SCALABLE DIFFUSION SBI FOR COMPOSITIONAL INFERENCE UNDER SIMULATOR MISSPECIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Simulation-based inference (SBI) is challenging when many heterogeneous observations must be composed, hierarchical latent structure must be preserved, and the simulator is misspecified relative to observed data. We develop sampling and fine-tuning methods for diffusion-based SBI in design-conditional settings, where the same simulator is queried across different experimental conditions ξ. We extend compositional score-based inference with a continuous-time diffusion coefficient that accounts for the number of observations, avoiding Jacobian and auxiliary-covariance corrections. We introduce Hierarchical Blockwise Diffusion Sampling (HBDS), which infers shared parameters and group-specific latent states using a single pretrained (PT) model, with the hierarchy specified only at sampling time. Together, these methods support variable observation sets and groupings without retraining. To address misspecification, we introduce path-regularized fine-tuning that adapts the learned likelihood to observations and transfers corrections to posterior inference. Using Girsanov's theorem, we quantify path divergence between PT and fine-tuned (FT) models across experimental designs and interpret it alongside predictive errors to distinguish candidate misspecification correction from unnecessary adaptation. We evaluate compositional sampling on exact-score Gaussian and SLCP benchmarks, HBDS with analytic and learned scores on a controlled hierarchical model, and fine-tuning and localization on a separate analytic model with known design-dependent discrepancy. Finally, we apply the framework to 940 measurements across four cell lines in a mechanistic Bone Morphogenetic Protein (BMP) signaling model, where fine-tuning improves both posterior-predictive accuracy and posterior parameter inference relative to the PT model.

1 INTRODUCTION

Mechanistic simulators compress scientific knowledge into predictive models whose latent parameters carry physical or biological meaning. Simulation-based inference (SBI) (Cranmer et al., 2020) can amortize Bayesian inference in these models even when their likelihoods cannot be evaluated. In reality, however, inference rarely consists of conditioning on a single observation from a perfectly specified simulator. Data may contain many heterogeneous observations, shared and group-specific latent variables, and systematic discrepancies between the simulator and the observed system.

These challenges have motivated several complementary directions in SBI. Compositional methods aggregate multiple observations in a score-based model without retraining a posterior estimator, beginning with Factorized Neural Posterior Score Estimation (F-NPSE) (Geffner et al., 2023) and followed by covariance-aware diffusion samplers for tall-data SBI (Linhart et al., 2026). Meanwhile, recent hierarchical approaches explicitly model shared and local latent structure through hierarchical estimators or training constructions (Arruda et al., 2026; Charles et al., 2026; Heinrich et al., 2024). Separately, simulator misspecification has motivated a growing literature on detection and mitigation (Boelts, 2026; Cannon et al., 2022), including explicit discrepancy models (Ward et al., 2022), misspecification-aware representations and summary statistics (Huang et al., 2023b; Schmitt et al., 2024), and distributional alignment or calibration methods (Wehenkel et al., 2025). These lines of work address composition, hierarchy, and misspecification, but typically as distinct problems.

We study how these requirements can be combined within a single diffusion-based SBI framework (Sharrock et al., 2024). Our goal is to reuse an amortized model across arbitrary collections of

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.

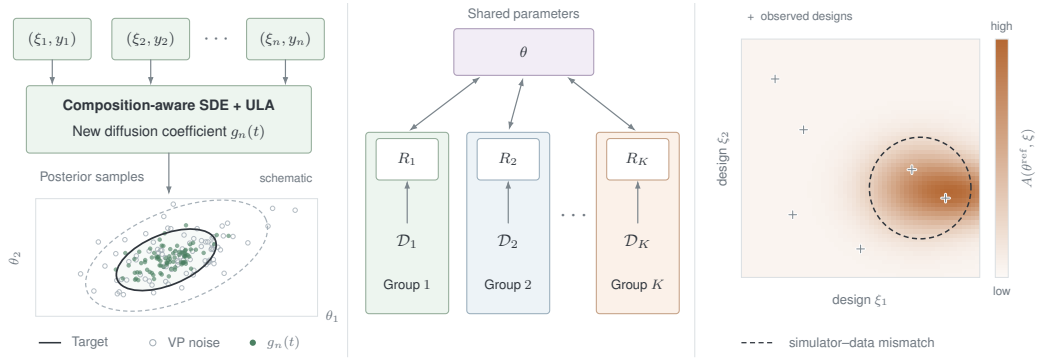

Figure 1: Schematic overview of the framework. (*Left*) Compositional sampling combines observations using the derived diffusion coefficient $g_n(t)$ and optional Langevin correctors, without retraining. (*Middle*) HBDS alternates updates of shared parameters θ and group-specific latent states R_k conditioned on each group's data $\mathcal{D}_k$. (*Right*) Path divergence $\mathcal{A}(\theta^{\text{ref}}, \xi)$ measures the magnitude of likelihood adaptation across experimental designs. Comparing this field with predictive improvement helps distinguish candidate misspecification correction from over-adaptation.

experimental observations, introduce hierarchical structure at sampling time rather than fixing it in a hierarchy-specific estimator, and adapt a simulator-trained surrogate to observed data while retaining posterior inference in the original mechanistic parameterization.

Achieving the last of these requires likelihood and posterior score representations to remain coupled as the simulator-trained surrogate is adapted to observed data. We use the Simformer of Gloeckler et al. (2024) as the diffusion backbone and introduce a new mechanism for transfer learning (Pan & Yang, 2010) through shared embeddings, allowing likelihood fine-tuning to update representations used for posterior inference (§ G.6). The framework is summarized in Figure 1. Building on this coupled surrogate, our contributions are:

- *Composition-aware diffusion sampling.* We derive an n-dependent continuous-time diffusion coefficient for compositional SBI, extending F-NPSE to efficient SDE sampling while avoiding the Jacobian and auxiliary-covariance costs of JAC and GAUSS.
- *Hierarchical inference at sampling time.* We introduce Hierarchical Blockwise Diffusion Sampling (HBDS), reusing a single joint diffusion model to infer shared and group-specific latents with hierarchy introduced only at sampling time.
- *Path-space adaptation and localization.* We fine-tune the surrogate likelihood to observations while transferring the correction to posterior inference, and use Girsanov's theorem to isolate *where* model misspecification may occur.
- *Large-scale systems biology inference.* We apply the full framework to a 65-parameter systems biology simulator of the Bone Morphogenetic Protein (BMP) signaling pathway with 940 observations, showing how composition, hierarchy, and misspecification-aware adaptation jointly make SBI practical on a nontrivial scientific model.

2 Background

2.1 Design-dependent simulation-based inference

SBI considers a simulator from which observations can be sampled conditional on latent parameters and, when relevant, experimental conditions. In that setting we write $y \sim p_{\text{eval}}(y \mid \theta)$, where $p_{\text{eval}}(y, \theta) = p_{\text{eval}}(y \mid \theta)\, p(\theta)$ is an *expert simulator*, distinct from the unknown true data-generating process $p^*(y, \theta)$. We adopt the $p_{\text{eval}}(y \mid \theta)$ notation of Smith et al. (2025) and use the *design-dependent* setting where an experimental design ξ alters the joint model and resulting posterior, yielding $p_{\text{eval}}(y, \theta \mid \xi)$ and related $p^*(y, \theta \mid \xi)$. We represent a simulator as a function

$$f : \Xi \times \Theta \longrightarrow \mathcal{Y}, \quad \Xi \subset \mathbb{R}^D,\ \Theta \subset \mathbb{R}^M,\ \mathcal{Y} \subset \mathbb{R}^B,$$

where Ξ, Θ, and $\mathcal{Y}$ denote the spaces of experimental designs, latent model parameters, and observations, respectively. The simulator generates samples from the prior predictive distribution, $\theta \sim p(\theta)$

and $y \sim p_{\text{eval}}(y \mid \theta, \xi)$, which provide training data for amortized density, ratio, or score estimators targeting quantities such as the posterior $p(\theta \mid y, \xi)$ or surrogate likelihood $p(y \mid \theta, \xi)$.

2.2 Score-based diffusion models

Score-based diffusion models define a forward noising process that gradually transforms a data distribution $p_0(y_0)$ into a tractable reference distribution $p_T(y_T)$, typically a Gaussian distribution, and learn the time-dependent score $s_\psi(y_t, t) \approx \nabla_{y_t} \log p_t(y_t)$ by denoising score matching (Hyvärinen, 2005; Vincent, 2011). In continuous time, the forward process and its score-based reverse-time counterpart take the form

$$dy_t = \begin{cases} f(y_t, t)dt + g(t)dw, & \text{(forward)} \\ [f(y_t, t) - g(t)^2 s_\psi(y_t, t)]dt + g(t)d\overleftarrow{w}, & \text{(reverse)}, \end{cases}$$

where w and $\overleftarrow{w}$ are forward- and reverse-time Wiener processes, and f and g are the drift and diffusion coefficients (Anderson, 1982; Song et al., 2021). Integrating the reverse process from samples $y_T \sim p_T$ approximately recovers samples from p_0.

3 Methods

3.1 Joint diffusion surrogate pretraining

Diffusion models have been proposed in SBI (Sharrock et al., 2024) but are typically constrained by the type of inference object they infer, be it a posterior or likelihood. This limits the types of training and downstream analyses that an SBI inference object can perform. To alleviate this constraint, Gloeckler et al. (2024) proposed a mask-conditioned joint diffusion model to support different conditioning patterns over its variables, enabling sampling from either the likelihood, posterior, or any combination of random variables — provided it was trained on masks that encode the desired conditioning. It does this by creating tokens from the random variables modeled, e.g., y, ξ, or θ, where tokenization also includes whether the variable and its value are to be inferred (latent) or not (conditioned). A transformer architecture allows graphical constraints of the expert model via an attention mask that enforces domain-relevant relations when training the joint model (Weilbach et al., 2023). Each variable has a condition state, giving $M_C \in \{0, 1\}$, where the null vector $M_C = (0, \ldots, 0)$ is an unconditional model. We can manipulate M_C to model a likelihood, posterior, or any combination of variables in

$$\ell(\phi, M_C, t, \mathbf{y}_0, \mathbf{y}_t) = \zeta(t)(1 - M_C)\Big(s_\phi^{M_E}\big(\mathbf{y}_t^{M_C}, t\big) - \nabla_{\mathbf{y}_t} \log p_t\big(\mathbf{y}_t|\mathbf{y}_0\big)\Big),$$

where $\zeta(t)$ is a noise schedule, and the score depends on masks M_E and M_C. We then minimize this loss function over samples

$$\mathcal{L}(\phi) = \mathbb{E}_{M_C, t, \mathbf{y}_0, \mathbf{y}_t}\Big[\big\|\ell\big(\phi, M_C, t, \mathbf{y}_0, \mathbf{y}_t\big)\big\|_2^2\Big], \tag{1}$$

by conventional score matching (Hyvärinen, 2005; Vincent, 2011), and resulting in the pretrained (PT) model $p_{\phi_0}(y, \theta \mid \xi)$, where ϕ_0 denotes the PT network parameters. We train using a variance-preserving SDE (VP-SDE) schedule (Song et al., 2021). The choice of condition masks and their frequency during training influences the subsequent model's conditional sampling performance, which we discuss in § F.

3.2 Fine-tuning to address model misspecification

The PT joint surrogate provides a conditional likelihood $p_{\phi_0}(y \mid \theta, \xi) \approx p_{\text{eval}}(y \mid \theta, \xi)$, which inherits simulator misspecification relative to $p^*(y \mid \theta, \xi)$. Following Cannon et al. (2022), we conceptualize misspecification via an unknown transformation that maps simulator outputs to observations. In our setting this is naturally a map between *different* sample spaces, since simulator outputs must be post-processed (e.g. normalized) to match the observation representation:

$$y \sim p_{\text{eval}}(\cdot \mid \theta, \xi), \qquad y_o = T(y), \qquad T : \mathcal{Y}_{\text{sim}} \to \mathcal{Y}_{\text{obs}}.$$

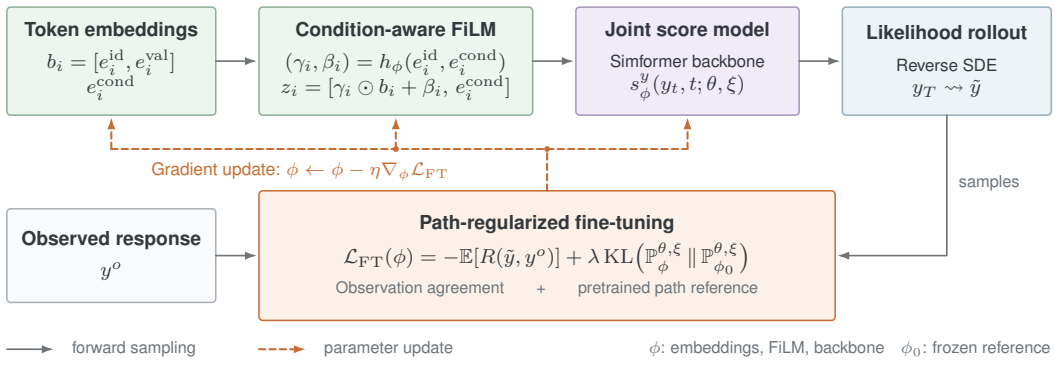

Figure 2: Likelihood fine-tuning of the joint diffusion surrogate. At fixed reference parameters and design, generated responses are evaluated against observations with a path-space KL penalty relative to the PT model. Gradients through sampling update the token embeddings, FiLM layers, and score network, which also support posterior queries.

