$ and the truncation normalizer are common to all components. They therefore cancel in posterior normalization, so the Gaussian kernels can be used directly as ϕ_k with their supplied mixture coefficients as b_k, up to a common scale. This representation holds only on B and does not extend the training support. Appendix I gives the corresponding construction when the Gaussian mixture itself is obtained by approximation.

G End-to-end error decomposition

G.1 Mixture error bound

Fix an observation and let r be the intended prior ratio and $\widetilde{r}$ its positive-factor approximation. The exact target is $q^{r,\star}$. The positive-factor approximation induces

$$q^{\widetilde{r},\star} = \sum_k \alpha_k^\star q_k^\star, \qquad \alpha_k^\star(x) = \frac{b_k Z_k(x)}{\sum_j b_j Z_j(x)},$$

where $q_k^\star$ denotes the exact posterior component induced by ϕ_k. For a fixed weight-estimation protocol, let $\mathbb{P}_0^{(R)}$ be its base endpoint law after R transitions (Appendix H). Assume its factor expectations are finite and positive, and define

$$\widetilde{Z}_k(x) = \mathbb{E}_{\mathbb{P}_0^{(R)}}[\phi_k(\theta)], \qquad \alpha_k^P(x) = \frac{b_k \widetilde{Z}_k(x)}{\sum_j b_j \widetilde{Z}_j(x)}. \tag{76}$$

Let $\widehat{\alpha}$ be the estimated weights and $\widehat{q}_k$ the endpoint law of the final component sampler. The final component sampler may use a different discretization and corrector configuration from the estimator's proposal paths. Conditional on the weight estimates, the output is $\widehat{q} = \sum_k \widehat{\alpha}_k \widehat{q}_k$, and

$$\begin{aligned} \mathrm{TV}(q^{r,\star}, \widehat{q}) \leq\ & \underbrace{\mathrm{TV}(q^{r,\star}, q^{\widetilde{r},\star})}_{\text{prior-ratio approximation}} + \underbrace{\tfrac{1}{2}\|\widehat{\alpha} - \alpha^\star\|_1}_{\text{component weights}} \\ & + \underbrace{\sum_k \widehat{\alpha}_k\, \mathrm{TV}(q_k^\star, \widehat{q}_k)}_{\text{component sampling}}, \end{aligned} \tag{77}$$

where

$$\|\widehat{\alpha} - \alpha^\star\|_1 \leq \underbrace{\|\widehat{\alpha} - \alpha^P\|_1}_{\text{weight estimator}} + \underbrace{\|\alpha^P - \alpha^\star\|_1}_{\text{base-sampler discrepancy}}. \tag{78}$$

The effective sample size (ESS) is an overlap diagnostic for the estimator, not a bound on either weight-error term. Assessing the base-sampler discrepancy requires an exact or independently validated posterior reference.

Insert the target induced by the approximated prior ratio and apply the triangle inequality:

$$\mathrm{TV}(q^{r,\star}, \widehat{q}) \leq \mathrm{TV}(q^{r,\star}, q^{\widetilde{r},\star}) + \mathrm{TV}(q^{\widetilde{r},\star}, \widehat{q}). \tag{79}$$

For the second term, insert the intermediate mixture $\bar{q} = \sum_k \widehat{\alpha}_k q_k^\star$:

$$\mathrm{TV}(q^{\widetilde{r},\star}, \widehat{q}) \leq \mathrm{TV}\left(\sum_k \alpha_k^\star q_k^\star, \sum_k \widehat{\alpha}_k q_k^\star\right) + \mathrm{TV}\left(\sum_k \widehat{\alpha}_k q_k^\star, \sum_k \widehat{\alpha}_k \widehat{q}_k\right) \tag{80}$$

$$\leq \frac{1}{2}\|\widehat{\alpha} - \alpha^\star\|_1 + \sum_k \widehat{\alpha}_k\, \mathrm{TV}(q_k^\star, \widehat{q}_k). \tag{81}$$

Equation (78) is the triangle inequality on the simplex.

G.2 Binary component-weight sensitivity

For $q_\alpha = \alpha q_1 + (1-\alpha)q_2$,

$$q_{\widehat{\alpha}} - q_\alpha = (\widehat{\alpha} - \alpha)(q_1 - q_2). \tag{82}$$

Taking half the L^1 norm gives

$$\mathrm{TV}(q_{\widehat{\alpha}}, q_\alpha) = |\widehat{\alpha} - \alpha|\ \mathrm{TV}(q_1, q_2). \tag{83}$$

A given absolute weight error therefore matters most when the component distributions are well separated and has little effect when they strongly overlap. If

$$\delta = \log \frac{\widehat{\alpha}}{1-\widehat{\alpha}} - \log \frac{\alpha}{1-\alpha}$$

is the log-odds error, then the logistic map has derivative at most $1/4$, so

$$|\widehat{\alpha} - \alpha| \le |\delta|/4. \tag{84}$$

For small δ, the first-order response is $|\widehat{\alpha} - \alpha| \approx \alpha(1-\alpha)|\delta|$, so for a small log-odds error the weight itself is most sensitive near $\alpha = 1/2$.

G.3 Approximate-ratio posterior TV

Let $q^{r,\star} = pr/Z_r$ and $q^{\widetilde{r},\star} = p\widetilde{r}/Z_{\widetilde{r}}$ for a fixed observation, with $0 < Z_r, Z_{\widetilde{r}} < \infty$, and define $\varepsilon = \mathbb{E}_p|r - \widetilde{r}| < \infty$. Since $|Z_r - Z_{\widetilde{r}}| \le \varepsilon$,

$$\|q^{r,\star} - q^{\widetilde{r},\star}\|_1 \le \frac{\mathbb{E}_p|r-\widetilde{r}|}{Z_r} + Z_{\widetilde{r}}\left|\frac{1}{Z_r} - \frac{1}{Z_{\widetilde{r}}}\right| \tag{85}$$

$$\le \frac{2\varepsilon}{Z_r}. \tag{86}$$

Therefore $\mathrm{TV}(q^{r,\star}, q^{\widetilde{r},\star}) \le \varepsilon/Z_r$. The expectation defining ε must be under the observation-specific base posterior. An approximation error measured only under the training prior does not by itself control every observation-specific target posterior.

H Normalizer-estimation details

The path-space estimator estimates the finite-chain normalizers $\widetilde{Z}_k(x)$ by a change of measure on reverse trajectories. We first derive the path-space identity and its overlap diagnostic, then describe the alternative direct estimator, and finally summarize the complete inference procedure. Sampling budgets and aggregation are specified in Appendix J.3.

H.1 Finite-chain path-space identity

Let R denote the number of transitions in the normalizer estimator, independently of the output sampler's (N, L) configuration. Write the estimator time grid as $t_R > \cdots > t_0$, with $\tau_n = \sigma(t_n)^2$, and a reverse trajectory as

$$\omega = (z_R, z_{R-1}, \ldots, z_0).$$

Suppressing the fixed observation, the estimator's base and component proposal path laws are

$$\mathbb{P}^{(R)}(d\omega) = p_{\mathrm{term}}(z_R) \prod_{n=0}^{R-1} P_n(z_n \mid z_{n+1})\, d\omega, \tag{87}$$

$$\mathbb{Q}_k^{(R)}(d\omega) = q_{k,\mathrm{term}}(z_R) \prod_{n=0}^{R-1} Q_{k,n}(z_n \mid z_{n+1})\, d\omega. \tag{88}$$

Here p_{term} and $q_{k,\mathrm{term}}$ denote the terminal distributions used by the two chains. When the terminal distributions and Markov transitions have densities with common support, the trajectory likelihood ratio is

$$\log \frac{d\mathbb{P}^{(R)}}{d\mathbb{Q}_k^{(R)}} = \log \frac{p_{\mathrm{term}}(z_R)}{q_{k,\mathrm{term}}(z_R)} + \sum_{n=0}^{R-1} \log \frac{P_n(z_n \mid z_{n+1})}{Q_{k,n}(z_n \mid z_{n+1})}. \tag{89}$$

If the two chains use the same terminal distribution, the first term is identically zero. Equation (89) is the formal version of the trajectory likelihood ratio P/Q_k used in Section 3.6.

Change of measure then gives

$$\widetilde{Z}_k(x) = \int \phi_k(z_0)\mathbb{P}^{(R)}(d\omega) = \mathbb{E}_{\mathbb{Q}_k^{(R)}}\left[\phi_k(z_0)\frac{d\mathbb{P}^{(R)}}{d\mathbb{Q}_k^{(R)}}\right]. \tag{90}$$

Trajectories can be drawn from the component proposal $\mathbb{Q}_k^{(R)}$, and the likelihood ratio corrects the expectation back to the base path law $\mathbb{P}^{(R)}$. This is Eq. (16) with the trajectory density ratio written in measure notation.

Implemented transitions. Both path laws use Gaussian Euler–Maruyama transitions with the same covariance and differ only in their score-driven means. For the transition from t_{n+1} to t_n, define

$$\Delta_n = (t_{n+1} - t_n)\, g(t_{n+1})^2, \qquad g(t)^2 = 2\sigma(t)\dot{\sigma}(t). \tag{91}$$

An Euler–Maruyama step of Eq. (3) for the base chain is

$$z_n = z_{n+1} + \Delta_n\, \widehat{s}_p(z_{n+1}, \tau_{n+1}, x) + \sqrt{\Delta_n}\,\epsilon_n, \qquad \epsilon_n \sim \mathcal{N}(0, I). \tag{92}$$

Hence P_n is Gaussian with mean

$$z_{n+1} + \Delta_n \widehat{s}_p(z_{n+1}, \tau_{n+1}, x)$$

and covariance $\Delta_n I$.