Under this representation, $p^*(\cdot \mid \theta, \xi)$ is the pushforward $p^*(\cdot \mid \theta, \xi) = T_{\#} p_{\text{eval}}(\cdot \mid \theta, \xi)$. Rather than specifying T explicitly, we seek to learn an *implicit* misspecification correction that absorbs this transformation into the joint model's parameters. We draw on diffusion fine-tuning to address simulator misspecification, adapting the surrogate likelihood to observed data. Specifically, we use the *implicit diffusion* method of Marion et al. (2025), which allows us to optimize agreement with observations by differentiating through likelihood sampling. We discuss alternative diffusion fine-tuning approaches and their suitability for SBI in § G.5.

In SBI, observations y_o, or design–observation pairs (y_o, ξ), provide a target for fine-tuning the surrogate likelihood. The learned correction must also inform the posterior over θ conditioned on those observations. We enable this transfer through token-aware Feature-wise Linear Modulation (FiLM) (Perez et al., 2018), which modulates token embeddings according to their identity and whether they are latent or conditioned. These embeddings are shared across likelihood and posterior queries in the masked transformer, allowing likelihood fine-tuning to update representations used for posterior inference; Figure 2 illustrates this mechanism, with embedding details in § G.6.

How can we fine-tune the likelihood $p_\phi(y \mid \theta, \xi)$ to observed data when the corresponding mechanistic parameters θ are unknown? For observed design–response pairs (y_i^o, ξ_i), we condition the surrogate likelihood on a fixed, simulator-aligned reference parameter θ^{ref}. In supervised settings, θ^{ref} may be provided by a labeled real-world dataset $\mathcal{D}_{\text{cal}} = \{(\theta^i, y_o^i)\}_{i=1}^{N_{\text{cal}}}$, where each observation y_o^i is paired with its ground-truth parameter θ^i, as in Wehenkel et al. (2025). When it is unknown, it may instead be an externally obtained best-fit estimate. In the BMP application, we use the domain-standard least-squares estimate $\theta^{\text{LSR}} \in \arg\min_\theta \sum_i \ell(f(\theta, \xi_i), y_i^o)$, which provides a parameter configuration explicitly grounded in the expert simulator. We do not interpret θ^{LSR} as a true parameter vector; it serves only as a fixed reference for adapting the likelihood to observed data.

For fixed $(\theta^{\text{ref}}, \xi)$, we draw samples $y \sim p_\phi(y \mid \theta^{\text{ref}}, \xi)$ and evaluate their agreement with observations through a reward $R(y; y^o)$. The diffusion sampler induces an output distribution $\pi^\star(\phi; \theta^{\text{ref}}, \xi)$ defined implicitly by the network parameters ϕ. We formulate fine-tuning as optimization of a functional $\mathcal{F}$ of this sampled distribution, differentiating through the sampler with the adjoint method of Implicit Diffusion (Marion et al., 2025). Suppressing the fixed conditioning arguments, the endpoint-regularized problem is $\min_\phi \mathcal{F}(\pi^\star(\phi))$, with

$$\mathcal{F}(p) := -\mathbb{E}_{y \sim p}[R(y; y^o)] + \lambda \,\text{KL}(p \,\|\, \pi^\star(\phi_0)) \,, \tag{2}$$

where p is an output distribution and $\lambda \geq 0$ penalizes departure from the PT output law. In our experiments, R is the negative mean squared error (MSE) between generated and observed responses. In implementation, we replace the endpoint KL by a path-space KL penalty, giving the stronger regularization described in §§ 3.3 and G.2. Alternative fine-tuning frameworks are discussed in § G.5. Surprisingly, anchoring likelihood fine-tuning at a fixed θ^{ref} does not reduce posterior inference to a point estimate — in the BMP example, the fine-tuned (FT) posterior marginals generally shift toward the reference relative to pretraining while retaining substantial spread (Figure 15).

3.3 LOCALIZING SIMULATOR MISSPECIFICATION WITH PATH-SPACE DIVERGENCE

For fixed (θ, ξ), the reverse-SDE sampler underlying $\pi^\star(\phi; \theta, \xi)$ induces a path measure $\mathbb{P}_\phi^{\theta,\xi}$ over trajectories $Y = (Y_t)_{t\in[0,T]}$, whose generated endpoint distribution is $\pi^\star(\phi; \theta, \xi)$. Fine-tuning from ϕ_0 to ϕ therefore changes not only the endpoint distribution of generated responses but the probability law of the complete sampling trajectory, giving PT and FT path measures $\mathbb{P}_{\phi_0}^{\theta,\xi}$ and $\mathbb{P}_\phi^{\theta,\xi}$. We quantify this change using path-space KL, which we use in place of the endpoint penalty in Eq. (2). Since the generated response is a projection of the complete diffusion trajectory, the data-processing inequality implies that path-space KL upper-bounds the corresponding endpoint divergence and can therefore capture adaptation not visible from the final samples alone. Under the regularity and integrability conditions stated in § G.1, Girsanov's theorem gives

$$\mathrm{KL}\Big(\mathbb{P}_\phi^{\theta,\xi} \,\Big\|\, \mathbb{P}_{\phi_0}^{\theta,\xi}\Big) = \frac{1}{2}\,\mathbb{E}_{\mathbb{P}_\phi^{\theta,\xi}}\left[\int_0^T g(t)^2 \left\| s_\phi^y(Y_t, t; \theta, \xi) - s_{\phi_0}^y(Y_t, t; \theta, \xi)\right\|_2^2 dt\right]. \tag{3}$$

We denote this discrepancy by $\mathcal{A}(\theta, \xi)$. Evaluating $\mathcal{A}$ at the same fixed reference parameter θ^{ref} used during fine-tuning yields a non-negative scalar field over design space, assigning each design a measure of the trajectory-level displacement between the PT and FT reverse-SDEs.

Stochastic optimal control (SOC) interpretation. Equation (3) admits a SOC interpretation. In SOC, a control field $u(x, t)$ perturbs a reference drift through a controlled SDE $dX_t^u = [b_0(X_t^u, t) + \sigma(t)u(X_t^u, t)]dt + \sigma(t)dW_t$, whose solution induces a controlled path measure $\mathbb{P}^u$. Taking the PT reverse-SDE as the reference process and $\sigma(t) = g(t)I$, the FT score corresponds to the control $u_\phi = -g(t)(s_\phi^y - s_{\phi_0}^y)$. Hence, $\mathcal{A}(\theta, \xi)$ is the quadratic control energy associated with transporting the PT path law to the FT one, connecting likelihood adaptation to stochastic optimal control and KL-constrained path-space transport (Blessing et al., 2025).

Fisher geometry over conditioning variables. Beyond comparing PT and FT path laws at fixed (θ, ξ), the same path-space formulation induces a local geometry over the conditioning variables themselves. Let $z = (\theta, \xi)$ denote the continuous mechanistic and design coordinates, and fix the learned model ϕ. Assuming the induced path laws vary smoothly with z, nearby path measures satisfy the absolute-continuity conditions required by Girsanov's theorem, and distinct nearby conditioning values induce distinct path laws, the family

$$\mathcal{M}_\phi = \big\{\mathbb{P}_\phi^z : z \in \Theta \times \Xi\big\} \subset \mathcal{P}\big(C([0, T]; \mathbb{R}^{d_y})\big)$$

can be treated locally as a statistical manifold (Figure 3). Here, $C([0, T]; \mathbb{R}^{d_y})$ is the space of continuous trajectories $Y : [0, T] \to \mathbb{R}^{d_y}$, and $\mathcal{P}(\cdot)$ denotes probability measures on that space.

With a shared diffusion schedule and terminal law, the same Girsanov argument gives

$$\mathcal{I}_\phi^{\mathrm{path}}(z) := \mathbb{E}_{Y\sim\mathbb{P}_\phi^z}\left[\int_0^T g(t)^2 J_z s_\phi^y(Y_t, t; z)^\top J_z s_\phi^y(Y_t, t; z)\, dt\right], \tag{4}$$

where $J_z s_\phi^y = \partial s_\phi^y / \partial z$. For a small perturbation δz,

$$\mathrm{KL}\Big(\mathbb{P}_\phi^{z+\delta z} \,\Big\|\, \mathbb{P}_\phi^z\Big) = \frac{1}{2}\delta z^\top \mathcal{I}_\phi^{\mathrm{path}}(z)\delta z + o(\|\delta z\|^2).$$

Thus, $\mathcal{I}_\phi^{\mathrm{path}}(z)$ is the local Fisher–Rao metric induced by KL divergence on the conditional reverse-SDE path laws (Blessing et al., 2025). The tangent space $T_{\mathbb{P}_\phi^z}\mathcal{M}_\phi$ captures first-order perturbations of the conditional path law around $\mathbb{P}_\phi^z$, with $\mathcal{I}_\phi^{\mathrm{path}}(z)$ defining the local metric on this space. We further discuss the parameter- and design-space structure of this geometry in § G.3.

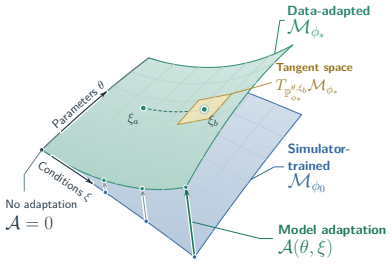

Figure 3: Conditional path-law geometry under fine-tuning.

3.4 COMPOSITIONAL SAMPLING

A simformer can represent arbitrary conditionals of a fixed joint distribution, but explicitly representing each design–observation pair in the joint makes the modeled state grow with the number of experiments and ties inference to a fixed number of observation tokens. Instead, for each observation (y_j, ξ_j) we use the posterior score $s_\phi(\theta_t, t; y_j, \xi_j) \approx \nabla_{\theta_t} \log p_t(\theta_t \mid y_j, \xi_j)$, and combine these per-observation scores at inference to approximate the posterior conditioned on $\mathcal{D}_n = \{(y_j, \xi_j)\}_{j=1}^n$. Since the same score model is reused for every (y_j, ξ_j), the number and subset of observations can vary without changing or retraining the learned joint model. To achieve this, we first review the discrete F-NPSE construction of Geffner et al. (2023), then introduce a continuous-time extension.

Discrete compositional sampling. Under conditional independence given θ, the target posterior factorizes as $p(\theta \mid \mathcal{D}_n) \propto p(\theta)^{1-n} \prod_{j=1}^n p(\theta \mid y_j, \xi_j)$. F-NPSE exploits this factorization by defining intermediate distributions whose scores combine the per-observation posterior scores above with the corresponding prior-score correction. In the discrete diffusion probabilistic modeling (DDPM) setting (Ho et al., 2020), k indexes diffusion steps and α_k denotes one-step signal retention, with signal scaling $\sqrt{\alpha_k}$ and added noise variance $1 - \alpha_k$ per coordinate. The resulting normalized reverse transition can be written as

$$\widetilde{p}_k(\theta_{k-1} \mid \theta_k, \mathcal{D}_n) = \mathcal{N}\left(\theta_{k-1}; \mu_k^{\mathrm{G}}, \frac{1-\alpha_k}{\kappa_k^{(n)}} I\right), \qquad \kappa_k^{(n)} = 1 + (n-1)(1-\alpha_k), \tag{5}$$

where μ_k^{G} combines the learned reverse scores for each (y_j, ξ_j) with the prior-score correction implied by the posterior factorization above; its full form and assumptions are given in § A.2. The transition variance in Equation (5) decreases with the number of observations n and approaches $1/n$ as $\alpha_k \to 0$. This transition-level contraction motivates a narrower cumulative noising path than the single-observation VP process, echoing the n^{-1} covariance contraction of Bernstein–von Mises theory (van der Vaart, 1998, Ch. 10).