Let $\widehat{s}_{q_k}$ denote the transported component-score estimate obtained from Eq. (11) by replacing s_p with $\widehat{s}_p$. The component proposal $Q_{k,n}$ uses the same time grid and covariance, with mean

$$z_{n+1} + \Delta_n \widehat{s}_{q_k}(z_{n+1}, \tau_{n+1}, x).$$

For a transition generated under $Q_{k,n}$,

$$z_n = z_{n+1} + \Delta_n \widehat{s}_{q_k}(z_{n+1}, \tau_{n+1}, x) + \sqrt{\Delta_n}\,\epsilon_n, \tag{93}$$

the corresponding log-density ratio is

$$\log \frac{P_n(z_n \mid z_{n+1})}{Q_{k,n}(z_n \mid z_{n+1})} = \sqrt{\Delta_n}\,\epsilon_n^\top (\widehat{s}_p - \widehat{s}_{q_k}) - \frac{\Delta_n}{2}\,\|\widehat{s}_p - \widehat{s}_{q_k}\|^2, \tag{94}$$

where both scores are evaluated at (z_{n+1}, τ_{n+1}, x). The normalizer estimator is defined on this corrector-free chain. Langevin corrector updates are used only by the final output sampler.

Score evaluations per transition. Computing the proposal transition requires $\widehat{s}_{q_k}(z_{n+1}, \tau_{n+1}, x)$, which uses one frozen-model query at the transformed state and noise level

$$\left(m_{k,\tau_{n+1}}(z_{n+1}), \rho_{k,\tau_{n+1}}\right).$$

Evaluating the base transition density in Eq. (94) additionally requires $\widehat{s}_p(z_{n+1}, \tau_{n+1}, x)$. The implemented path-space estimator therefore uses two frozen-model score evaluations per transition. This additional evaluation applies only to weight estimation. Final posterior sampling uses one frozen-model query per transported-score evaluation, as reported in Appendix M.

To characterize the overlap relevant to this importance sampler, define the endpoint-tilted path law

$$\mathbb{P}_k^{\phi}(d\omega) = \frac{\phi_k(z_0)}{\widetilde{Z}_k(x)}\mathbb{P}^{(R)}(d\omega). \tag{95}$$

The normalized importance weight is

$$\bar{w}_k = \frac{d\mathbb{P}_k^{\phi}}{d\mathbb{Q}_k^{(R)}}.$$

For probability laws $P \ll Q$, define the chi-squared divergence as

$$\chi^2(P\|Q) = \mathbb{E}_Q\left[\left(\frac{dP}{dQ} - 1\right)^2\right]. \tag{96}$$

Since $\mathbb{E}_{\mathbb{Q}_k^{(R)}}[\bar{w}_k] = 1$,

$$\mathbb{E}_{\mathbb{Q}_k^{(R)}}[\bar{w}_k^2] = 1 + \chi^2\left(\mathbb{P}_k^{\phi} \| \mathbb{Q}_k^{(R)}\right). \tag{97}$$

For independent proposal trajectories $\omega_1, \ldots, \omega_{n_{\mathrm{P}}} \sim \mathbb{Q}_k^{(R)}$, the empirical effective sample size is $\mathrm{ESS}_k = (\sum_i \bar{w}_k(\omega_i))^2 / \sum_i \bar{w}_k(\omega_i)^2$. With a finite second moment, $\mathrm{ESS}_k / n_{\mathrm{P}}$ therefore converges to the reciprocal of this quantity (Agapiou et al., 2017). A larger ESS indicates that the proposal $\mathbb{Q}_k^{(R)}$ has better overlap with the endpoint-tilted path law and that the importance weights are less concentrated. This diagnostic concerns the finite-chain importance-sampling problem. It does not measure the discrepancy between the base endpoint law $\mathbb{P}_0^{(R)}$ and the exact Bayesian posterior $p(\theta \mid x)$. The proposal path law used for normalizer estimation may also differ from the path law used for final posterior sampling.

H.2 Direct estimation and overlap

Given independent samples θ_n from the fixed base-sampling protocol, Spectra-Direct estimates the corresponding factor expectation by

$$\widehat{Z}_k^{\mathrm{dir}} = \frac{1}{n_{\mathrm{B}}} \sum_{n=1}^{n_{\mathrm{B}}} \phi_k(\theta_n). \tag{98}$$

Here n_{B} is the bank size. Once the bank has been sampled, it can be reused for all K factors at $O(n_{\mathrm{B}}K)$ factor-evaluation cost. The estimate can be unstable when the bank poorly covers regions where ϕ_k is large.

For the base samples in Eq. (98), write $\phi_{kn} = \phi_k(\theta_n)$. The usual effective-sample-size diagnostic (Agapiou et al., 2017) is

$$\mathrm{ESS}_k^{\mathrm{dir}} = \frac{\left(\sum_n \phi_{kn}\right)^2}{\sum_n \phi_{kn}^2}. \tag{99}$$

It equals n_{B} when all samples contribute equally and approaches one when a single sample dominates. A low ESS therefore indicates that the estimate is dominated by only a small part of the bank. A high ESS means that the observed contributions are spread more evenly across the sampled points, which is favorable for stability. However, ESS only describes the samples that are actually present in the bank. It cannot detect an important region that the entire bank failed to sample, and it does not measure the discrepancy between the base-sampling endpoint distribution and the exact Bayesian posterior.

H.3 Complete Spectra inference procedure

Algorithm 1 Spectra inference for one observation

Input: frozen score $\widehat{s}_p$, observation x, supplied positive representation $\{b_k, \phi_k\}_{k=1}^K$ of r or $\widetilde{r}$, output count M, and estimator and sampler settings.

Check $b_k > 0$ and the factor form in Eq. (8). At every noise level used by the output sampler, and by the normalizer estimator when $K > 1$, check $D_{k,\tau} > 0$ and that $\rho_{k,\tau}$ lies in the noise range supported by the learned score model.
if $K = 1$: set $\widehat{\alpha}_1 = 1$.
else: estimate $\widehat{Z}_k$ using the chosen normalizer estimator and set $\widehat{\alpha}_k = \dfrac{b_k \widehat{Z}_k}{\sum_j b_j \widehat{Z}_j}$.
for $m = 1, \ldots, M$:
  Draw J once with $\Pr(J = k) = \widehat{\alpha}_k$.
  Draw z_T from the shared base terminal Gaussian approximation in Eq. (110).
  for each reverse step:
    Advance the configured reverse sampler using component J's transported score estimate, obtained from Eq. (11) with s_p replaced by $\widehat{s}_p$ whenever a score evaluation is required.
  Store the endpoint as $\theta^{(m)}$.
return $\{\theta^{(m)}\}_{m=1}^M$.

I APPROXIMATING CORRELATED GAUSSIANS WITH EXPONENTIAL–QUADRATIC FACTORS

A correlated Gaussian prior change generally does not satisfy the single-factor condition in Appendix F.6. This section gives one concrete way to obtain a positive-factor approximation in that setting. We first write the correlated Gaussian exactly as a continuous mixture of isotropic Gaussian kernels, then replace the continuous mixture by a finite positive quadrature. Under the training priors considered below, the resulting kernels induce transportable exponential–quadratic factors. Appendix N uses this representation.

I.1 CONTINUOUS MIXTURE AND POSITIVE QUADRATURE

Let $0 < \lambda < \lambda_{\min}(\Sigma)$ and $U \sim \mathcal{N}(\mu, \Sigma - \lambda I)$. Adding independent $\mathcal{N}(0, \lambda I)$ noise to U gives $\mathcal{N}(\mu, \Sigma)$, hence

$$\mathcal{N}(\theta; \mu, \Sigma) = \mathbb{E}_U[\mathcal{N}(\theta; U, \lambda I)]\,. \tag{100}$$

Eq. (100) writes the correlated Gaussian exactly as a continuous mixture of isotropic Gaussian kernels with common covariance λI.

The remaining task is to replace the continuous distribution over centers U by finitely many centers with positive weights. Write

$$\Sigma - \lambda I = V \operatorname{diag}(s_1^2, \dots, s_d^2) V^\top.$$

Then

$$U = \mu + V \operatorname{diag}(s_i)\xi, \qquad \xi \sim \mathcal{N}(0, I).$$

For each coordinate i, choose a positive Gauss–Hermite rule with nodes $\xi_{i,j}$ and weights $\eta_{i,j}$ for the standard-normal expectation, normalized so that $\sum_j \eta_{i,j} = 1$. The tensor node indexed by $k = (j_1, \dots, j_d)$ has

$$u_k = \mu + V \operatorname{diag}(s_i)(\xi_{1,j_1}, \dots, \xi_{d,j_d})^\top, \qquad \omega_k = \prod_{i=1}^{d} \eta_{i,j_i}. \tag{101}$$

Every ω_k is positive and $\sum_k \omega_k = 1$. Replacing the continuous expectation by this quadrature gives

$$\widetilde{\pi}_{\text{new}}(\theta) = \sum_k \omega_k \mathcal{N}(\theta; u_k, \lambda I). \tag{102}$$

Equation (100) is exact. The finite quadrature in Eq. (102) is the approximation.

The implementation uses the probabilists' Gauss–Hermite rule, whose raw weights integrate against $e^{-\xi^2/2}$. It normalizes the weights along each coordinate before forming the tensor products in Eq. (101). The quadrature order along axis i is

$$n_i = \min\left(n_{\max}, \left\lfloor h \frac{s_i}{\sqrt{\lambda}} \right\rfloor + 2\right), \qquad K = \prod_i n_i, \tag{103}$$

where $h > 0$ controls the node density and $n_{\max}$ caps the per-axis order. Wider principal axes receive more nodes. For $n_{\max} \geq 2$ every axis receives at least two nodes, so $K \geq 2^d$ grows exponentially with dimension, and path-space weight estimation needs separate proposal trajectories for each component. We use this construction only in low-dimensional demonstrations.