From discrete to continuous dynamics. To construct a continuous-time noising process, consider a hypothetical draw θ_0 from the desired posterior. Ordinary VP noising attenuates this draw by $\sqrt{\alpha(t)}$ and adds Gaussian noise with variance $1 - \alpha(t)$ per coordinate, where $\alpha(t)$ denotes cumulative squared signal retention. Motivated by the discrete contraction in Eq. (5), we retain this mean attenuation but prescribe a smaller accumulated noise variance $V_n(t) = (1 - \alpha(t))/\kappa^{(n)}(t)$, where $\kappa^{(n)}(t) = 1 + (n-1)(1-\alpha(t))$. This introduces observation-count dependence without estimating the posterior covariance. Keeping the ordinary VP SDE drift $f(t) = \dot{\alpha}(t)/(2\alpha(t))$ (Song et al., 2021), the variance evolution $\dot{V}_n(t) = 2f(t)V_n(t) + g_n(t)^2$ then determines the diffusion coefficient for this noising process

$$g_n(t)^2 = \beta(t) \frac{1 + (n-1)\,(1-\alpha(t))^2}{[\kappa^{(n)}(t)]^2}, \qquad \beta(t) = -\frac{\dot{\alpha}(t)}{\alpha(t)}. \tag{6}$$

The coefficient satisfies $g_n(t)^2 \le \beta(t)$, with $g_1(t)^2 = \beta(t)$ and $g_n(t)^2/\beta(t) \to 1/n$ as $\alpha(t) \to 0$. We call the resulting sampler F-NPSE-SDE. It uses $g_n(t)$ in the reverse-SDE predictor, which advances between noise levels, and in L optional unadjusted Langevin corrections, which explore the current compositional marginal at fixed diffusion time. The smaller coefficient moderates score-driven drift and noise, reducing the risk of overshoot in stiff dynamics. Our controlled exact-score Gaussian example shows that these numerical issues arise even in a simple model and that the schedule reduces both step-size sensitivity and sampling error (§ A.5). This yields an inexpensive schedule depending only on diffusion time and observation count, with its derivation in § A.3.

Practical scaling. F-NPSE-SDE requires only score evaluations, avoiding the score Jacobians of JAC and auxiliary covariance estimation of GAUSS (Linhart et al., 2026). These costs become especially important in HBDS (§ 3.5), where compositional sampling is repeated within each block update. Computational complexity and observed runtime and memory limitations on the BMP model are discussed in § A.8.

3.5 HIERARCHICAL BLOCKWISE DIFFUSION SAMPLING

Naïve compositional sampling, which we refer to as *pooled compositional sampling*, treats all experimental observations as if they share a single posterior distribution. This ignores the fact that different groups of experiments may induce distinct posteriors, leading to suboptimal inference. For example, the BMP dataset comprises $K = 4$ cell lines consisting of a parent and three receptor-knockdown lines. Each line has its own receptor state R^k, shared across its ligand-dose experiments, while the biophysical parameters θ are shared across all four lines (Appendix C). For K groups with unique R parameters, with designs $\xi^{1:K}$ and observations $y^{1:K}$, we want to target the posterior $p(\theta, R^{1:K} \mid y^{1:K}, \xi^{1:K})$, retaining shared parameters θ and a local latent state R^k per group (Gelman et al., 1995). HBDS samples this target by initializing a shared parameter state and separate receptor states. At every reverse-diffusion time, each R^k is updated with its own (y^k, ξ^k) and current global θ, then θ is updated using all receptor blocks. Repeating yields samples with one globally consistent parameter vector and K distinct local latent states without training a separate hierarchical estimator or tokenizing the complete dataset. The HBDS algorithm and analytic example are in §§ B and B.1.

4 EXPERIMENTS

We evaluate $g_n(t)$ on an exact toy Gaussian model and the Simple Likelihood, Complex Posterior (SLCP) Papamakarios et al. (2019) benchmarks (§ 4.1), hierarchical inference on a controlled toy (§ 4.2), and misspecification-aware fine-tuning (§ 4.3). The BMP application (§ 4.4) combines likelihood correction with HBDS across 940 observations, assessing predictive improvement and its relationship to design-localized path divergence.

4.1 EVALUATING F-NPSE-SDE

Exact-score Gaussian benchmark. We compare F-NPSE-SDE with GAUSS, JAC, and Langevin (Linhart et al., 2026), and the original F-NPSE (Geffner et al., 2023). The 10-dimensional Gaussian model (§ A.5) has correlation $\rho = 0.8$ and exact compositional scores and posteriors. Here, n counts observations, T predictor steps, and L Langevin corrections per predictor. At $n = 100$ and $T = 400$, Table 1 shows that F-NPSE-SDE is competitive with GAUSS and JAC in maximum sliced Wasserstein distance, matches F-NPSE, and improves over Langevin. Full sweep settings and results appear in § A.6.

SLCP benchmark. We reproduce Linhart et al. (2026) using three independently trained single-observation score networks and $n \in \{1, 14, 30\}$. Within each comparison, all samplers use the same frozen network. At $n = 30$, GAUSS and JAC remain the most accurate, while F-NPSE-SDE improves over Langevin and F-NPSE in sliced Wasserstein distance and MMD (Table 1), without JAC's score Jacobians or GAUSS's auxiliary DDIM rollouts. Training, stabilization, and results across observation counts and network seeds are detailed in § A.7.

Table 1: Comparison of compositional samplers on the Gaussian benchmark ($n = 100$, $T = 400$) and SLCP ($n = 30$). Entries are mean ± standard deviation over five seeds for Gaussian and 25 test instances using one network seed for SLCP.

	Gaussian ($T = 400$, $n = 100$)	**SLCP** ($n = 30$)		
Method	**max-sW** ↓	**sW** ↓	**MMD** ↓	**C2ST** → 0.5
GAUSS	0.24 ± 0.23	0.45 ± 0.18	0.05 ± 0.16	0.97 ± 0.04
JAC	0.26 ± 0.22	0.53 ± 0.19	0.05 ± 0.14	0.95 ± 0.05
Langevin	0.52 ± 0.50	0.81 ± 0.28	0.13 ± 0.22	0.97 ± 0.03
F-NPSE	0.29 ± 0.24	0.83 ± 0.29	0.11 ± 0.13	0.98 ± 0.03
F-NPSE-SDE ($L = 1$)	0.29 ± 0.25	0.74 ± 0.31	0.10 ± 0.15	0.98 ± 0.03
F-NPSE-SDE ($L = 5$)	0.31 ± 0.27	0.73 ± 0.28	0.10 ± 0.15	0.98 ± 0.03

4.2 Evaluating pooled compositional sampling and HBDS

Hierarchical toy model. We first isolate the effect of hierarchical structure in a controlled model with a shared continuous parameter $\theta \in \mathbb{R}$ and group-specific latent states $z_g \in \{-1, +1\}$,

$$z_g \sim \mathrm{Rad}\left(\tfrac{1}{2}\right), \qquad \theta \sim \mathcal{N}(0, 1), \qquad x_{gi} \mid \theta, z_g \sim \mathcal{N}(5z_g + \theta, 1).$$

Here $\mathrm{Rad}(\frac{1}{2})$ assigns equal probability to the two signs. We compare learned-score samplers using $n = 20$ observations generated with $\theta = 1$, and ten samples from $z = +1$ and $z = -1$. HBDS permits each group to select its own latent mode, whereas pooled compositional sampling ties all groups to a single latent state. Figure 4 shows pooled sampling cannot represent groups requiring different latent modes, while HBDS maintains separate group-specific states together with a shared θ. The analytic posterior structure and additional score-based diagnostics are given in § B.1. We further evaluate HBDS on the BMP model in § 4.4.

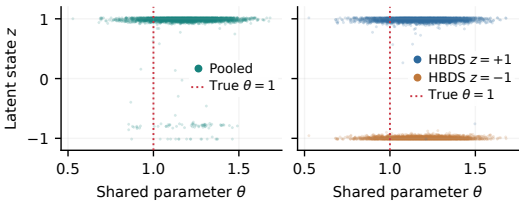

Figure 4: (*Left*) Pooled sampling uses one shared latent. (*Right*) HBDS retains separate group latents with a shared θ. Colors identify the generating groups; plotted latent states are continuous sampler outputs. Dotted lines mark the generating $\theta = 1$.

4.3 Fine-tuning under simulator misspecification

Likelihood correction and path-divergence localization. To study how fine-tuning responds to model misspecification and whether path divergence localizes the resulting adaptation, we use a toy linear-Gaussian simulator $y \mid \theta, \xi \sim \mathcal{N}(\xi^\top \theta, \sigma^2)$ with $\theta \sim \mathcal{N}(0, I_2)$, designs $\xi \in [-2, 2]^2$, and $\sigma = 0.2$. At a fixed generating parameter $\theta^\star$, the noise-free observation response is $y_o(\xi) = \xi^\top \theta^\star + h(\xi)$, where h is a localized Gaussian bump omitted from the simulator. We fine-tune on ten observations with Gaussian noise of variance σ^2, then assess where the correction helps or hurts across design space. We then estimate the PT and FT likelihood means from 16 samples per design and measure their absolute errors against $y_o(\xi)$, giving the pointwise error maps $e_{\mathrm{PT}}(\xi)$ and $e_{\mathrm{FT}}(\xi)$ in Figure 5. We measure mean absolute error (MAE) averages over 1,681 grid points. Fine-tuning reduces error near the bump but adds errors elsewhere, raising MAE from 0.14 to 0.15. However, we employ autoguidance Karras et al. (2024) to blend the PT & FT models to lower MAE to 0.12 (Figure 12). Without guidance, mean $\mathcal{A}(\theta^{\mathrm{LSR}}, \xi)$ inside the misspecified region ($h \geq 0.1$) is 4.73 times that outside and correlates with h (Pearson $r = 0.75$). Thus, divergence localizes adaptation, while error maps distinguish correction from over-adaptation. Regularization and guidance sweeps are in § G.8.

SBI misspecification comparison. On BMP (§ 4.4), we compare with Flow Matching Corrected Posterior Estimation (FMCPE), which corrects an NPE using calibration pairs (Ruhlmann et al., 2026), here constructed from LSR fits. FMCPE uses subset-specific posterior models, whereas FT also adapts the likelihood and supports flexible conditioning, composition, and HBDS. Across the FMCPE observation- and LSR-budget sweep, FT achieves the lowest mean median predictive distance, while FMCPE-LSR has lower best-draw RMSE (Table 7). These metrics evaluate predictions through the original simulator and remain subject to its misspecification. Protocols, PT comparisons, and calibration diagnostics are in § G.7.

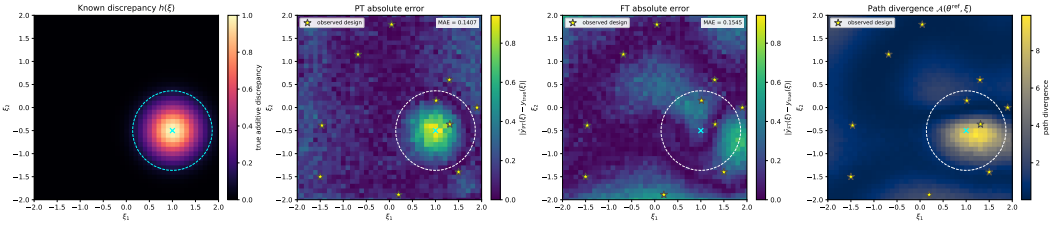

Figure 5: Likelihood adaptation and prediction error in the analytic toy at a fixed LSR reference. (*Left to right*) Known discrepancy $h(\xi)$, PT absolute error $e_{\mathrm{PT}}(\xi)$, FT absolute error $e_{\mathrm{FT}}(\xi)$, and path divergence $\mathcal{A}(\theta^{\mathrm{ref}}, \xi)$.

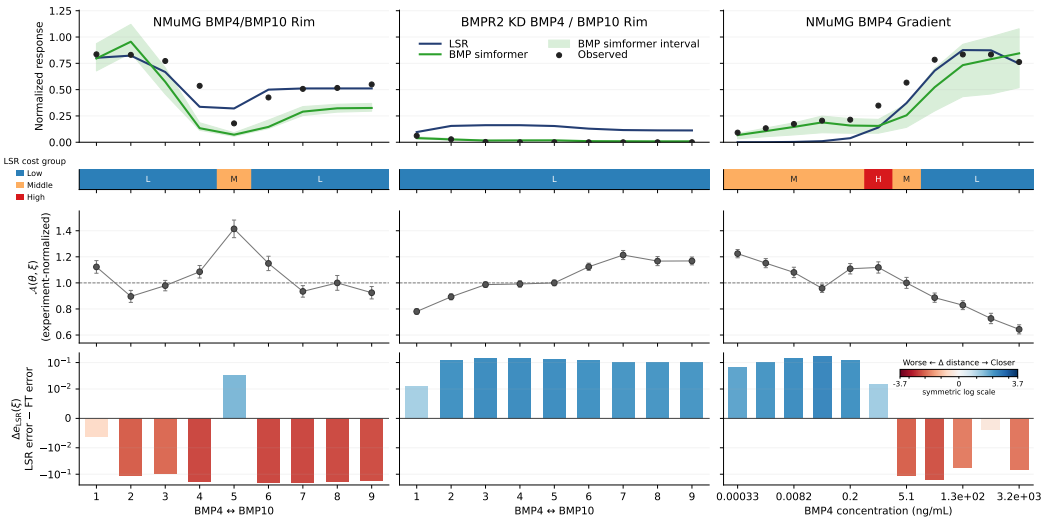

Figure 6: BMP predictions and design-localized adaptation. (*Left*) BMP4/BMP10 competition in NMuMG. (*Middle*) The same competition in BMPR2 KD. (*Right*) BMP4 titration in NMuMG. From top to bottom: LSR and FT predictions against observations; low (L), middle (M), and high (H) LSR cost groups; experiment-normalized path divergence $\mathcal{A}(\theta, \xi)$; and reduction in absolute prediction error from LSR to the FT bootstrap median.