The bandwidth, node orders, centers, and weights depend only on the specified prior and construction settings, not on the observation or posterior samples.

I.2 FROM PRIOR QUADRATURE TO PRIOR-RATIO COEFFICIENTS

The quadrature above approximates the target prior, but Spectra requires a representation of the prior ratio. Dividing by the training prior gives

$$\widetilde{r}(\theta) = \sum_k \omega_k \frac{\mathcal{N}(\theta; u_k, \lambda I)}{\pi_{\text{tr}}(\theta)}. \tag{104}$$

For the uniform training prior used in the Two Moons example, π_{tr} is constant inside its support B. Up to the common scale discussed in Appendix F.6, the implementation therefore uses

$$b_k = \omega_k, \qquad \phi_k(\theta) = \mathcal{N}(\theta; u_k, \lambda I), \tag{105}$$

with exponential–quadratic parameters

$$\kappa_k = \lambda^{-1}, \qquad a_k = u_k/\lambda, \qquad c_k = -\frac{\|u_k\|^2}{2\lambda} - \frac{d}{2}\log(2\pi\lambda). \tag{106}$$

If the Gaussian mixture is restricted to B and normalized as a whole, the additional normalization constant is common to all factors and therefore cancels from the component weights. No observation-specific fitting of b_k is performed.

For an isotropic Gaussian training prior, division by π_{tr} instead follows the Gaussian-ratio construction in Appendix F.6, with $\mu_k = u_k$ and $v_k = \lambda$. An arbitrary training prior need not preserve the isotropic factor family under division. Appendix G separates the resulting prior-ratio approximation error from weight-estimation and component-sampling errors.

J Experimental protocols and reference validation

This appendix specifies the benchmark tasks, prior shifts, evaluation and sampling protocols, posterior references, and PriorGuide baselines used in the experiments. Results are reported in Appendices K–N.

J.1 Tasks and prior shifts

For all six tasks in Section 4, the target posterior is either available analytically or approximated by a reference procedure validated in Appendix J.4. We follow the prior-shift construction of PriorGuide (Yang et al., 2026). For Gaussian training priors, the marginal standard deviation of the target prior is 0.5 times that of the training prior for the mild shift and 0.2 times that of the training prior for the strong shift. For uniform training boxes, we apply the same multipliers to the standard deviation of each uniform marginal.

Two Moons and SLCP use uniform training priors on $[-1, 1]^d$, with $d = 2$ and $d = 5$, respectively. For SLCP, this corresponds to rescaling the standard box $[-3, 3]^5$ by $1/3$. BCI uses a five-dimensional standard Gaussian, while the Gaussian Linear tasks use $\mathcal{N}(0, 0.1I)$ in 10 or 20 dimensions. OUP uses a uniform prior on $[-1, 1]^2$ after transforming its physical parameter box $[0, 2] \times [-2, 2]$.

For each task, we train three Simformer backbones with training seeds 0, 1, and 2 using the default architecture and training setup from the released PriorGuide code. Each backbone is a six-layer transformer with four attention heads and uses the variance-exploding schedule of Appendix J.3. Training uses 10,000 simulations from the training prior, with 5% held out for validation, a batch size of 1,000, and 30,000 Adam steps. SLCP, which the released PriorGuide code does not include, uses the same setup. No run triggered early stopping, so we use the final checkpoint in every case. Across the six tasks, this gives 18 backbones, and every experiment uses backbones from this set.

For each task and shift level, the single-factor benchmark uses one target prior shared across ten test observations. For a training box $[l, h]$, a target with standard deviation s has its center drawn uniformly from $[l + 3s, h - 3s]$ coordinatewise. For a Gaussian training prior with mean μ and marginal scale s_{tr}, the center is drawn coordinatewise from $[\mu - 2s_{\mathrm{tr}} + 3s, \mu + 2s_{\mathrm{tr}} - 3s]$. The mild and strong settings use different center draws and separately simulated observations, so they differ in more than prior width. Appendix K.4 isolates the effect of width in a separate fixed-center sweep. Target priors are generated with the released PriorGuide prior generator using a fixed seed. SLCP follows the same uniform-prior recipe. Within each setting, all methods use the same prior, observation, and frozen backbone.

For each task, the two-component benchmark uses ten mixture priors from the same generator. Each prior contains two Gaussian components with the strong-shift width, and their centers follow the single-factor rule above. The first mixture coefficient is drawn uniformly from $[0.2, 0.8]$, with the second set to its complement. For each mixture prior, we draw three parameter values from that prior and simulate one observation from each, yielding 30 prior–observation pairs per task.

J.2 METRICS AND STATISTICAL AGGREGATION

For each single-factor task and shift, we evaluate ten simulated observations with three independently trained backbones. Base and the guided samplers use three sampling seeds for each observation–backbone pair. We first average over sampling seeds, then over backbones within each observation, and finally summarize across the ten observations. Mixture results use the same hierarchy with 30 prior–observation pairs per task. Because three observations are generated from each mixture prior, these 30 pairs are not 30 independent prior draws. SIR uses one base-sample bank and one resampling draw for each observation–backbone or pair–backbone combination.

C2ST (Lopez-Paz & Oquab, 2017) assesses joint agreement between method and reference samples, with ideal value 0.5. MMTV averages total variation distances across one-dimensional marginals, with ideal value zero. Both use the metric implementations from the released PriorGuide code. For C2ST, a random-forest classifier with five-fold cross-validation distinguishes 1,000 method samples from 1,000 samples drawn without replacement from the 10,000-sample reference bank. We repeat this over five reference subsamples and average the classification accuracy. MMTV compares Gaussian kernel density estimates of each marginal of the method samples and the full reference bank. MMTV is undefined when a sample set has at most two distinct points. This occurs only for SIR on the Gaussian Linear tasks. We also report the root-mean-square error (RMSE) to the generating parameter as a supplementary diagnostic. Because a multimodal posterior need not concentrate around the generating parameter, RMSE is not used as a distributional accuracy metric.

J.3 SAMPLING AND WEIGHT-ESTIMATION SETTINGS

Final posterior sampling. The main benchmark uses $(N, L) = (25, 8)$ for PriorGuide and Spectra, except for single-factor BCI at $(250, 0)$. Base and SIR use $(100, 0)$. Each method returns 1,000 posterior samples. The controlled comparisons in Appendices K and L use $(100, 0)$, while the cost sweep in Appendix M evaluates six sampler configurations. The $(100, 0)$ and $(25, 8)$ settings differ in both the time grid and the number of corrector updates.

Weight-estimation budgets. For the systematic mixture benchmarks, weight estimates are fixed across the two output-sampling configurations. Path-space estimation uses 4,096 trajectories per component on a 400-point power grid and three estimator seeds. The path-space estimator and the final posterior samplers use the range-preserving grid

$$t_i = t_{\min} + (t_{\max} - t_{\min})u_i^2, \qquad u_i = \frac{i}{R-1}, \qquad i = 0, \dots, R-1, \tag{107}$$

which preserves both endpoints while placing more points at low noise. All backbones use the same variance-exploding schedule,

$$\sigma(t) = \sigma_{\min} \left(\frac{\sigma_{\max}}{\sigma_{\min}} \right)^t,$$

with $\sigma_{\min} = 10^{-4}$, $\sigma_{\max} = 15$, $t_{\min} = 10^{-5}$, and $t_{\max} = 1$.

The direct estimator uses three independent base banks of 4,000 samples, each generated with a 500-point uniform reverse grid. Neither weight estimator uses Langevin corrector updates. For path-space estimation, the proposal transitions enter the likelihood ratio in Eq. (89). Corrector updates are used only by the final output sampler when enabled. Each estimator seed s produces one log-normalizer estimate $\log \widehat{Z}_{k,s}$ per component. We average these values across the three estimator seeds and compute

$$\overline{\log \widehat{Z}_k} = \frac{1}{S} \sum_{s=1}^{S} \log \widehat{Z}_{k,s}, \qquad \widehat{\alpha} = \mathrm{softmax}\left(\log b + \overline{\log \widehat{Z}} \right), \qquad S = 3. \tag{108}$$

We use this aggregation for all component counts.

Transformed noise levels. Because σ is log-linear in t, the transformed noise level $\rho_{k,\tau}$ in Eq. (10) corresponds to

$$t_\rho = t - \frac{\log D_{k,\tau}}{2 \log(\sigma_{\max}/\sigma_{\min})}. \tag{109}$$

We query the frozen model directly at t_ρ without clipping. Clipping would change the queried noise level and invalidate the identity in Eq. (11). In all tasks, shifts, and sampler configurations reported here, t_ρ remains within $[t_{\min}, t_{\max}]$.

Terminal initialization. All Spectra components and matched guided baselines use the frozen base model's terminal Gaussian. Let μ_{train} and v_{train} denote the empirical mean and coordinatewise variances of the training parameters in model coordinates. At the largest noise level $\tau_T = \sigma(t_{\max})^2$, we initialize

$$z_T \sim \mathcal{N}\left(\mu_{\text{train}}, \operatorname{diag}(v_{\text{train}}) + \tau_T I\right). \tag{110}$$

For a fixed backbone, this initialization is shared across observations and component labels. The selected component changes only the score used during reverse sampling.

The ideal component sampler in Appendix F would instead initialize from $q_{k,\tau_T}(\cdot \mid x)$. The shared Gaussian in Eq. (110) is a high-noise approximation to this component-specific terminal distribution and is generally not identical to it at finite τ_T. This approximation is separate from the exact score identity and contributes to the component-sampling error in Appendix G, together with learned-score and numerical integration errors.

For the path-space estimator, the base and component chains use the same initialization, so the terminal-density term in Eq. (89) is identically zero in our experiments. Appendix H retains this term for the general case of different terminal distributions.

J.4 Posterior references and support checks

Posterior references are analytic for the Gaussian Linear tasks and constructed numerically for the remaining tasks. Table 3 summarizes each construction and its validation. Every reference bank contains 10,000 samples. SLCP and BCI use tempered sequential Monte Carlo (SMC), following Del Moral et al. (2006). For BCI, lower Gauss–Hermite orders measurably change the posterior, whereas orders 301 and 601 agree within the tested target region.

Table 3: **Posterior references and validation.** Reported validation statistics are worst-case values over the evaluated single-factor observations; BCI also includes the mixture pairs. Cross-run C2ST compares independent reference runs of the same posterior (ideal value 0.5).

Task	Reference	Validation
GL-10D/20D	closed-form posterior	analytic
Two Moons	exact-likelihood quadrature on a 4096×4096 grid	grid refinement changes the log normalizer by at most 2×10^{-4} nats
OUP	exact-likelihood quadrature on a 4096×4096 grid	grid refinement changes the log normalizer by less than 10^{-12} nats; target mass on the grid boundary at most 4×10^{-10} (both shifts)
SLCP	tempered SMC, 40,000 particles, four independent runs	cross-run C2ST at most 0.509 (strong) and 0.521 (mild)
BCI	tempered SMC, 8,000 particles, four independent runs; Gauss–Hermite likelihood of order 301	cross-run C2ST at most 0.509 (both shifts and mixtures); quadrature-order check

Reference component weights. For the two-component benchmark, reference weights are obtained by averaging the responsibilities $b_k\phi_k(\theta)/\sum_j b_j\phi_j(\theta)$ under the target-posterior reference. For Two Moons and OUP, we evaluate this expectation on the quadrature grid restricted to the training box. Because the base posterior is supported there, this gives the weights in Eq. (14). For SLCP and BCI, we average over their SMC reference samples. The Gaussian Linear weights are available in closed form from the conjugate posterior. Appendices L.3 and N describe the reference weights used in the remaining mixture studies.

Spectra variants. The mixture results in Appendices L and N report Spectra with three sources of component weights. Spectra-PS, the method in the main text, uses the path-space estimator of Appendix H, and Spectra-Direct uses the direct estimator of Appendix H.2. Spectra-Ref uses the

reference weights above. It needs the reference posterior and is therefore not a practical method, but it separates weight-estimation error from the rest of the sampler.

Reference support. For tasks with uniform training support, observation parameters are drawn from the target prior restricted to the training box. Two Moons and OUP truncate each Gaussian component separately while retaining the mixture coefficients, whereas SLCP truncates the mixture as a whole. The two constructions coincide for a single component, and all component centers lie at least three component standard deviations inside the box. Spectra does not extend the support of the training prior.

The benchmark metrics nevertheless use the original untruncated Gaussian reference. To check the effect of this mismatch, we reevaluate the single-factor Two Moons, SLCP, and OUP experiments against references restricted to the training support. Across Base, SIR, PriorGuide, and Spectra, all 12 C2ST/MMTV method orderings are unchanged, and the largest change in an aggregate metric is 0.0013. At least 98.69% of every reference bank lies inside the training support. This support mismatch therefore does not affect the reported method comparisons in these settings.

J.5 PRIORGUIDE PROTOCOLS

Exact-ratio comparison. The main comparison supplies PriorGuide with the same exact analytic prior ratio as Spectra, bypassing Gaussian mixture model fitting wherever it would otherwise be required. The Gaussian reverse-conditional approximation remains part of PriorGuide's guidance. Both methods use the sampling settings in Appendix J.3, following Yang et al. (2026, Appendix C.3.2): $(25, 8)$ for the uniform-prior and Gaussian Linear tasks and $(250, 0)$ for BCI. We use the BCI setting for single-factor shifts only, and BCI mixtures use $(25, 8)$. This removes prior-ratio fitting error from the comparison while retaining the difference between the two score updates.

Langevin update convention. All matched guided methods use the corrector implemented in the released PriorGuide code. For a positive step magnitude Δt,

$$\delta = \frac{\eta g(t)^2 \Delta t}{2}, \qquad z \leftarrow z + \delta \widehat{s}_q(z, t, x) + \sqrt{2\delta + 10^{-8}}\, \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \quad \eta = 0.5,$$

where $g(t)^2 = 2\sigma(t)\sigma'(t)$. PriorGuide's Eq. (A1) gives a step half this size under the same convention. We follow the executable update for all methods. For Spectra, $g(t)$ is evaluated at the current sampler time t, while the score itself is queried at the transformed noise level.

Vector–Jacobian-product implementation. PriorGuide's guidance uses the denoiser Jacobian only through a product with a vector, so the same update can be evaluated with a reverse-mode vector–Jacobian product (VJP) instead of explicitly forming the full parameter–parameter Jacobian. We use this implementation, denoted PriorGuide (VJP), as the strengthened timing baseline in Appendix M. It changes only the implementation of the guidance update.

At $(N, L) = (25, 8)$, a paired validation on Two Moons, SLCP, and GL-20D gives absolute mean C2ST differences of at most 5×10^{-4} and MMTV differences below 10^{-4} between PriorGuide (VJP) and the benchmarked implementation, with all 95% paired intervals containing zero. Accuracy tables therefore use the benchmarked implementation, while timing comparisons include the stronger VJP implementation. PG-FullCov does not admit the same reduction because its Tweedie covariance requires the full Jacobian.

J.6 PRIORGUIDE-FULLCOV

PriorGuide-FullCov (PG-FullCov) is an additional moment-matched baseline constructed in this work, rather than a variant proposed by Yang et al. (2026). It tests whether PriorGuide's scalar covariance approximation explains the difference from exact transport. We retain a Gaussian approximation to the reverse conditional but replace its scalar covariance with a moment-consistent full covariance, following the Tweedie moment-projection approach of Boys et al. (2024).

Moment identities. For fixed x, suppress the observation argument and write $\mu_{p,\tau}(z) = \mathbb{E}_p[\theta \mid z, x]$. The Gaussian-channel moment identities underlying Tweedie's formula (Efron, 2011, Section 2)

give

$$\mu_{p,\tau}(z) = z + \tau s_p(z,\tau), \qquad \Sigma_{p,\tau}(z) = \mathrm{Cov}_p(\theta \mid z, x) = \tau \nabla_z \mu_{p,\tau}(z) = \tau I + \tau^2 \nabla_z s_p(z,\tau). \tag{111}$$

These are exact moment identities for the exact base score. They do not imply that the reverse conditional itself is Gaussian.

In the implementation, we substitute the learned score and repair the resulting covariance before using it in the Gaussian surrogate. Let $J = \nabla_z \mu_{p,\tau}(z)$ denote the learned denoiser Jacobian on the parameter block, so that $\Sigma_{p,\tau} = \tau J$. We symmetrize J and floor its eigenvalues,

$$J_{\mathrm{sym}} = \tfrac{1}{2}(J + J^\top) = U \,\mathrm{diag}(\lambda) U^\top, \qquad J_{\mathrm{psd}} = U \,\mathrm{diag}(\max(\lambda, \varepsilon))\, U^\top, \qquad \varepsilon = 10^{-6}, \tag{112}$$

and use $C = \tau J_{\mathrm{psd}}$ as the local covariance.

Gaussian twist. Let m and C denote the resulting local mean and covariance. For a supplied ratio

$$r(\theta) \propto \sum_k w_k \mathcal{N}(\theta; \mu_k, \Sigma_k), \qquad w_k > 0,$$

multiplying the surrogate $\mathcal{N}(\theta; m, C)$ by component k gives

$$\widetilde{\mu}_k = m + C(C + \Sigma_k)^{-1}(\mu_k - m), \qquad \ell_k = \mathcal{N}(\mu_k; m, C + \Sigma_k). \tag{113}$$

The corresponding twisted mean and guidance are

$$\beta_k = \frac{w_k \ell_k}{\sum_j w_j \ell_j}, \qquad \widetilde{m}_q = \sum_k \beta_k \widetilde{\mu}_k, \qquad g_{\mathrm{FC}} = \frac{\widetilde{m}_q - m}{\tau}. \tag{114}$$

Here β_k are noisy-state-dependent weights within the Gaussian surrogate, not Spectra's observation-level component weights. The guidance substitutes this Gaussian-product mean into the identity $s_q - s_p = (m_q - m_p)/\tau$.

For a single isotropic Gaussian ratio, this construction reduces, before covariance repair, to the TMPD update of Boys et al. (2024, Eq. (11)) with an identity observation operator under VE corruption. Computing C requires a score Jacobian at every guidance evaluation, including corrector updates when enabled. Both PriorGuide and PG-FullCov receive the exact analytic prior ratio in these controlled comparisons.

K SINGLE-FACTOR RESULTS

This appendix provides supporting single-factor results, including a controlled $(N, L) = (100, 0)$ comparison with PG-FullCov, paired and overlap diagnostics, the practical sampling configuration, and the fixed-center severity sweep behind Figure 2.

K.1 CONTROLLED COMPARISON WITHOUT CORRECTOR UPDATES

Table 4 compares all six tasks at $(N, L) = (100, 0)$, without Langevin corrector updates. Under strong shifts, Spectra improves on PG-FullCov on Two Moons and SLCP. The two methods are similar on OUP and BCI, while PG-FullCov has a small C2ST advantage on the Gaussian Linear controls. Under mild shifts, Spectra is again better on Two Moons, whereas PG-FullCov attains slightly lower C2ST on BCI and both Gaussian Linear tasks, by at most 0.008 (Table 5). Appendix M reports the computational cost of this additional baseline.