4.4 BMP APPLICATION

We combine likelihood fine-tuning with HBDS on 940 observations across four cell lines, inferring shared biophysical parameters and cell-line-specific receptor states (model and priors: §§ C and D; training: §§ F and G). To assess likelihood correction at the fixed LSR reference, we define $\Delta e_{\mathrm{LSR}}(\xi)$ as the LSR simulator's absolute error against $y_o(\xi)$ minus that of the FT bootstrap median. The bottom row of Figure 6 shows this reduction on a symmetric-logarithmic axis (linear near zero), with positive values favoring FT. FT improves predictions throughout the BMPR2-knockdown competition series and at lower BMP4 doses. In the BMP4 titration, these improvements generally coincide with larger path divergence, whereas adaptation in the NMuMG competition series often increases error. Together, these quantities distinguish where the likelihood changes from whether that change improves predictions. LSR cost-group analysis further identifies conditions where parameter fits with higher global cost perform better locally (§ G.9). For the FT model with $\lambda = 5 \times 10^{-4}$, HBDS yields RMSE 0.34 and median distance 14.94, compared with 0.36 and 66.77 under pooled sampling. Sampler/FiLM ablations, regularization sweeps, and θ^{LSR} comparison appear in Tables 3 to 5.

5 DISCUSSION

Together, these methods enable flexible diffusion SBI across changing observation sets, hierarchical latent structure, and simulator misspecification, with the BMP model providing a large-scale scientific stress test. Rather than requiring a PT SBI model to anticipate this downstream structure, our methods adapt it to the problem at hand.

Limitations and future work. *Compositional sampling and HBDS* both rely on approximations to intermediate-time posteriors. More accurate intermediate targets, combined with learned flow maps for faster sampling (Boffi et al., 2025), could improve both compositional inference and the repeated block updates in HBDS. *Fine-tuning* introduces a separate set of challenges. Adaptation is anchored at a fixed reference parameter θ^{ref} and can improve misspecified regions while degrading regions that were already well modeled. The path-space formulation provides tools for addressing this directly, e.g. Fisher geometry over (θ, ξ) suggests design-dependent adaptation, particle- or geometry-based refinement of θ^{ref}, and diffusion-based Bayesian experimental design over ξ (Iollo et al., 2025).

REFERENCES

Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. ISSN 0304-4149. doi: 10.1016/0304-4149(82)90051-5.

Yaron E Antebi, James M Linton, Heidi Klumpe, Bogdan Bintu, Mengsha Gong, Christina Su, Reed McCardell, and Michael B Elowitz. Combinatorial signal perception in the bmp pathway. *Cell*, 170(6):1184–1196, 2017.

Jonas Arruda, Niels Bracher, Ullrich Köthe, Jan Hasenauer, and Stefan T. Radev. Diffusion models in simulation-based inference: A tutorial review. arXiv preprint arXiv:2512.20685, 2025.

Jonas Arruda, Vikas Pandey, Catherine Sherry, Margarida Barroso, Xavier Intes, Jan Hasenauer, and Stefan T. Radev. Compositional amortized inference for large-scale hierarchical bayesian models. In *International Conference on Learning Representations*, 2026.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 41–48. Association for Computing Machinery, 2009. doi: 10.1145/1553374.1553380.

Christopher M. Bishop. Mixture density networks. Technical Report NCRG/94/004, Neural Computing Research Group, Aston University, Birmingham, UK, 1994.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, pp. 4965–4987, 2024.

Denis Blessing, Julius Berner, Lorenz Richter, Carles Domingo-Enrich, Yuanqi Du, Arash Vahdat, and Gerhard Neumann. Trust region constrained measure transport in path space for stochastic optimal control and inference. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Jan Boelts. Model misspecification in simulation-based inference: Recent advances and open challenges. ICLR Blogposts 2026, April 2026. Published April 27, 2026.

Nicholas M. Boffi, Michael S. Albergo, and Eric Vanden-Eijnden. Flow map matching with stochastic interpolants: A mathematical framework for consistency models. *Transactions on Machine Learning Research*, 2025.

Patrick Cannon, Daniel Ward, and Sebastian M. Schmon. Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*, 2022. doi: 10.48550/arxiv.2209.01845.

Pierre Cesar, Sofya Dymchenko, Abhishek Purandare, and Bruno Raffin. Learning where to simulate: Generative active sampling for online PDE surrogate training. arXiv preprint arXiv:2606.09949, 2026.

Giovanni Charles, Cosmo Santoni, Seth Flaxman, and Elizaveta Semenova. Tokenised flow matching for hierarchical simulation based inference. arXiv preprint arXiv:2604.20723, 2026.

Kevin Clark, Paul Vicol, Kevin Swersky, and David Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *International Conference on Learning Representations*, pp. 4793–4822, 2024.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, November 2020. ISSN 0027-8424. doi: 10.1073/pnas.1912789117. arXiv: 1911.01429 Publisher: Proceedings of the National Academy of Sciences.

Riccardo De Santi, Marin Vlastelica, Ya-Ping Hsieh, Zebang Shen, Niao He, and Andreas Krause. Flow density control: Generative optimization beyond entropy-regularized fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 11056–11088, 2025.

Daniel Dimitrov, Stefan Schrod, Martin Rohbeck, and Oliver Stegle. Interpretation, extrapolation and perturbation of single cells. *Nature Reviews Genetics*, pp. 1–22, 2026.

Robert M. Dirks, Justin S. Bois, Joseph M. Schaeffer, Erik Winfree, and Niles A. Pierce. Thermodynamic analysis of interacting nucleic acid strands. *SIAM Review*, 49(1):64–88, 2007. doi: 10.1137/060651100.

Carles Domingo i Enrich, Michal Drozdzal, Brian Karrer, and Ricky T. Q. Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In *International Conference on Learning Representations*, pp. 53791–53846, 2025.

Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2771–2781. PMLR, 2020.

Alain Durmus and Éric Moulines. Nonasymptotic convergence analysis for the unadjusted Langevin algorithm. *The Annals of Applied Probability*, 27(3):1551–1587, 2017. doi: 10.1214/16-AAP1238.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. DPOK: Reinforcement learning for fine-tuning text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 79858–79885, 2023.

Richard Gao, Michael Deistler, and Jakob H Macke. Generalized bayesian inference for scientific simulators via amortized cost estimation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 80191–80219, 2023.

Tomas Geffner, George Papamakarios, and Andriy Mnih. Compositional score modeling for simulation-based inference. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11098–11116. PMLR, 23–29 Jul 2023.

Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 1995.

Manuel Gloeckler, Michael Deistler, Christian Dietrich Weilbach, Frank Wood, and Jakob H. Macke. All-in-one simulation-based inference. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15735–15766. PMLR, 21–27 Jul 2024.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pp. 2404–2414. PMLR, 2019.

Daniel Habermann, Marvin Schmitt, Lars Kühmichel, Andreas Bulling, Stefan T. Radev, and Paul-Christian Bürkner. Amortized bayesian multilevel models. *Bayesian Analysis*, pp. 1–30, 2025. doi: 10.1214/25-BA1570.

Aaron J. Havens, Benjamin Kurt Miller, Bing Yan, Carles Domingo-Enrich, Anuroop Sriram, Daniel S. Levine, Brandon M. Wood, Bin Hu, Brandon Amos, Brian Karrer, Xiang Fu, Guan-Horng Liu, and Ricky T. Q. Chen. Adjoint sampling: Highly scalable diffusion samplers via adjoint matching. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 22204–22237. PMLR, 2025.

Lukas Heinrich, Siddharth Mishra-Sharma, Chris Pollard, and Philipp Windischhofer. Hierarchical neural simulation-based inference over event ensembles. *Transactions on Machine Learning Research*, 2024, 2024. ISSN 2835-8856.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Daolang Huang, Ayush Bharti, Amauri Souza, Luigi Acerbi, and Samuel Kaski. Learning robust statistics for simulation-based inference under model misspecification. In *Advances in Neural Information Processing Systems*, volume 36, pp. 7289–7310, 2023a.

Peide Huang, Xilun Zhang, Ziang Cao, Shiqi Liu, Mengdi Xu, Wenhao Ding, Jonathan Francis, Bingqing Chen, and Ding Zhao. What went wrong? closing the sim-to-real gap via differentiable causal discovery. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 734–760. PMLR, 2023b.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(24):695–709, 2005.

Jacopo Iollo, Christophe Heinkelé, Pierre Alliez, and Florence Forbes. Bayesian experimental design via contrastive diffusions. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 73125–73151, 2025.

Cristian Perez Jensen, Luca Schaufelberger, Riccardo De Santi, Kjell Jorner, and Andreas Krause. Value matching: Scalable and gradient-free reward-guided flow adaptation. In *International Conference on Learning Representations*, 2026.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Heidi E Klumpe, Matthew A Langley, James M Linton, Christina J Su, Yaron E Antebi, and Michael B Elowitz. The context-dependent, combinatorial logic of bmp signaling. *Cell systems*, 13(5):388–407, 2022.

Julia Linhart, Alexandre Gramfort, and Pedro Rodrigues. ℓ-C2ST: Local diagnostics for posterior approximations in simulation-based inference. In *Advances in Neural Information Processing Systems*, volume 36, pp. 56384–56410, 2023.

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.

Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jan-Matthis Lueckmann, Giacomo Bassetto, Theofanis Karaletsos, and Jakob H. Macke. Likelihood-free inference with emulator networks. In *Proceedings of The 1st Symposium on Advances in Approximate Bayesian Inference*, volume 96 of *Proceedings of Machine Learning Research*, pp. 32–53. PMLR, 2019.

Pierre Marion, Anna Korba, Peter Bartlett, Mathieu Blondel, Valentin De Bortoli, Arnaud Doucet, Felipe Llinares-López, Courtney Paquette, and Quentin Berthet. Implicit diffusion: Efficient optimization through stochastic sampling. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 1999–2007. PMLR, 2025.

Benjamin Kurt Miller, Christoph Weniger, and Patrick Forré. Contrastive Neural Ratio Estimation. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

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.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010.

George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with Bayesian conditional density estimation. In *Advances in Neural Information Processing Systems*, pp. 1036–1044, 2016.

George Papamakarios, David C. Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 837–848. PMLR, 2019.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11671.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. arXiv preprint arXiv:2310.03739, 2023. Preprint withdrawn and subsumed by arXiv:2407.08737.

Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996. doi: 10.2307/3318418.

Pierre-Louis Ruhlmann, Michael Arbel, Florence Forbes, and Pedro L. C. Rodrigues. Flow matching calibration for simulation-based inference under model misspecification. In *International Conference on Machine Learning*, 2026.

Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Detecting model misspecification in amortized bayesian inference with neural networks. In *Pattern Recognition. DAGM GCPR 2023*, volume 14264 of *Lecture Notes in Computer Science*, pp. 541–557. Springer, 2024. doi: 10.1007/978-3-031-54605-1_35.

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. PMLR, 2024.

Freddie Bickford Smith, Jannik Kossen, Eleanor Trollope, Mark van der Wilk, Adam Foster, and Tom Rainforth. Rethinking aleatoric and epistemic uncertainty. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Henry D. Smith, Nathaniel L. Diamant, and Brian L. Trippe. Calibrating generative models to distributional constraints. In *International Conference on Machine Learning*, 2026.

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.

Christina J Su, Arvind Murugan, James M Linton, Akshay Yeluri, Justin Bois, Heidi Klumpe, Matthew A Langley, Yaron E Antebi, and Michael B Elowitz. Ligand-receptor promiscuity enables cellular addressing. *Cell systems*, 13(5):408–425, 2022.

Camille Touron, Gabriel V. Cardoso, Julyan Arbel, and Pedro L. C. Rodrigues. Theoretical guidelines for annealed Langevin dynamics in compositional simulation-based inference. arXiv preprint arXiv:2605.21253, 2026.

Masatoshi Uehara, Yulai Zhao, Tommaso Biancalani, and Sergey Levine. Understanding reinforcement learning-based fine-tuning of diffusion models: A tutorial and review. arXiv preprint arXiv:2407.13734, 2024a.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M. Tseng, Tommaso Biancalani, and Sergey Levine. Fine-tuning of continuous-time diffusion models as entropy-regularized control. arXiv preprint arXiv:2402.15194, 2024b.

Masatoshi Uehara, Yulai Zhao, Kevin Black, Ehsan Hajiramezanali, Gabriele Scalia, Nathaniel Lee Diamant, Alex M. Tseng, Sergey Levine, and Tommaso Biancalani. Feedback efficient online fine-tuning of diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 48892–48918. PMLR, 2024c.