K.2 PAIRED COMPARISONS AND OVERLAP DIAGNOSTICS

Table 5 reports observation-level paired C2ST differences for all six tasks under the controlled configuration of Table 4.

Clean-space overlap. For SIR, we measure overlap using the effective sample size $(\sum_i w_i)^2 / \sum_i w_i^2$ of the prior-ratio weights $w_i = r(\theta_i)$ on the 20,000-sample Base bank, normalized by the bank size. Under strong shifts, the median ESS fractions across the 30 observation–backbone pairs are approximately 45%, 6.0%, and 1.8% on Two Moons, SLCP, and BCI, but only 8×10^{-5} and 5×10^{-5} on GL-10D and GL-20D. The latter correspond to roughly two effective samples or fewer from the full bank. Under mild shifts, the corresponding fractions are 79%, 42%, and 34% on the first three tasks, compared with 1.9% and 0.14% on the Gaussian Linear tasks. The collapse in overlap on the Gaussian Linear tasks is consistent with the severe SIR degeneration in Table 4.

Table 4: **Single-factor comparison at $(N, L) = (100, 0)$.** Mean (standard deviation) across ten observations, after averaging sampling seeds within each backbone and then averaging the three backbones within each observation. C2ST is ideal at 0.5 and MMTV at zero. Bold indicates the displayed mean closest to the ideal value within each task and shift, including rounded ties. RMSE to the generating parameter is a supplementary diagnostic and is not bolded.

		Mild			Strong		
Task	Method	C2ST	MMTV	RMSE	C2ST	MMTV	RMSE
Two Moons	Base	0.568(0.036)	0.219(0.087)	0.397(0.197)	0.691(0.095)	0.426(0.192)	0.174(0.106)
	SIR	0.525(0.005)	0.075(0.011)	0.340(0.147)	0.544(0.018)	0.070(0.012)	0.100(0.061)
	PriorGuide (exact ratio)	0.522(0.010)	0.105(0.040)	0.371(0.173)	0.554(0.032)	0.167(0.085)	0.136(0.063)
	PG-FullCov	0.528(0.015)	0.123(0.057)	0.379(0.180)	0.552(0.038)	0.159(0.093)	0.142(0.070)
	Spectra	**0.512**(0.006)	**0.072**(0.007)	0.346(0.151)	**0.519**(0.009)	**0.068**(0.012)	0.102(0.059)
SLCP	Base	0.765(0.057)	0.255(0.041)	0.324(0.089)	0.917(0.019)	0.466(0.072)	0.312(0.060)
	SIR	0.640(0.086)	0.123(0.054)	0.256(0.088)	0.713(0.067)	0.101(0.043)	0.097(0.023)
	PriorGuide (exact ratio)	0.617(0.091)	0.121(0.055)	0.257(0.085)	0.635(0.046)	0.127(0.027)	0.127(0.027)
	PG-FullCov	0.618(0.091)	0.121(0.057)	0.254(0.084)	0.618(0.054)	0.112(0.033)	0.123(0.027)
	Spectra	**0.615**(0.092)	**0.116**(0.057)	0.251(0.089)	**0.587**(0.067)	**0.097**(0.044)	0.096(0.022)
OUP	Base	0.587(0.064)	0.186(0.075)	0.196(0.066)	0.742(0.088)	0.397(0.079)	0.221(0.079)
	SIR	0.524(0.009)	0.054(0.009)	0.184(0.075)	0.548(0.050)	**0.045**(0.011)	0.127(0.036)
	PriorGuide (exact ratio)	0.509(0.005)	0.058(0.011)	0.187(0.074)	0.521(0.023)	0.076(0.033)	0.131(0.037)
	PG-FullCov	0.509(0.004)	0.057(0.009)	0.185(0.074)	0.507(0.005)	0.048(0.012)	0.129(0.037)
	Spectra	**0.508**(0.003)	**0.052**(0.008)	0.184(0.075)	**0.506**(0.006)	0.047(0.011)	0.129(0.037)
BCI	Base	0.758(0.030)	0.254(0.049)	0.684(0.058)	0.948(0.016)	0.497(0.045)	0.714(0.026)
	SIR	0.591(0.026)	0.086(0.031)	0.450(0.061)	0.799(0.066)	0.224(0.312)	0.222(0.027)
	PriorGuide (exact ratio)	0.568(0.024)	0.085(0.028)	0.448(0.060)	0.572(0.015)	0.079(0.010)	0.224(0.027)
	PG-FullCov	**0.564**(0.022)	**0.083**(0.027)	0.451(0.061)	0.552(0.015)	**0.065**(0.006)	0.222(0.026)
	Spectra	0.567(0.024)	0.085(0.028)	0.449(0.059)	**0.551**(0.015)	0.066(0.007)	0.222(0.026)
GL-10D	Base	0.898(0.024)	0.308(0.028)	0.298(0.024)	0.997(0.002)	0.664(0.038)	0.328(0.033)
	SIR	0.760(0.056)	0.225(0.260)	0.212(0.024)	0.998(0.002)	0.965(0.064)	0.123(0.014)
	PriorGuide (exact ratio)	0.548(0.015)	0.055(0.004)	0.216(0.023)	0.558(0.029)	0.058(0.010)	0.095(0.011)
	PG-FullCov	**0.531**(0.015)	**0.043**(0.005)	0.212(0.024)	**0.522**(0.015)	**0.040**(0.005)	0.093(0.011)
	Spectra	0.536(0.017)	0.045(0.005)	0.212(0.024)	0.525(0.015)	**0.040**(0.004)	0.093(0.011)
GL-20D	Base	0.947(0.016)	0.300(0.017)	0.289(0.019)	1.000(0.000)	0.645(0.017)	0.303(0.018)
	SIR	0.927(0.029)	0.888(0.061)	0.208(0.017)	1.000(0.000)	0.852(0.139)†	0.149(0.013)
	PriorGuide (exact ratio)	0.584(0.016)	0.063(0.009)	0.213(0.016)	0.580(0.019)	0.057(0.005)	0.093(0.007)
	PG-FullCov	**0.546**(0.017)	**0.042**(0.003)	0.206(0.016)	**0.528**(0.017)	**0.037**(0.003)	0.091(0.007)
	Spectra	0.554(0.019)	0.044(0.003)	0.206(0.016)	0.531(0.018)	0.038(0.003)	0.090(0.007)

† Strong-shift SIR MMTV is defined for six of the ten GL-20D observations. Degenerate resampling prevents evaluation on the remaining four, so the reported mean and standard deviation use the six valid observations.

Table 5: **Paired C2ST differences at $(N, L) = (100, 0)$.** Entries are baseline minus Spectra, with 95% paired Student-t confidence intervals across ten observations. Positive values correspond to lower C2ST for Spectra.

Shift	Task	PriorGuide − Spectra	PG-FullCov − Spectra
Mild	Two Moons	$+0.0097$ $[+0.0018, +0.0177]$	$+0.0151$ $[+0.0039, +0.0264]$
	SLCP	$+0.0019$ $[-0.0070, +0.0108]$	$+0.0031$ $[-0.0026, +0.0087]$
	OUP	$+0.0014$ $[-0.0025, +0.0052]$	$+0.0011$ $[-0.0011, +0.0034]$
	BCI	$+0.0009$ $[-0.0022, +0.0040]$	-0.0026 $[-0.0047, -0.0004]$
	GL-10D	$+0.0113$ $[+0.0044, +0.0182]$	-0.0049 $[-0.0074, -0.0024]$
	GL-20D	$+0.0309$ $[+0.0130, +0.0488]$	-0.0073 $[-0.0109, -0.0038]$
Strong	Two Moons	$+0.0346$ $[+0.0155, +0.0537]$	$+0.0330$ $[+0.0097, +0.0562]$
	SLCP	$+0.0476$ $[+0.0260, +0.0692]$	$+0.0310$ $[+0.0128, +0.0491]$
	OUP	$+0.0156$ $[+0.0014, +0.0297]$	$+0.0007$ $[-0.0008, +0.0021]$
	BCI	$+0.0214$ $[+0.0128, +0.0300]$	$+0.0012$ $[-0.0023, +0.0047]$
	GL-10D	$+0.0339$ $[+0.0216, +0.0461]$	-0.0021 $[-0.0036, -0.0007]$
	GL-20D	$+0.0492$ $[+0.0382, +0.0602]$	-0.0030 $[-0.0051, -0.0009]$

K.3 PRACTICAL SAMPLING CONFIGURATIONS

OUP under the practical configuration. PriorGuide and Spectra have similar OUP accuracy at $(N, L) = (25, 8)$. Under the strong shift, moving from $(100, 0)$ to $(25, 8)$ improves PriorGuide's C2ST from 0.521 to 0.505, while Spectra changes from 0.506 to 0.505. The practical configuration therefore removes the small C2ST gap seen in the controlled comparison. Under the mild shift, the remaining differences are also small, with C2ST differing by 0.0013 and MMTV by 0.0007 in opposite directions.

Held-out evaluation with corrector updates. Following the prespecified selection rule, we froze $(25, 8)$ and the $(100, 0)$ control after development on three observations with one backbone. Table 6 evaluates the remaining seven observations at $(25, 8)$ using three independently trained backbones and three sampling seeds. On Two Moons and SLCP, Spectra has lower C2ST and MMTV than both PriorGuide and PG-FullCov, with the corresponding 95% paired Student-t intervals excluding zero. On GL-20D, the C2ST intervals against both baselines include zero. These seven observations are the held-out subset of the ten-observation practical benchmark, so this is a configuration-selection check rather than an independent replication.

Table 6: **Held-out single-factor comparison with Langevin corrections.** Strong shifts at $(N, L) = (25, 8)$. Mean (standard deviation) across seven held-out observations after averaging three sampling seeds within each of three backbones. C2ST is ideal at 0.5 and MMTV at zero. Bold indicates the displayed mean closest to the ideal value within each task, including rounded ties.

	Two Moons		SLCP		GL-20D	
Method	C2ST	MMTV	C2ST	MMTV	C2ST	MMTV
PriorGuide (exact ratio)	0.560(0.039)	0.174(0.099)	0.617(0.029)	0.115(0.030)	0.531(0.025)	**0.035**(0.004)
PG-FullCov	0.567(0.047)	0.192(0.118)	0.620(0.030)	0.117(0.032)	**0.530**(0.023)	**0.035**(0.004)
Spectra	**0.534**(0.017)	**0.103**(0.033)	**0.586**(0.038)	**0.100**(0.041)	0.531(0.024)	**0.035**(0.004)

K.4 FIXED-CENTER SHIFT-SEVERITY SWEEP

Because the mild and strong benchmarks differ in prior center and observations as well as width, they do not isolate the effect of shift severity. The sweep in Figure 2 therefore varies prior width while holding the other experimental factors fixed. For Two Moons and OUP, we fix the prior center, the ten observations from the mild single-factor setting, the three backbones, and the sampler, and set the target-prior marginal standard deviation to 0.50, 0.40, 0.30, 0.25, and 0.20 times that of the training prior. PriorGuide receives the exact analytic ratio, and both guided methods use $(N, L) = (25, 8)$ with three sampling seeds. All runs are evaluated against references restricted to the training support (Appendix J.4), whereas Table 1 uses the full benchmark reference. The two sets of results are therefore not compared numerically.

On Two Moons, the gap grows systematically as the target prior narrows in both metrics, and the paired difference is positive for at least nine of ten observations at every width. On OUP, all differences remain below 0.002 in magnitude and show no systematic trend.

L MIXTURE RESULTS

This appendix reports the systematic two-component benchmark, examines posterior component allocation and weight estimation, and evaluates composition beyond two components.

L.1 SYSTEMATIC MIXTURE BENCHMARK

The systematic benchmark evaluates ten mixture priors with three observations per prior on each of six tasks (Appendix J). Table 8 reports the controlled comparison at $(N, L) = (100, 0)$. At the practical sampling configurations of Table 1, Spectra-Direct and Spectra-PS use the same fixed weight estimates for both output configurations and share the same final component sampler. The two give nearly identical results on Two Moons, SLCP, OUP, and BCI. On the Gaussian Linear tasks, Spectra-PS gives lower mean C2ST and MMTV, with the larger difference on GL-10D.

Table 7: **Fixed-center shift-severity sweep.** Entries are mean paired differences (PriorGuide minus Spectra) across ten observations, with 95% paired Student-t confidence intervals after averaging over three sampling seeds and three backbones within each observation. Width is the target-prior marginal standard deviation relative to the training prior. Positive differences favor Spectra. Favor gives the number of observations with a positive paired difference.

		C2ST		MMTV	
Task	Width	Difference [95% CI]	Favor	Difference [95% CI]	Favor
Two Moons	0.50	+0.0075 [+0.0006, +0.0144]	10/10	+0.0297 [+0.0062, +0.0532]	10/10
	0.40	+0.0191 [+0.0032, +0.0350]	9/10	+0.0596 [+0.0203, +0.0990]	10/10
	0.30	+0.0434 [+0.0179, +0.0688]	10/10	+0.1140 [+0.0532, +0.1748]	10/10
	0.25	+0.0571 [+0.0280, +0.0861]	9/10	+0.1486 [+0.0788, +0.2185]	10/10
	0.20	+0.0727 [+0.0414, +0.1039]	10/10	+0.1828 [+0.1065, +0.2590]	10/10
OUP	0.50	−0.0013 [−0.0027, +0.0001]	3/10	+0.0008 [+0.0001, +0.0015]	8/10
	0.40	−0.0003 [−0.0015, +0.0009]	4/10	+0.0009 [−0.0003, +0.0021]	6/10
	0.30	+0.0003 [−0.0007, +0.0012]	5/10	+0.0011 [−0.0003, +0.0025]	7/10
	0.25	0.0000 [−0.0025, +0.0025]	5/10	+0.0013 [−0.0001, +0.0027]	8/10
	0.20	−0.0002 [−0.0024, +0.0019]	4/10	+0.0004 [−0.0022, +0.0029]	7/10

Table 8: **Mixture posterior inference across 180 prior–observation pairs.** Mean (standard deviation) across 30 pairs per task, after averaging sampling seeds within each backbone and then the three backbones within each pair. All output samplers use $(N, L) = (100, 0)$. PriorGuide receives the exact prior ratio. Direct, PS, and Ref are the Spectra variants of Appendix J.4. Ref uses reference component weights and is included only as a diagnostic. C2ST is ideal at 0.5 and MMTV at zero. Bold indicates the displayed mean closest to the ideal value among methods that do not use reference weights, including rounded ties.

					Spectra		
Task	Base	SIR	PriorGuide	PG-FullCov	Direct	PS	Ref
C2ST							
Two Moons	0.736(0.074)	0.543(0.018)	0.552(0.035)	0.538(0.021)	**0.514**(0.007)	0.515(0.007)	0.515(0.008)
SLCP	0.946(0.029)	0.796(0.086)	0.611(0.043)	0.587(0.042)	**0.554**(0.040)	0.555(0.040)	0.554(0.040)
OUP	0.703(0.063)	0.529(0.011)	0.517(0.009)	0.507(0.008)	**0.506**(0.006)	0.507(0.006)	0.506(0.006)
GL-10D	0.996(0.003)	0.999(0.001)	0.598(0.035)	0.539(0.025)	0.570(0.094)	**0.536**(0.023)	0.531(0.025)
GL-20D	0.999(0.001)	1.000(0.000)	0.614(0.031)	**0.526**(0.009)	0.532(0.039)	**0.526**(0.013)	0.525(0.012)
BCI	0.946(0.015)	0.789(0.061)	0.553(0.018)	**0.540**(0.017)	0.542(0.018)	0.542(0.018)	0.542(0.018)
MMTV							
Two Moons	0.510(0.147)	0.062(0.009)	0.159(0.081)	0.121(0.052)	**0.060**(0.008)	**0.060**(0.008)	0.060(0.008)
SLCP	0.498(0.075)	0.205(0.269)	0.139(0.059)	0.107(0.042)	**0.072**(0.025)	0.073(0.026)	0.071(0.025)
OUP	0.322(0.085)	0.046(0.012)	0.073(0.022)	0.048(0.023)	**0.045**(0.013)	**0.045**(0.013)	0.043(0.010)
GL-10D	0.636(0.044)	0.965(0.103)†	0.110(0.033)	0.062(0.030)	0.100(0.136)	**0.053**(0.017)	0.044(0.008)
GL-20D	0.640(0.030)	0.900(0.125)†	0.127(0.064)	0.043(0.013)	0.049(0.061)	**0.041**(0.016)	0.038(0.002)
BCI	0.485(0.039)	0.183(0.206)	0.074(0.015)	**0.061**(0.012)	0.062(0.012)	0.062(0.012)	0.062(0.012)

† SIR MMTV excludes degenerate runs. It is defined for 80/90 model-level runs on GL-10D and 48/90 on GL-20D, yielding aggregate values for all 30 GL-10D pairs and 25 of the 30 GL-20D pairs.

L.2 Component allocation and weight estimation

To isolate cases in which both posterior components remain relevant after observing x, we stratify the 150 non-BCI pairs by their reference component weights. Both components carry at least 5% posterior mass in 18 pairs: three Two Moons, four OUP, two SLCP, eight GL-10D, and one GL-20D pair. In the remaining 132 pairs, one component carries more than 95% of the posterior mass. Only four pairs have at least 20% mass in both components. These subsets are descriptive diagnostics within the existing benchmark, not separate evaluation sets.

On this 18-pair subset, Spectra-PS has lower mean C2ST and MMTV than Spectra-Direct under both output configurations (Table 9). Spectra-Direct has higher mean C2ST than PriorGuide under both output configurations (0.591 and 0.595 versus 0.568 and 0.549). At $(100, 0)$, this reversal is driven

by four Gaussian Linear pairs with large errors, and Spectra-Direct is worse in only 4 of the 18 pairs. Spectra-PS is more stable but still falls short of Spectra-Ref: its mean C2ST is 0.529 versus 0.519, and its MMTV is higher on all 18 pairs, averaging 0.072 versus 0.049. This remaining gap shows that practical mixture accuracy is still sensitive to component-weight estimation.

BCI component allocation. The remaining 30 BCI pairs are treated separately because their reference weights are finite-sample estimates from the SMC reference (Appendix J.4). None has at least 5% posterior mass in both components, and the largest minority weight is 7×10^{-4}. Spectra-Direct and Spectra-PS select the reference-dominant component in all 90 pair–backbone combinations, and their practical C2ST and MMTV agree to three decimal places. This extreme-allocation regime therefore says little about how accurately the very small minority weights are estimated.

Table 9: **Mixtures with substantial mass in both posterior components.** Mean (standard deviation) across the 18 pairs for which each component carries at least 5% reference posterior mass. Weight estimates are fixed across output configurations. Spectra-Ref uses reference component weights and is shown only as a diagnostic at $(N, L) = (100, 0)$.