Masatoshi Uehara, Yulai Zhao, Ehsan Hajiramezanali, Gabriele Scalia, Gokcen Eraslan, Avantika Lal, Sergey Levine, and Tommaso Biancalani. Bridging model-based optimization and generative modeling via conservative fine-tuning of diffusion models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 127511–127535, 2024d.

A. W. van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 1998. doi: 10.1017/CBO9780511802256.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Comput.*, 23(7):1661–1674, July 2011. ISSN 0899-7667. doi: 10.1162/NECO_a_00142.

Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian Schmon. Robust neural posterior estimation and statistical model criticism. In *Advances in Neural Information Processing Systems*, volume 35, pp. 33845–33859, 2022.

Antoine Wehenkel, Juan L. Gamella, Ozan Sener, Jens Behrmann, Guillermo Sapiro, Joern-Henrik Jacobsen, and Marco Cuturi. Addressing misspecification in simulation-based inference through data-driven calibration. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 65949–65980. PMLR, 2025.

Christian Weilbach, William Harvey, and Frank Wood. Graphically structured diffusion models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Christopher Williams, Andrew Campbell, Arnaud Doucet, and Saifuddin Syed. Score-optimal diffusion schedules. In *Advances in Neural Information Processing Systems*, volume 37, pp. 107960–107983, 2024.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 15903–15935, 2023.

APPENDICES

TABLE OF CONTENTS

A Compositional sampling details **16**
A.1 Existence of cumulative transition score 16
A.2 Derivation of the reverse transition kernel 16
A.3 Deriving the continuous-time diffusion coefficient 18
A.4 Langevin refinement and numerical accuracy 20
A.5 Experimental evaluation 21
A.6 Exact-score Gaussian benchmark 24
A.7 SLCP benchmark 24
A.8 Computational cost of compositional samplers 24
A.9 Compositional sampling related works 25

B Hierarchical blockwise diffusion sampling details **26**
B.1 Toy simulator model example 27
B.2 Hierarchical SBI related works 28

C BMP signaling pathway mathematical model **28**

D Cell receptor hierarchical modeling **29**

E Additional BMP experimental results **31**
E.1 Calibrated receptor diffusion 31
E.2 BMP posterior-sampling ablations 31

F Pretraining details **32**

G Fine-tuning and path-space details **33**
G.1 Path-space formulation and Girsanov's theorem 34
G.2 Stochastic optimal control connection 34
G.3 Fisher–Rao optimization analysis 35
G.4 Related work on simulator misspecification in SBI 37
G.5 Related work on diffusion fine-tuning 38
G.6 Embedding-based transfer learning 40
G.7 FMCPE-LSR baseline 40
G.8 Design-dependent misspecification toy 41
G.9 BMP misspecification evaluation 43

H AI use statement **48**

A Compositional sampling details

A.1 Existence of cumulative transition score

We begin with a proposition for the existence of a cumulative transition score.

Proposition A.1 (Existence of a cumulative transition score)**.** *Under the reverse–forward consistency assumption in Eq. (7), we construct a cumulative transition $\tilde{p}_k(\theta_{k-1}|\theta_k)$ whose score combines the per-observation reverse scores with the common forward-kernel correction. This gives the compositional transition used below.*

Proof (existence of a cumulative transition kernel). Fix a discrete step $k \in \{1, \dots, K\}$. For each observation y_j let $p_k^j(\theta_k)$ denote the marginal at diffusion time t_k and $q_k(\theta_k|\theta_{k-1})$ the common forward kernel. Assume the (exact) reverse–forward consistency

$$p_k^j(\theta_k)\, p_k^j(\theta_{k-1}|\theta_k) \;=\; p_{k-1}^j(\theta_{k-1})\, q_k(\theta_k|\theta_{k-1}) \qquad \text{for each } j \in \{1, \dots, n\}. \tag{7}$$

We seek a transition kernel $\tilde{p}_k(\theta_{k-1}|\theta_k)$ that satisfies the composition condition (cf. Eq. (24) of Geffner et al. (2023))

$$\tilde{p}_{k-1}(\theta_{k-1}) \;=\; \int \tilde{p}_k(\theta_k)\, \tilde{p}_k(\theta_{k-1}|\theta_k)\, d\theta_k. \tag{8}$$

where $\tilde{p}_k(\theta_k) \;=\; Z_k^{-1} \prod_j^n p_k^j(\theta_k)$. Following Geffner et al. (2023), we rewrite Eq. (8) (cf. Eqs. (25)–(26) therein) by using Eq. (7) to express ratios of marginals and reverse kernels:

$$\tilde{p}_{k-1}(\theta_{k-1}) = \int d\theta_k\, p_k^1(\theta_k)\, \frac{q_k(\theta_k|\theta_{k-1})}{p_k^2(\theta_{k-1}|\theta_k)} \cdots \frac{q_k(\theta_k|\theta_{k-1})}{p_k^n(\theta_{k-1}|\theta_k)}\, \frac{Z_k}{Z_{k-1}}\, \tilde{p}_k(\theta_{k-1}|\theta_k).$$

One way to satisfy this equality, as noted in Geffner et al. (2023), is to set

$$\tilde{p}_k(\theta_{k-1}|\theta_k) = p_k^1(\theta_{k-1}|\theta_k)\, \frac{Z_k}{Z_{k-1}}\, \frac{p_k^2(\theta_{k-1}|\theta_k)}{q_k(\theta_k|\theta_{k-1})} \cdots \frac{p_k^n(\theta_{k-1}|\theta_k)}{q_k(\theta_k|\theta_{k-1})}. \tag{9}$$

Hence, under exact reversibility, a cumulative transition kernel of the form exists, establishing the claim. □

This derivation follows the compositional construction of Geffner et al. (2023), using the per-observation reverse–forward consistency relation in Eq. (7). We spell out the intermediate algebra to clarify how the individual reverse kernels and common forward kernel enter the cumulative transition, then derive its score under a Gaussian parameterization in § A.2. Our aim is to make the cumulative transition score explicit under these assumptions, rather than to establish an exact reverse kernel for the joint posterior.

A.2 Derivation of the reverse transition kernel

We next derive the analytic form of the cumulative transition score under a Gaussian diffusion parameterization. Starting from the cumulative reverse transition (cf. Eq. (9))

$$\tilde{p}_k(\theta_{k-1}|\theta_k) = p_k^1(\theta_{k-1}|\theta_k)\, \frac{Z_k}{Z_{k-1}} \prod_{j=2}^{n} \frac{p_k^j(\theta_{k-1}|\theta_k)}{q_k(\theta_k|\theta_{k-1})},$$

where Z_k and Z_{k-1} are normalization constants ensuring integration to one. We retain the DDPM notation, with k indexing discrete steps at diffusion times t_k and α_k denoting one-step signal retention, distinct from the continuous-time cumulative factor $\alpha(t)$ used in § A.3. The Gaussian kernels are parameterized as

$$p_k^j(\theta_{k-1}|\theta_k) = \mathcal{N}\big(\theta_{k-1};\, \mu_k^j, (1-\alpha_k)I\big), \qquad q_k(\theta_k|\theta_{k-1}) = \mathcal{N}\big(\theta_k;\, \sqrt{\alpha_k}\,\theta_{k-1}, (1-\alpha_k)I\big), \tag{10}$$

and we differentiate $\log \tilde{p}_k(\theta_{k-1}|\theta_k)$ with respect to θ_{k-1} to get

$$\nabla_{\theta_{k-1}} \log \tilde{p}_k(\theta_{k-1}|\theta_k) = \sum_{j=1}^{n} \nabla_{\theta_{k-1}} \log p_k^j(\theta_{k-1}|\theta_k) - (n-1)\, \nabla_{\theta_{k-1}} \log q_k(\theta_k|\theta_{k-1}), \tag{11}$$

where the normalization constants cancel after differentiation. For the reverse Gaussian kernels, the mean can be written in terms of the learned score function as

$$\mu_k^j = \frac{1}{\sqrt{\alpha_k}}\, \theta_k + \frac{1-\alpha_k}{\sqrt{\alpha_k}}\, s_\phi^j(\theta_k, t_k), \tag{12}$$

where $s_\phi^j(\theta_k, t_k)$ denotes the estimated score for observation y_j. Before substituting the score of the log of these kernels into Eq. (11), we recall two Gaussian score identities. For a Gaussian density $\mathcal{N}(x; \mu, \Sigma)$, the score with respect to the argument x is

$$\nabla_x \log \mathcal{N}(x; \mu, \Sigma) = -\Sigma^{-1}(x - \mu),$$

while the score with respect to the mean μ is

$$\nabla_\mu \log \mathcal{N}(x; \mu, \Sigma) = \Sigma^{-1}(x - \mu).$$

Applying these identities gives

$$\nabla_{\theta_{k-1}} \log p_k^j(\theta_{k-1}|\theta_k) = \frac{\mu_k^j - \theta_{k-1}}{1-\alpha_k}, \quad \nabla_{\theta_{k-1}} \log q_k(\theta_k|\theta_{k-1}) = \frac{\sqrt{\alpha_k}}{1-\alpha_k}\, \theta_k - \frac{\alpha_k}{1-\alpha_k}\, \theta_{k-1}. \tag{13}$$

We then substitute these into Eq. (11) and group by θ_k and θ_{k-1} terms to form the cumulative transition kernel in terms of the mean values at the transition

$$\nabla_{\theta_{k-1}} \log \tilde{p}_k(\theta_{k-1}|\theta_k) = \sum_{j=1}^{n} \frac{\mu_k^j - \theta_{k-1}}{1-\alpha_k} \; - \; (n-1)\left(\frac{\sqrt{\alpha_k}}{1-\alpha_k}\theta_k - \frac{\alpha_k}{1-\alpha_k}\theta_{k-1}\right) \tag{14}$$

$$= \frac{\sum_{j=1}^{n} \mu_k^j - n\,\theta_{k-1}}{1-\alpha_k} - \frac{(n-1)\sqrt{\alpha_k}}{1-\alpha_k}\, \theta_k + \frac{(n-1)\alpha_k}{1-\alpha_k}\, \theta_{k-1} \tag{15}$$

$$= \frac{\sum_{j=1}^{n} \mu_k^j - (n-1)\sqrt{\alpha_k}\,\theta_k}{1-\alpha_k} - \frac{n - \alpha_k(n-1)}{1-\alpha_k}\, \theta_{k-1}. \tag{16}$$

We can now identify the terms previously used by Geffner et al. (2023) for compositional sampling. Matching the grouped score to the Gaussian identity

$$\nabla_{\theta_{k-1}} \log \mathcal{N}(\theta_{k-1}; \mu_k, \sigma_k^2) = \sigma_k^{-2}\mu_k - \sigma_k^{-2}\theta_{k-1},$$

we identify the coefficients of θ_{k-1} and the constant term as

$$\sigma_k^{-2} = \frac{n - \alpha_k(n-1)}{1-\alpha_k}, \qquad \sigma_k^{-2}\mu_k = \frac{1}{1-\alpha_k}\sum_{j=1}^{n} \mu_k^j - \frac{(n-1)\sqrt{\alpha_k}}{1-\alpha_k}\, \theta_k.$$

Therefore,

$$\mu_k = \frac{\sum_{j=1}^{n} \mu_k^j - (n-1)\sqrt{\alpha_k}\,\theta_k}{n - \alpha_k(n-1)}, \qquad \sigma_k^2 = \frac{1-\alpha_k}{n - \alpha_k(n-1)}, \tag{17}$$

as originally used by Geffner et al. (2023). We can now substitute approximate scores for a final cumulative score expression, $\nabla_{\theta_{k-1}} \log \tilde{p}_k(\theta_{k-1}|\theta_k)$. Using the equivalence between diffusion means and scores (12),

$$\sum_{j=1}^{n} \mu_k^j = \frac{n}{\sqrt{\alpha_k}}\, \theta_k + \frac{1-\alpha_k}{\sqrt{\alpha_k}} \sum_{j=1}^{n} s_\phi^j(\theta_k, t_k),$$

Writing the right-hand side of Eq. (16) as RHS, we substitute the score parameterization to obtain

$$\text{RHS} = \frac{1}{1-\alpha_k}\sum_{j=1}^{n}\mu_k^j - \frac{(n-1)\sqrt{\alpha_k}}{1-\alpha_k}\,\theta_k + \frac{(n-1)\alpha_k - n}{1-\alpha_k}\,\theta_{k-1} \tag{18}$$

$$= \frac{1}{1-\alpha_k}\left(\frac{n}{\sqrt{\alpha_k}}\theta_k + \frac{1-\alpha_k}{\sqrt{\alpha_k}}\sum_{j=1}^{n}s_\phi^j(\theta_k,t_k)\right) - \frac{(n-1)\sqrt{\alpha_k}}{1-\alpha_k}\,\theta_k + \frac{(n-1)\alpha_k - n}{1-\alpha_k}\,\theta_{k-1} \tag{19}$$

$$= \frac{1}{\sqrt{\alpha_k}}\sum_{j=1}^{n}s_\phi^j(\theta_k,t_k) + \left[\frac{n}{(1-\alpha_k)\sqrt{\alpha_k}} - \frac{(n-1)\sqrt{\alpha_k}}{1-\alpha_k}\right]\theta_k + \frac{(n-1)\alpha_k - n}{1-\alpha_k}\,\theta_{k-1} \tag{20}$$

$$= \frac{1}{\sqrt{\alpha_k}}\sum_{j=1}^{n}s_\phi^j(\theta_k,t_k) + \frac{\big(n-(n-1)\alpha_k\big)\theta_k + \sqrt{\alpha_k}\,\big((n-1)\alpha_k - n\big)\theta_{k-1}}{(1-\alpha_k)\sqrt{\alpha_k}}. \tag{21}$$

Finally, we arrive at the cumulative transition score. To apply this insight in practice, we need to define the μ_k and σ_k^2 terms that will be used in a Gaussian transition. Grouping the θ_k and θ_{k-1} terms as before and simplifying we arrive at the mean transition in terms of different scores

$$\mu_k = \frac{1}{\sqrt{\alpha_k}}\theta_k + \frac{1-\alpha_k}{\sqrt{\alpha_k}\big(n-\alpha_k(n-1)\big)}\sum_{j=1}^{n}s_\phi^j(\theta_k,t_k), \qquad \sigma_k^2 = \frac{1-\alpha_k}{n-\alpha_k(n-1)}. \tag{22}$$

Notice that both the aggregate score contribution and the transition variance contain the common factor $n-\alpha_k(n-1)$, which depends jointly on the number of composed observations and the noise schedule. We denote this quantity by

$$\kappa_k^{(n)} := n - \alpha_k(n-1), \tag{23}$$

and refer to it as the *compositional precision factor for n composed observations*. It controls how composition rescales the aggregate score and the variance of the normalized reverse transition.