	$(100, 0)$		$(25, 8)$	
Method	C2ST	MMTV	C2ST	MMTV
PriorGuide	0.568(0.043)	0.112(0.038)	0.549(0.038)	0.100(0.044)
Spectra-Direct	0.591(0.123)	0.158(0.171)	0.595(0.122)	0.161(0.170)
Spectra-PS	0.529(0.022)	0.072(0.023)	0.534(0.026)	0.074(0.024)
Spectra-Ref	0.519(0.019)	0.049(0.012)	–	–

Table 10: **Accuracy as the number of prior components increases on Two Moons.** Mean (standard deviation) across three observations at $(N, L) = (25, 8)$ after averaging three sampling seeds within each backbone. Spectra-PS uses one backbone, while Spectra-Ref and PriorGuide use three backbones. All methods are evaluated against the support-matched reference.

	C2ST			MMTV		
K	Spectra-PS	Spectra-Ref	PriorGuide	Spectra-PS	Spectra-Ref	PriorGuide
2	0.517(0.008)	0.518(0.009)	0.573(0.062)	0.064(0.016)	0.070(0.011)	0.211(0.152)
4	0.521(0.013)	0.517(0.008)	0.586(0.049)	0.074(0.024)	0.075(0.011)	0.260(0.134)
8	0.520(0.010)	0.519(0.009)	0.555(0.049)	0.062(0.004)	0.069(0.004)	0.178(0.141)
16	0.507(0.001)	0.509(0.003)	0.545(0.038)	0.063(0.011)	0.070(0.002)	0.170(0.115)

L.3 Composition beyond two components

We next evaluate mixtures with more than two components. On Two Moons, the target priors are equal-weight mixtures of $K \in \{2, 4, 8, 16\}$ Gaussian components with the strong-shift width. Their centers are evenly spaced on a circle around the mild-prior center, with radius one quarter of the training-box side, and each smaller component set is contained in the larger ones. We evaluate accuracy up to $K = 16$, beyond which adjacent components are separated by less than one component standard deviation and overlap heavily.

Sampling uses $(N, L) = (25, 8)$ with three sampling seeds on three observations. The support-matched quadrature reference also provides the reference component weights. Spectra-PS uses the weight-estimation budget of the two-component benchmark (Appendix J.3) and one backbone. Spectra-Ref and PriorGuide average over three backbones.

Both Spectra variants remain accurate as K increases. Conditioning reduces the effective number of components relative to the nominal K, but multiple components retain substantial posterior mass for the larger mixtures (Table 11). At $K = 16$, the effective number of components ranges from 2.5 to 4.5, and five components carry more than 5% posterior mass for every observation. Path-space estimates closely track the reference weights, with a maximum absolute error of 1.6×10^{-2} across all K, components, and observations. With only three observations, we treat this as a composition study rather than a method comparison.

Table 11: **Component allocation and path-space weight accuracy.** For each of the three observations, the table reports the effective number of components $1/\sum_k \alpha_k^2$ and the number carrying at least 5% of the posterior mass. The final column gives the largest absolute difference between path-space and reference component weights across all components and observations.

	Reference weights		Path-space estimate	
K	Effective components	Components with at least 5%	Effective components	Largest weight error
2	1.00 / 1.02 / 1.00	1 / 1 / 1	1.00 / 1.02 / 1.00	3.4×10^{-4}
4	1.19 / 1.36 / 1.12	2 / 2 / 2	1.19 / 1.35 / 1.12	2.2×10^{-3}
8	1.21 / 3.09 / 2.18	2 / 3 / 2	1.21 / 3.07 / 2.18	1.6×10^{-2}
16	2.50 / 4.54 / 4.10	5 / 5 / 5	2.50 / 4.52 / 4.15	8.4×10^{-3}

M COMPUTATIONAL COST

This appendix reports single-factor online sampling time, the first-use cost of mixture adaptation, and scaling with the number of prior components.

The timing experiments use the same Spectra implementation as the accuracy experiments. For PriorGuide, we report both the benchmarked implementation and the vector–Jacobian-product (VJP) implementation described in Appendix J.5. The latter evaluates the same guidance update without explicitly forming the full denoiser Jacobian and serves as the strengthened timing baseline. Across the 22 timed configurations, PriorGuide (VJP) is 2.2–3.2× slower than Spectra, compared with 3.5–19.7× for the benchmarked implementation. These timings are specific to the implementations and hardware used here.

Timing protocol. Unless stated otherwise, online sampling times are medians of five synchronized repeats for 1,000 posterior samples on a single A100. Compilation and input preparation are excluded from all timed repeats. Method-specific setup costs are reported separately when first-use cost is considered in Appendix M.2. Each timing uses one observation and one backbone per task.

M.1 SINGLE-FACTOR ACCURACY AND ONLINE COST

Figure 5 reports online sampling time on Two Moons, SLCP, BCI, GL-10D, and GL-20D at the sampler configurations used in the main benchmark. Both the benchmarked and VJP implementations of PriorGuide are shown.

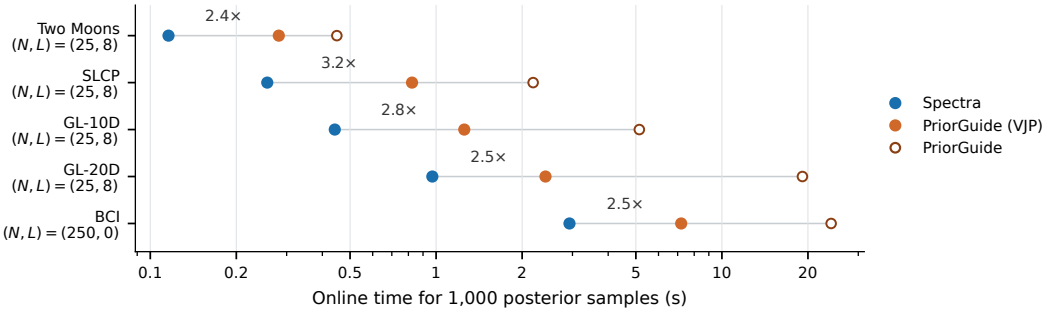


Figure 5: **Online sampling time across the five timed tasks.** Each task uses the sampler configuration of the main benchmark and the timing protocol above. Labels show the runtime ratio of PriorGuide (VJP) to Spectra.

Figure 6 shows the MMTV–time trade-off and adds PG-FullCov to the methods in Figure 4. The strong-shift development study uses three observations, one backbone, and three sampling seeds, and evaluates $(N, L) \in \{(25, 0), (25, 8), (50, 2), (100, 0), (100, 2), (250, 0)\}$. Accuracy is averaged over seeds and then over the three observations, with observation-level standard errors. These are the development observations used for configuration selection. Table 6 reports the held-out evaluation. Online time is measured on one of the three observations using the same backbone.

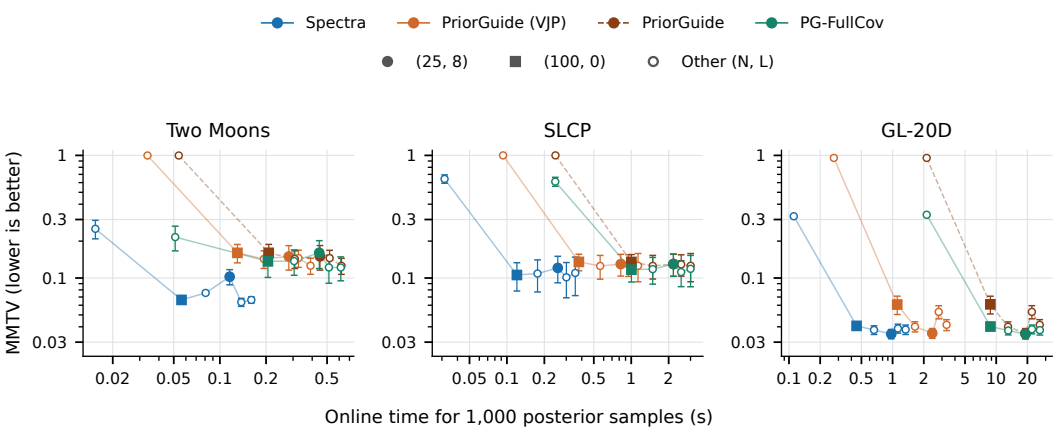


Figure 6: **Single-factor MMTV versus online sampling time.** The figure adds PG-FullCov to the methods shown in Figure 4. Filled circles mark $(25, 8)$, squares mark $(100, 0)$, and hollow circles mark the remaining sampler configurations. Error bars are observation-level standard errors. Both axes are logarithmic.

M.2 Multi-component first-use cost

Figure 7 compares the cost of the first 1,000 mixture posterior samples for PriorGuide, PG-FullCov, Spectra-Direct, and Spectra-PS. Each task uses two prespecified prior–observation pairs, one backbone, and $(N, L) = (100, 0)$. Stage costs are summarized by the median across the two pairs. Each stage uses five timed repeats, except path-space weight estimation, which uses three. Compilation and input preparation are excluded.

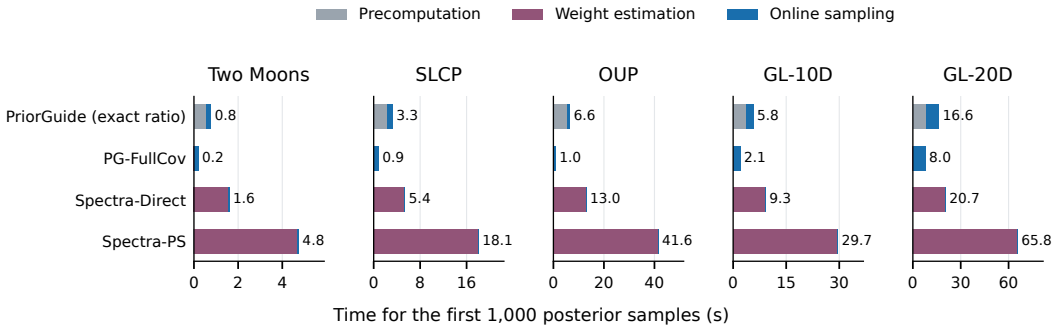


Figure 7: **Cost of the first 1,000 mixture posterior samples.** Stacks separate precomputation, component-weight estimation, and online sampling for PriorGuide, PG-FullCov, Spectra-Direct, and Spectra-PS. Labels give the total displayed cost. Panels use different linear time scales.