The mean and variance of this Gaussian reverse kernel define a DDPM-style predictor,

$$\theta_{k-1} = \frac{1}{\sqrt{\alpha_k}}\,\theta_k \;+\; \frac{1-\alpha_k}{\sqrt{\alpha_k}\,\kappa_k^{(n)}}\sum_{j=1}^{n}s_\phi^j(\theta_k,t_k) \;+\; \sqrt{\frac{1-\alpha_k}{\kappa_k^{(n)}}}\,z_k,\;\; z_k\sim\mathcal{N}(0,I). \tag{24}$$

To relate these steps to a continuous VP schedule, take a forward-time grid $0=t_0<\cdots<t_K=1$ and let $\alpha(t)$ denote cumulative squared signal retention, with $\alpha(0)=1$. The one-step factors and their cumulative product satisfy

$$\alpha_k = \frac{\alpha(t_k)}{\alpha(t_{k-1})}, \qquad \prod_{j=1}^{k}\alpha_j = \alpha(t_k).$$

The score network is evaluated at the corresponding continuous time t_k, while α_k controls only the transition between t_{k-1} and t_k.

A.3 Deriving the continuous-time diffusion coefficient

We seek a continuous-time counterpart to the discrete predictor in Eq. (24). Here t denotes continuous diffusion time, $\alpha(t)$ denotes cumulative squared signal retention from the clean state, and a dot denotes differentiation with respect to t. The one-step DDPM factors remain α_k, with discrete steps indexed by k. For a forward process with drift $f(t)\theta$, diffusion coefficient $g(t)$, and corresponding marginal p_t, the reverse SDE admits a predictor-corrector (PC) decomposition (Song et al., 2021),

$$\mathrm{d}\Theta_t = \underbrace{\big(f(t)\theta_t - \tfrac{1}{2}g(t)^2\nabla\log p_t(\theta_t)\big)\,\mathrm{d}t}_{\text{Probability Flow Prediction ODE}} + \underbrace{\big(-\tfrac{1}{2}g(t)^2\nabla\log p_t(\theta_t)\,\mathrm{d}t + g(t)\,\mathrm{d}\widetilde{W}_t\big)}_{\text{Langevin Correction SDE}}. \tag{25}$$

Sampling follows this equation toward decreasing t. The probability-flow term advances samples between noise levels, while the Langevin term provides score-driven exploration at the current level. The shared coefficient $g(t)$ determines the strength of both the score-driven motion and the injected noise. For compositional sampling, we seek an observation-count-dependent choice $g(t) = g_n(t)$ to improve the stability and numerical accuracy of these corrections. We examine its effects and the associated exploration trade-offs in § A.4.

Here we construct $g_n(t)$ by retaining the ordinary VP mean attenuation and prescribing a Gaussian noising variance that contracts with the observation count n, motivated by the discrete variance contraction in Eq. (24). Specifying this conditional kernel does not require knowing the clean posterior or assuming that it is Gaussian. With the VP drift fixed, the diffusion coefficient then follows from the chosen variance evolution.

Deriving the diffusion coefficient $g_n(t)$. Let $x = x_{1:n}$ collect the observed design–response pairs. Suppose we could draw $\theta_0 \sim p_0(\theta \mid x)$ from the clean posterior. Ordinary VP noising would perturb this draw according to

$$\Theta_t \mid \Theta_0 = \theta_0 \sim \mathcal{N}\Big(\sqrt{\alpha(t)}\,\theta_0,\, [1-\alpha(t)]I\Big).$$

Here $\alpha(t)$ is the cumulative squared signal retention, with $\alpha(0) = 1$ and $\dot\alpha(t) = -\beta(t)\alpha(t)$ for the VP noise-rate schedule $\beta(t) \geq 0$. This kernel has accumulated noise variance $1-\alpha(t)$ per coordinate, independent of the number of composed observations n.

The discrete compositional update in Eq. (24) contracts its one-step Gaussian reverse-transition variance by $\kappa_k^{(n)} = 1 + (n-1)(1-\alpha_k)$. We use the same functional form with cumulative retention $\alpha(t)$ to prescribe the conditional noising variance, defining a distinct cumulative contraction factor $\kappa^{(n)}(t)$,

$$\mathrm{Cov}(\Theta_t \mid \Theta_0 = \theta_0) = V_n(t)I, \qquad V_n(t) \coloneqq \frac{1-\alpha(t)}{\kappa^{(n)}(t)}, \qquad \kappa^{(n)}(t) \coloneqq 1+(n-1)(1-\alpha(t)). \tag{26}$$

This cumulative contraction is a modeling prescription, not the infinitesimal limit of the DDPM transition. The factors $\kappa_k^{(n)}$ and $\kappa^{(n)}(t)$ share a functional form but act on one-step and accumulated noise variances, respectively. The prescribed kernel recovers ordinary VP noising at $n = 1$ and reduces the noising variance for $n > 1$.

With $V_n(t)$ specified, the remaining task is to find the diffusion coefficient $g_n(t)$ that produces it while retaining the VP drift $f_n(t) = -\beta(t)/2$. For the forward SDE $\mathrm{d}\Theta_t = f_n(t)\Theta_t\,\mathrm{d}t + g_n(t)\,\mathrm{d}W_t$, applying Itô's formula to each coordinate's squared deviation from its conditional mean and taking expectations gives the variance evolution

$$\dot V_n(t) = 2f_n(t)V_n(t) + g_n(t)^2, \qquad V_n(0) = 0. \tag{27}$$

The first term comes from differentiating the squared deviation under the drift. With the chosen VP drift, $2f_n(t)V_n(t) = -\beta(t)V_n(t)$, so this contribution contracts the existing variance. The second term is Itô's additional contribution from the noise increment's variance $g_n(t)^2\,\mathrm{d}t$, which the ordinary chain rule would miss. The remaining stochastic term has zero expectation. Consequently, the required diffusion variance rate is

$$g_n(t)^2 = \dot V_n(t) - 2f_n(t)V_n(t) = \dot V_n(t) + \beta(t)V_n(t). \tag{28}$$

Thus, $g_n(t)^2$ must both compensate for contraction under the drift and produce the prescribed increase in cumulative variance; it is not simply $V_n(t)$ or $\dot V_n(t)$.

To evaluate this expression, we first differentiate the prescribed variance with respect to cumulative retention, then apply the chain rule to obtain its time derivative. Holding n fixed and temporarily writing α for $\alpha(t)$, the numerator $1-\alpha$ has derivative -1, while the denominator $n-(n-1)\alpha$ has derivative $-(n-1)$. The quotient rule gives

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\alpha}\left(\frac{1-\alpha}{n-(n-1)\alpha}\right) &= \frac{(-1)\big[n-(n-1)\alpha\big] - (1-\alpha)\big[-(n-1)\big]}{[n-(n-1)\alpha]^2} \\
&= \frac{-n+(n-1)\alpha+(n-1)-(n-1)\alpha}{[n-(n-1)\alpha]^2} \\
&= -\frac{1}{[n-(n-1)\alpha]^2}.
\end{aligned}$$

The terms proportional to α cancel, leaving $-n + (n-1) = -1$ in the numerator. Restoring the time dependence and applying the chain rule yields

$$\begin{aligned}\dot{V}_n(t) &= \frac{\mathrm{d}}{\mathrm{d}\alpha}\left(\frac{1-\alpha}{n-(n-1)\alpha}\right)\Big|_{\alpha=\alpha(t)} \dot{\alpha}(t) \\ &= -\frac{\dot{\alpha}(t)}{[\kappa^{(n)}(t)]^2} = \frac{\beta(t)\alpha(t)}{[\kappa^{(n)}(t)]^2},\end{aligned} \tag{29}$$

where the final equality uses $\dot{\alpha}(t) = -\beta(t)\alpha(t)$.

Substituting this derivative and the prescribed variance $V_n(t) = (1-\alpha(t))/\kappa^{(n)}(t)$ into Eq. (28) gives

$$\begin{aligned}g_n(t)^2 &= \frac{\beta(t)\alpha(t)}{[\kappa^{(n)}(t)]^2} + \beta(t)\frac{1-\alpha(t)}{\kappa^{(n)}(t)} \\ &= \beta(t)\frac{\alpha(t) + (1-\alpha(t))\kappa^{(n)}(t)}{[\kappa^{(n)}(t)]^2}.\end{aligned}$$

To simplify the numerator, substitute $\kappa^{(n)}(t) = 1 + (n-1)(1-\alpha(t))$:

$$\begin{aligned}\alpha(t) + (1-\alpha(t))\kappa^{(n)}(t) &= \alpha(t) + (1-\alpha(t))\big[1 + (n-1)(1-\alpha(t))\big] \\ &= \alpha(t) + (1-\alpha(t)) + (n-1)(1-\alpha(t))^2 \\ &= 1 + (n-1)(1-\alpha(t))^2.\end{aligned}$$

Consequently, the diffusion variance rate is

$$g_n(t)^2 = -\frac{\dot{\alpha}(t)}{\alpha(t)}\frac{1 + (n-1)(1-\alpha(t))^2}{[\kappa^{(n)}(t)]^2}. \tag{30}$$

Equivalently, in terms of the VP noise-rate schedule,

$$g_n(t)^2 = \beta(t)\frac{1 + (n-1)(1-\alpha(t))^2}{[1 + (n-1)(1-\alpha(t))]^2}. \tag{31}$$

For $n = 1$, $\kappa^{(1)}(t) = 1$, so the prescribed cumulative variance reduces to $V_1(t) = 1 - \alpha(t)$ and the diffusion coefficient satisfies $g_1(t)^2 = \beta(t)$, recovering the ordinary VP-SDE. For general n, these coefficients realize the chosen linear Gaussian forward process with ordinary VP mean attenuation and reduced cumulative noising variance $V_n(t)$.

The coefficient satisfies $g_n(t)^2 \le \beta(t)$ and depends only on diffusion time and observation count. It therefore provides a composition-dependent sampling scale without score Jacobians or auxiliary covariance estimates. This construction specifies a chosen forward noising process, rather than establishing an exact reverse process for the product bridge.

A.4 Langevin refinement and numerical accuracy

Discretizing the Langevin term in Eq. (25) gives an unadjusted Langevin algorithm (ULA) correction between predictor steps. We now examine how the choice $g(t) = g_n(t)$ affects the stability and accuracy of these updates. We assess how efficiently these corrections explore the compositional marginals $\widetilde{p}_t$ introduced in § 3.4.

Mixing at fixed diffusion time. The predictor advances between noise levels, while Langevin corrections help particles explore each intermediate target before moving on. Their effectiveness depends on both the amount of exploration and the numerical accuracy of the updates. *Strongly curved or anisotropic marginals* can make these corrections sensitive, e.g. rapidly contracting directions may force small steps even when exploration in other directions requires much longer trajectories – a form of numerical stiffness. Starting from the predictor output $\theta_t^{(0)}$ and writing $\widetilde{s}_t = \nabla_\theta \log \widetilde{p}_t$, one ULA correction is

$$\theta_t^{(\ell+1)} = \theta_t^{(\ell)} + \underbrace{\frac{\eta g(t)^2}{2}\widetilde{s}_t(\theta_t^{(\ell)})}_{\text{drift update}} + \underbrace{\sqrt{\eta}\, g(t)\, \varepsilon_\ell}_{\text{diffusion noise}}, \qquad \varepsilon_\ell \overset{\text{iid}}{\sim} \mathcal{N}(0, I_d). \tag{32}$$

Here L is the number of corrections, $\ell = 0, \ldots, L-1$ indexes them, and d is the parameter dimension. The Langevin correction step size $\eta > 0$ controls the size of each update while the diffusion noise level t remains unchanged. This update uses the coefficient $g(t) = g_n(t)$ from Eq. (25). Its score-driven drift $\frac{1}{2}g(t)^2\widetilde{s}_t$ moves toward higher density, while its noise covariance $\eta g(t)^2 I_d$ supplies exploration. The term $f(t)\theta$ belongs to the predictor and is not part of this within-level correction. "Unadjusted" means there is no Metropolis accept/reject step. Consequently, for a fixed nonzero step size, a stable ULA chain generally has an invariant distribution that differs from $\widetilde{p}_t$, introducing discretization bias (Durmus & Moulines, 2017; Roberts & Tweedie, 1996).

Numerical accuracy of Langevin correction. As a one-dimensional Gaussian illustration of this step-size restriction, consider the target $\mathcal{N}(m, v)$ with mean m and variance v. Substituting its score $\widetilde{s}_t(\theta) = -(\theta - m)/v$ into Eq. (32) gives

$$\theta_t^{(\ell+1)} - m = \left(1 - \frac{\eta g(t)^2}{2v}\right)(\theta_t^{(\ell)} - m) + \sqrt{\eta}\, g(t)\, \varepsilon_\ell, \qquad \varepsilon_\ell \sim \mathcal{N}(0, 1). \tag{33}$$

For example, if $\eta g(t)^2 = 6v$, the deterministic part takes a displacement δ from the mean to -2δ. It crosses the mean and ends twice as far away. More generally, two trajectories receiving identical noise have their separation multiplied by $1 - \eta g(t)^2/(2v)$, which amplifies its magnitude when $\eta g(t)^2 > 4v$. Here the score changes at rate $-1/v$ with the parameter, so a narrower target produces a more rapidly varying restoring drift. *Stability therefore depends on the diffusion coefficient and correction step size relative to the target variance, not on score magnitude alone.*

Numerical stability does not by itself imply accurate sampling. Even in the stable range $0 < \eta g(t)^2 < 4v$, this chain has stationary variance $v/[1 - \eta g(t)^2/(4v)]$, which exceeds the target variance v. Reducing $g(t)^2$ or η reduces this discretization bias but also slows exploration, potentially requiring more corrections. Increasing L alone does not remove the bias.

In this illustration, using $g_n(t)^2 \leq \beta(t)$ reduces the update scale relative to v and, within the stable range, the stationary-variance inflation at a given η. Because the schedule depends on observation count rather than local curvature, the step size and correction count still need to be chosen appropriately.

Choosing correction step sizes and counts. Touron et al. (2026, §3) give rules for choosing the step size and number of Langevin steps together to meet a desired Wasserstein error tolerance in annealed Langevin sampling. Their guarantees require smooth, strongly log-concave intermediate densities and controlled score error. In their bound, more steps reduce mixing error but do not remove the residual error from the finite step size and score approximation. The required number of steps also depends on how much the target changes between noise levels, so the correction count need not be the same throughout sampling.

Choosing diffusion noise levels. Choosing the noise levels themselves is a separate optimization problem. For example, Williams et al. (2024) construct score-optimal schedules by estimating the cost of moving between successive marginals and redistributing the time points to approximately equalize that cost. This requires an additional cost-estimation and grid-adaptation procedure, distinct from choosing the number of ULA corrections.

A.5 Experimental evaluation

Gaussian benchmark. We used the correlated Gaussian model of Linhart et al. (2026), with parameter dimension $d = 10$ and observation-noise correlation $\rho = 0.8$. The prior and conditionally independent observations are

$$\theta \sim \mathcal{N}(0, I_d), \qquad x_j \mid \theta \overset{\text{iid}}{\sim} \mathcal{N}(\theta, \Sigma), \qquad \Sigma = (1-\rho)I_d + \rho \mathbf{1}\mathbf{1}^\top, \tag{34}$$

for $j = 1, \ldots, n$, where $\mathbf{1} \in \mathbb{R}^d$ is the all-ones vector. The exact posterior is $p(\theta \mid x_{1:n}) = \mathcal{N}(\theta; m_0, C_0)$, with mean m_0 and covariance C_0 before diffusion noise is added, given by

$$C_0 = (I_d + n\Sigma^{-1})^{-1}, \qquad m_0 = C_0 \Sigma^{-1} \sum_{j=1}^{n} x_j.$$

Controlled Gaussian setup. We compared the derived diffusion variance rate $g_n(t)^2$ with the ordinary VP rate $g_{\mathrm{VP}}(t)^2 = \beta(t)$ on the same prescribed Gaussian path. We examined numerical stability, covariance tracking, and posterior-sampling accuracy as n increases. Covariance tracking compares the sample covariance with its analytic target at each noise level. It checks whether the samples have the intended spread and correlations, which numerical stability alone does not guarantee. Exact scores let us isolate these effects without score-estimation error, and each paired comparison held the terminal particles, random noise draws, time grid, and correction count fixed. For Euler–Maruyama sampling, we used $\beta(t) = 0.05 + 19.95t$, $\alpha(t) = \exp(-0.05t - 9.975t^2)$, and the forward process from Eq. (26). At diffusion time t, this gives a Gaussian marginal with mean m_t, covariance C_t, and exact score s_t^{G}, the gradient of its log density,

$$m_t = \sqrt{\alpha(t)}\, m_0, \qquad C_t = \alpha(t) C_0 + \frac{1-\alpha(t)}{\kappa^{(n)}(t)} I_d, \qquad s_t^{\mathrm{G}}(\theta) = -C_t^{-1}(\theta - m_t).$$

Both coefficient choices, $c(t) = g_n^2(t)$ and $c(t) = \beta(t)$, retained this score, the linear reverse drift $\beta(t)\theta/2$, and the terminal law $\mathcal{N}(m_1, C_1)$. Each run used 1,000 particles and 400 equal reverse-time steps from 1 to 0, with step size $h = 1/400$ and no final Tweedie correction. We replaced the diffusion coefficient in both the predictor and ULA correctors while keeping the score and initialization unchanged. This isolates the effect of using $g_n(t)^2$ rather than $\beta(t)$ for the prescribed compositional noise path.

Stiffness and step-size sensitivity. Here the predictor advances between noise levels using an Euler–Maruyama step of the full reverse SDE in Eq. (25), before any additional ULA corrections. To assess stability, imagine two nearby particles receiving the same Gaussian noise. Their noise increments cancel in the difference, isolating how the drift changes their separation. This depends on the spatial derivative of the drift, its Jacobian, rather than on the drift's magnitude alone. Let $h > 0$ denote the sampling step size, so each predictor advances from diffusion time t to $t - h$. Combining the two score terms in Eq. (25) gives the sampling drift b_t and its Jacobian in this reverse-time direction,

$$\begin{aligned} b_t(\theta) &= \frac{\beta(t)}{2}\theta + c(t) s_t^{\mathrm{G}}(\theta) = \frac{\beta(t)}{2}\theta - c(t) C_t^{-1}(\theta - m_t), \\ J_t &\coloneqq \frac{\partial b_t}{\partial \theta} = \frac{\beta(t)}{2} I_d - c(t) C_t^{-1}. \end{aligned} \tag{35}$$

At a given noise level, m_t and C_t are fixed with respect to θ, so differentiating the Gaussian score contributes $-C_t^{-1}$. Writing δ for the separation between the two particles, one predictor step gives $\delta_{\mathrm{next}} = (I_d + hJ_t)\delta$. To check whether discretization turns contraction into expansion, we examine this update one eigendirection at a time. Here J_t is symmetric, so its eigenvectors form an orthogonal basis and any separation can be decomposed into components along these directions. Along each eigenvector, applying the Jacobian is exactly multiplication by its eigenvalue. This reduces the matrix update to a scalar calculation for each component, without changing the dynamics. For a negative eigenvalue $-r$, its magnitude $r > 0$ is the local contraction rate, so the continuous drift brings nearby particles closer along that direction. Taking δ along this eigenvector gives $J_t\delta = -r\delta$. Euler adds h times this rate of change to the current separation, giving

$$\delta_{\mathrm{next}} = \delta + hJ_t\delta = \delta - hr\delta = (1 - hr)\delta.$$

When $hr > 2$, the multiplier $1 - hr$ is less than -1. Its negative sign means that the particles exchange their ordering along this direction, while its absolute value exceeding one means that their distance increases. For example, $hr = 3$ gives $\delta_{\mathrm{next}} = -2\delta$, so the particles overshoot each other and end twice as far apart. This is numerical amplification in a direction where the continuous drift contracts. Checking all negative eigenvalues identifies the directions most sensitive to this overshoot.

We can summarize this sensitivity by asking how close each update comes to its local contraction limit. Along a predictor direction with contraction rate r, this limit is $2/r$, so the ratio $h/(2/r) = hr/2$ reaches one at the boundary between contraction and amplification. The same reasoning extends to a ULA corrector of duration h, whose contraction limit along a direction with variance v is $4v/c(t)$, as in Eq. (33). The corresponding ratio $hc(t)/(4v)$ is largest in the narrowest direction of the Gaussian, where the covariance eigenvalue is smallest. To capture the most restrictive update encountered during sampling, we define $R(n)$ as the largest of these ratios across both update types, their contracting

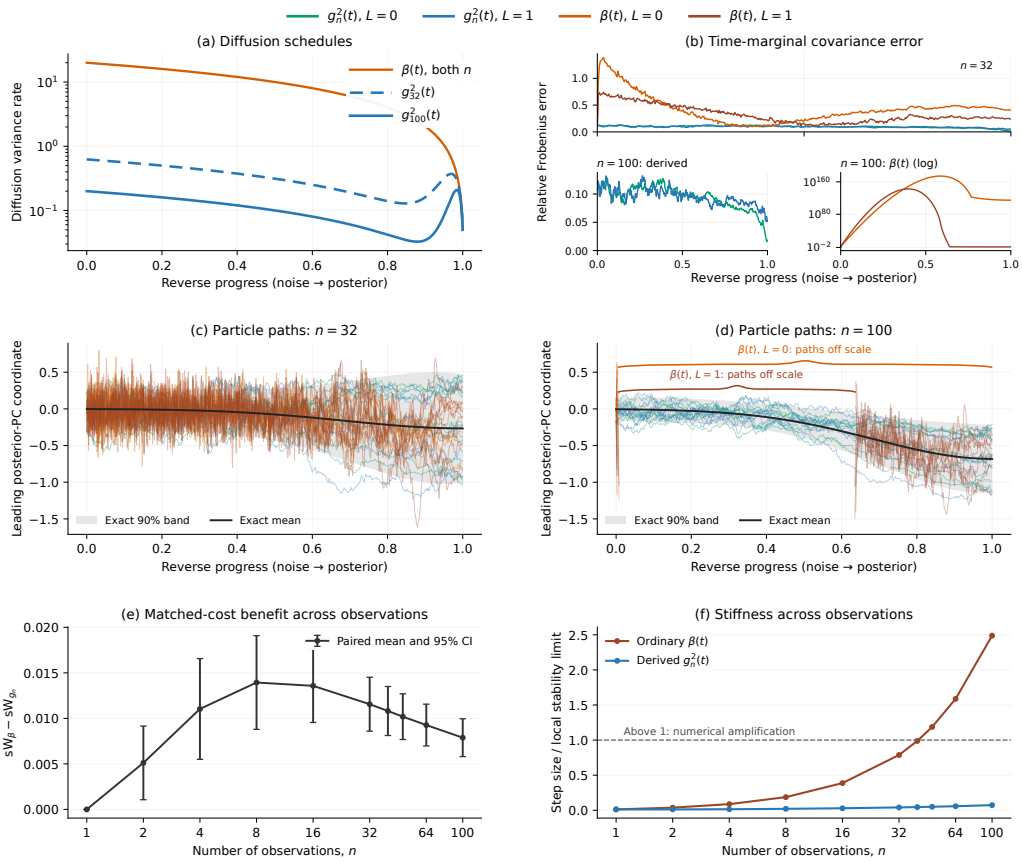

Figure 7: Compositional diffusion improves finite-budget Gaussian sampling. (*a–d*) Diffusion schedules, covariance tracking, and matched particle paths. At $n = 100$, ordinary-VP covariance errors require a logarithmic scale; path gaps and braces mark off-scale excursions. (*e*) Endpoint sW reduction at matched cost, with positive values favoring $g_n^2(t)$; bars are pointwise 95% intervals over five paired seeds. (*f*) Step size relative to the local contraction limit. Ordinary VP crosses 1, indicating numerical amplification, while the derived coefficient stays below it. Accuracy improves even below this threshold.

directions, and all sampling steps for n observations. With one correction per predictor step ($L = 1$) on the grid $t_k = kh$, the predictor uses time t_k and the subsequent corrector uses t_{k-1}, giving

$$R(n) = \max_{1 \le k \le 400} \left\{ \underbrace{\frac{h}{2} \max_i [-\lambda_i(J_{t_k})]_+}_{\text{predictor}}, \underbrace{\frac{h\, c(t_{k-1})}{4\lambda_{\min}(C_{t_{k-1}})}}_{\text{corrector}} \right\}, \tag{36}$$

where $[a]_+ = \max(a, 0)$ and λ_i denotes an eigenvalue. We exclude positive predictor eigenvalues because they describe expansion already present in the continuous drift, rather than amplification introduced by discretization. Panel (f) of Figure 7 reports $R(n)$ for each coefficient choice, showing how the same sampling step size fares as more observations are composed. Values below one mean that every tested contracting direction remains within its local step-size limit, whereas values above one identify at least one update that amplifies separation where the continuous drift would contract. A smaller $R(n)$ therefore leaves more room below these limits, although it does not by itself imply faster mixing or better posterior accuracy. It remains a local diagnostic rather than a guarantee of global stability or a measure of accumulated sampling error.