Spectra-Direct and Spectra-PS share the same inexpensive component sampler, but weight estimation can dominate the first adaptation. On SLCP, for example, Spectra-PS takes 0.11 seconds to draw 1,000 samples once the weights are available, but 18.09 seconds including weight estimation, compared with 3.28 seconds for PriorGuide including precomputation. Replacing PriorGuide's online stage with the VJP implementation lowers its first-use cost further, but Spectra-PS remains the most expensive first-use method on all five tasks.

Once estimated, the normalizers can be reused for the same observation and component set when only the prior mixture coefficients change. These measurements cover a single adaptation and therefore do not quantify the amortized cost over a sequence of prior changes.

M.3 Scaling with the number of components

Table 12 measures online sampling time on Two Moons as the number of prior components increases from $K = 1$ to $K = 64$. The target priors are equal-weight Gaussian mixtures with the strong-shift

component width. Their centers lie on a common circle, and each smaller component set is contained in the larger ones. For $K = 1$, the single component is placed at the center of the circle. Timing uses $(N, L) = (25, 8)$ and the protocol above. Component selection and PriorGuide precomputation are outside the timed region.

Table 12: **Online sampling time as the number of prior components increases on Two Moons.** Median seconds for 1,000 posterior samples at $(N, L) = (25, 8)$ under the timing protocol of Appendix M.

K	1	2	4	8	16	32	64
Spectra	0.1166	0.1165	0.1165	0.1165	0.1165	0.1165	0.1164
PriorGuide	0.4464	0.4440	0.4442	0.4446	0.4456	0.4466	0.4482

Spectra's online sampling time varies by less than 0.2% from $K = 1$ to $K = 64$, because each trajectory follows only the transported score of its selected component. PriorGuide combines all K components at every guidance step, but its time changes by less than 1% on this two-dimensional task, where network evaluation and the score Jacobian dominate the per-step cost. The PriorGuide-to-Spectra time ratio therefore remains between 3.81 and 3.85.

Table 12 uses the benchmarked PriorGuide implementation, rather than the VJP implementation used for the single-factor speed comparisons in Section 4.4. The separate path-space weight-estimation stage runs proposal trajectories for every component, so its cost grows linearly with K.

N Adaptation beyond the exact single-factor class

The main benchmarks use prior changes whose factors are exactly transportable. This appendix applies the positive-mixture approximation of Appendix I to two Gaussian targets outside that class, an anisotropy sweep and a rotated correlated prior, both on Two Moons.

N.1 Controlled anisotropy boundary

We start from the narrowest isotropic target in Figure 2, whose marginal standard deviation $s = 0.2/\sqrt{3}$ is one fifth of the training-prior value, and make it anisotropic while keeping its center and determinant fixed,

$$\Sigma_{\gamma,+} = \operatorname{diag}\left(\frac{s^2}{\gamma},\, s^2\gamma\right), \qquad \Sigma_{\gamma,-} = \operatorname{diag}\left(s^2\gamma,\, \frac{s^2}{\gamma}\right), \qquad \gamma \in \{1, 2, 4\}.$$

The condition number is γ^2. At $\gamma = 1$ the target is a single exact factor. For $\gamma > 1$ we use the construction of Appendix I with $\lambda = 0.9\,\lambda_{\min}(\Sigma)$ and $h = 6$, which gives $K = 52$ factors at $\gamma = 2$ and $K = 104$ at $\gamma = 4$. The ten observations, three backbones, and samplers are those of Figure 2, with $(N, L) = (25, 8)$ for PriorGuide and Spectra.

The approximation error grows with γ. On the training support, the prior TV distance to the exact target is 0.00149 at $\gamma = 2$ and 0.03312 at $\gamma = 4$. The mean posterior TV distance is 0.00306 and 0.00356 for the two $\gamma = 2$ orientations and 0.02360 and 0.02889 for $\gamma = 4$, and the reference weights spread over 3.46 to 5.49 effective components. This error is small relative to the differences between methods. Scoring the Spectra-Ref samples against the approximating target instead of the exact one changes mean C2ST by at most 0.0015 and mean MMTV by at most 0.0070.

Component-weight estimation is the less reliable step. On nine of the ten observations, Spectra-PS and Spectra-Ref differ in C2ST by at most 0.003 in every setting. On the remaining observation, path-space estimation places nearly all posterior mass on components with negligible reference weight in the $+$ orientation, and C2ST worsens by 0.417 at $\gamma = 2$ and 0.405 at $\gamma = 4$. In the $-$ orientation, the same observation degrades less, by 0.059 and 0.018. The error in the estimated normalizers may come from the learned finite base chain, from Monte Carlo error in the estimator, or from both, and our diagnostics do not separate them.

PriorGuide, although given the exact anisotropic ratio, is worse than Spectra-Ref in both metrics on every observation in all four anisotropic settings, and worse than Spectra-PS on at least nine of the ten. SIR is worse than Spectra-Ref in C2ST on every observation but has the lower MMTV on five to

eight of the ten, as on Two Moons in Table 1. Because the target posterior changes with γ, values should not be compared across rows.

Table 13: **Anisotropy sweep on Two Moons.** Mean (standard deviation) over ten observations, scored against support-matched references for the exact target. Each observation's value averages three backbones, and for PriorGuide and Spectra three sampling seeds per backbone. Sampling uses $(N, L) = (25, 8)$, except $(100, 0)$ for SIR. At $\gamma = 1$, $K = 1$ and Spectra-Ref and Spectra-PS coincide.

γ	Orient.	K	SIR	PriorGuide	Spectra-Ref	Spectra-PS
C2ST						
1	iso	1	0.613(0.069)	0.621(0.055)	0.548(0.016)	0.548(0.016)
2	+	52	0.589(0.090)	0.621(0.092)	0.528(0.013)	0.570(0.143)†
2	−	52	0.578(0.044)	0.604(0.063)	0.528(0.012)	0.535(0.027)
4	+	104	0.599(0.070)	0.609(0.084)	0.531(0.023)	0.571(0.149)†
4	−	104	0.590(0.073)	0.600(0.084)	0.527(0.018)	0.530(0.021)
MMTV						
1	iso	1	0.121(0.121)	0.310(0.118)	0.127(0.021)	0.127(0.021)
2	+	52	0.123(0.171)	0.311(0.191)	0.083(0.016)	0.166(0.272)†
2	−	52	0.116(0.095)	0.273(0.140)	0.085(0.014)	0.099(0.050)
4	+	104	0.088(0.076)	0.272(0.180)	0.072(0.019)	0.159(0.287)†
4	−	104	0.096(0.065)	0.251(0.171)	0.073(0.011)	0.079(0.017)

† Driven by one observation, on which Spectra-PS is worse than Spectra-Ref in C2ST by 0.417 ($\gamma = 2$) and 0.405 ($\gamma = 4$). On the other nine, the two differ by at most 0.003.

N.2 Rotated correlated Gaussian example

We apply the quadrature construction of Appendix I to a correlated Two Moons target prior $\mathcal{N}(\mu, \Sigma)$ with $\mu = (0.3, -0.3)$ and Σ obtained by rotating $\mathrm{diag}(0.04,\, 0.04/6)$ by $45°$. We use $\lambda = 0.9\,\lambda_{\min}(\Sigma) = 0.006$ and $h = 6$, which give Gauss–Hermite orders 16 and 4 and hence $K = 64$ factors. Posterior sampling uses $(N, L) = (100, 0)$ with three seeds of 1,000 samples.

Table 14: **Correlated-prior approximation on Two Moons.** Approx. TV is the posterior total variation distance induced by the $K = 64$ finite quadrature, evaluated on a 4096×4096 grid. C2ST is mean $\pm$ standard deviation over three sampling seeds. Approx. TV quantifies posterior error from the finite prior approximation, while C2ST evaluates the sampled posterior against the exact target. PriorGuide and PG-FullCov use the exact ratio; Spectra-Ref and Spectra-PS use the quadrature approximation with reference and path-space component weights, respectively.

Observation	Approx. TV	PriorGuide	PG-FullCov	Spectra-Ref	Spectra-PS
1	0.00306	0.535 ± 0.003	0.527 ± 0.007	0.507 ± 0.006	0.513 ± 0.006
2	0.00456	0.519 ± 0.011	0.519 ± 0.013	0.518 ± 0.008	0.518 ± 0.006
3	0.00285	0.549 ± 0.008	0.526 ± 0.009	0.530 ± 0.009	0.529 ± 0.010

The same quadrature is used for all three observations. Spectra-Ref computes component weights from the quadrature posterior on the grid, while Spectra-PS estimates all 64 weights using the path-space estimator and the budget of Appendix J.3. Across three observations and three backbones, the estimated weights are within 0.026–0.092 in L_1 distance of the reference weights. Replacing reference weights by path-space estimates changes C2ST by $+0.0013 \pm 0.0020$ and MMTV by $+0.0035 \pm 0.0018$ after averaging over backbones. This is a low-dimensional demonstration rather than a scalable construction for correlated priors. Accurate path-space weights here do not establish reliable weight estimation at large K, and in Appendix N.1 one observation fails at $K = 52$ and $K = 104$.