Assessing sampling accuracy. Avoiding numerical amplification does not ensure that the particles accurately represent the target distribution. For a Gaussian target, the mean and covariance together

determine the distribution, so samples can be centered correctly yet misrepresent uncertainty by being too concentrated, too dispersed, or incorrectly correlated. We therefore tracked whether the particles reproduce the intended spread and correlations throughout sampling by comparing their empirical covariance with the analytic target C_t, using the relative covariance error $\|\widehat{\mathrm{Cov}}(\theta_t) - C_t\|_F / \|C_t\|_F$, where $\|\cdot\|_F$ is the Frobenius norm. The plotted paths show eight matched particles projected onto the leading posterior principal component, against the exact mean and pointwise 90% band. For panels (e,f), we fixed $L = 1$ and swept $n \in \{1, 2, 4, 8, 16, 32, 40, 48, 64, 100\}$. Each method used 800 full compositional-score calls per run, each processing all particles. We measured empirical sliced 2-Wasserstein distance (sW) against 1,000 exact-posterior draws using 512 random unit directions, taking the square root of the mean squared difference between sorted projections. At each n, five paired seeds shared observations, terminal particles, predictor/corrector noise, reference draws, and projection directions across methods. Panel (e) reports the mean paired difference $\mathrm{sW}_\beta - \mathrm{sW}_{g_n}$, with pointwise 95% Student-t intervals using four degrees of freedom.

Effect of the diffusion coefficient. The two coefficients give identical samples at $n = 1$, while the derived coefficient lowers sW in all 45 paired comparisons with $n > 1$. At $n = 100$, mean sW falls from 0.0153 to 0.0074, a reduction of approximately 52%. The absolute improvement peaks near $n = 8$, rather than increasing monotonically. Ordinary VP crosses $R = 1$ between the tested values $n = 40$ and $n = 48$; at $n = 100$, R is 2.49 for ordinary VP versus 0.07 for the derived coefficient. Finite ordinary-VP endpoints with $L = 1$ can conceal transient relative covariance errors of order 10^{142}; these runs remain in the accuracy comparison. The coefficient therefore improves finite-budget accuracy and reduces step-size sensitivity, but the accuracy gains already appear below the local instability threshold. Changing the coefficient while retaining the score and linear drift also changes compatibility with the prescribed continuous-time marginals, so numerical stiffness alone does not explain the difference. These controlled Gaussian results do not establish the same behavior for learned-score SLCP or BMP sampling.

A.6 Exact-score Gaussian benchmark

We evaluated the complete observation-count sweep $n \in \{1, 2, 4, 8, 16, 32, 64, 100\}$ using the exact compositional score and posterior of the 10-dimensional correlated Gaussian model defined in § A.5. For every n, each method used five common data seeds, 1,000 posterior particles, and $T = 400$ reverse steps. The F-NPSE-SDE comparisons used $L \in \{1, 5\}$ Langevin corrections per predictor. This paired design separates changes caused by the compositional sampler from variation in observations or reference samples.

A.7 SLCP benchmark

We evaluated $n \in \{1, 14, 30\}$ across three independently trained single-observation score networks and 25 common SLCP test instances per network seed. Each network was trained on simulator-generated SLCP pairs. Within each seed and test instance, all compositional samplers used the same frozen score network and reference posterior, so the comparison isolates the sampling method. Following Linhart et al. (2026), JAC and Langevin bounded standardized iterates coordinatewise to $[-3, 3]$ for numerical stability. We report sliced Wasserstein distance, MMD, and C2ST using the same definitions as in Table 1.

A.8 Computational cost of compositional samplers

Consider sampling P posterior particles, or P simultaneous reverse-diffusion chains, over d unknown parameters conditioned on n observations using T reverse steps. Let C_s denote the architecture-dependent cost of one unbatched score-network evaluation for one particle-observation pair at one diffusion time. F-NPSE-SDE evaluates n scores for every particle in each predictor and repeats these evaluations in each of L ULA corrector steps. Its compute cost is therefore $\mathcal{O}(T(L+1)PnC_s)$. Because scores can be accumulated over observations and ULA steps run sequentially, its working state requires $\mathcal{O}(Pd)$ memory beyond the score network's minibatch activations. Thus, increasing L increases runtime but not the asymptotic peak state memory.

JAC forms a full $d \times d$ score Jacobian for every particle-observation pair at every reverse step. Computing these Jacobians requires roughly $\mathcal{O}(TPndC_s)$ work, corresponding to approximately

d reverse-mode passes per score, in addition to dense covariance inversions and solves. A fully vectorized implementation stores $\mathcal{O}(Pnd^2)$ Jacobian values, an nd-fold increase over the $\mathcal{O}(Pd)$ particle state of F-NPSE-SDE. For pooled BMP inference, $n = 940$ and $d = 65$, so this auxiliary state is approximately 6.1×10^4 times larger. Streaming over observations can instead reduce peak Jacobian storage to $\mathcal{O}(Pd^2)$, but serializes the same computation and retains the additional autodifferentiation and dense linear-algebra costs.

GAUSS avoids online Jacobians but estimates a covariance for each observation from M auxiliary samples generated by S-step DDIM rollouts. For fixed conditioning, its preprocessing cost is $\mathcal{O}(nMSC_s)$. A straightforward batched implementation uses $\mathcal{O}(nMd + nd^2)$ storage for auxiliary samples and per-observation covariances. Online covariance accumulation can remove the nMd term, and processing contexts sequentially can further reduce peak storage, but each context still requires the rollout computation and estimation of a $d \times d$ covariance. Sampling then requires $\mathcal{O}(TPnC_s)$ score evaluations and dense precision operations.

These costs are amplified by the conditional block updates in HBDS. Each reverse-time level updates four five-dimensional receptor blocks conditional on the current 60-dimensional global state, then updates the global state conditional on the refined receptors. JAC consequently requires four sets of local Jacobians and a global 60×60 Jacobian family. If fully vectorized, the global family alone is approximately $n(60)^2/(60+4\times5) \approx 4.2\times10^4$ times larger than the complete F-NPSE-SDE HBDS particle state. Streaming reduces peak memory but repeats this Jacobian computation serially within every block update. For GAUSS, the relevant conditional covariances change with the current global and receptor states, so its DDIM rollouts become nested inside the block refinements rather than being amortized once. By contrast, F-NPSE-SDE stores only the HBDS particle state, $\mathcal{O}(P(60 + 4 \times 5))$, beyond common score-network activations. HBDS therefore adds sequential score evaluations for F-NPSE-SDE without requiring these Jacobian or covariance calculations.

BMP runtime and memory. In our BMP experiments, JAC increased HBDS sampling from approximately one hour to more than ten hours, while GAUSS exhausted GPU memory during context-specific covariance estimation across 940 designs. We therefore omit both from the subsequent BMP evaluation.

A.9 Compositional sampling related works

Factorized posterior sampling. Our construction builds on F-NPSE (Geffner et al., 2023), which combines single-observation posterior scores with a prior correction for inference from multiple observations. We retain this factorization and use its discrete transition to motivate a continuous-time Gaussian surrogate. The contribution is the resulting variance prescription and compatible diffusion coefficient, rather than score composition itself; Arruda et al. (2025) review this broader family of SBI methods.

Inference-time schedule adjustments. Our scalar approximation is closely related to the sampling-time modifications of Arruda et al. (2026), who stabilize composition through score damping and an adjustable constant shift of the log signal-to-noise ratio (SNR), alongside minibatch score estimation to reduce memory requirements. For our Gaussian surrogate, define $\mathrm{SNR}_n(t) = \alpha(t)/V_n(t)$ and $\mathrm{SNR}_{\mathrm{VP}}(t) = \alpha(t)/(1-\alpha(t))$ at times with $0 < \alpha(t) < 1$. The prescribed variance in Eq. (26) gives

$$\log \mathrm{SNR}_n(t) = \log \mathrm{SNR}_{\mathrm{VP}}(t) + \log \kappa^{(n)}(t). \tag{37}$$

Thus, $\kappa^{(n)}(t)$ induces a time- and observation-count-dependent log-SNR shift. We obtain this adjustment by prescribing a composition-dependent cumulative variance and deriving its compatible diffusion coefficient while retaining the ordinary VP mean attenuation. Since the mean attenuation is unchanged while the variance is reduced, this is not simply a reparameterization of the ordinary VP schedule, nor is it equivalent to multiplying the composed score by a damping factor. The derivation realizes the chosen Gaussian surrogate but does not establish an exact reverse process for the full compositional posterior.

Covariance-aware sampling. The samplers of Linhart et al. (2026) address errors in composing diffusion-time scores through score Jacobians (JAC) or auxiliary posterior covariance estimates (GAUSS). We compare these methods directly on Gaussian and SLCP benchmarks in § 4.1; their

Algorithm 1: Hierarchical Blockwise Diffusion Sampling (HBDS)

Input: Groups $\{(\xi^k, y^k)\}_{k=1}^K$; score model s_ϕ; sampling schedule (f, g_n); time grid $t_T > \cdots > t_0 = 0$; block rounds $M \geq 1$; corrector steps $L \geq 0$.

$\mathrm{PC}_{t\to s}$ combines one predictor step $t \to s$ with L Langevin corrections at s.

Init: $\theta_{t_T} \sim q_{\text{init}}^{\theta}$; $R_{t_T}^k \sim q_{\text{init}}^{R^k}$ for $k = 1, \ldots, K$.

for $i = T, T-1, \ldots, 1$ **do**
  $(t, s) \leftarrow (t_i, t_{i-1})$
  for $m = 1, \ldots, M$ **do**
    for $k = 1$ **to** K **do**
      $R_s^k \leftarrow \mathrm{PC}_{t\to s}^{R}\big(R_t^k \mid \theta_t, \xi^k, y^k\big)$
    $\mathcal{G}_s \leftarrow \{(R_s^k, \xi^k, y^k)\}_{k=1}^K$
    $\theta_s \leftarrow \mathrm{PC}_{t\to s}^{\theta}(\theta_t \mid \mathcal{G}_s)$
    if $m < M$ **then**
      Re-noise $(\theta_s, R_s^{1:K})$ to obtain $(\theta_t, R_t^{1:K})$ at noise level t

Output: $(\theta_0, \{R_0^k\}_{k=1}^K)$ conditioned on $\{(\xi^k, y^k)\}_{k=1}^K$.

additional computations become more expensive when repeated inside HBDS block updates. In our BMP implementation, JAC increased HBDS sampling from approximately one hour to more than ten hours, while GAUSS exhausted GPU memory during covariance estimation across 940 designs (§ A.8). These observations concern the tested implementation, not an impossibility of adapting either method. Our score-only compositional sampler makes the repeated block updates practical in this setting. The hierarchical inference construction is discussed separately in § B.2.

B Hierarchical blockwise diffusion sampling details

Grouped posterior target. Suppose there are K experimental groups, each with design ξ^k, observation y^k, and local latent state R^k, together with global parameters θ. Assuming condition-specific independent receptor priors, as in our BMP model, the grouped posterior factorizes as

$$p(\theta, R^{1:K} \mid y^{1:K}, \xi^{1:K}) \propto p(\theta) \prod_{k=1}^{K} p(R^k \mid \xi^k)\, p(y^k \mid R^k, \theta, \xi^k).$$

A pooled model restricts this family by tying $R^1 = \cdots = R^K$. If two groups require different local states on a set with positive posterior mass, that restriction cannot represent the grouped posterior; § B.1 gives an analytic example.

Blockwise intermediate target. HBDS uses conditional scores to approximate the unavailable noised joint posterior $p_t(\theta, R^{1:K} \mid y^{1:K}, \xi^{1:K})$. Over each diffusion interval, it advances and refines the local receptor blocks before updating the shared parameters using all groups. Additional block rounds repeat this interval after re-noising the updated states, rather than adding another reverse transition after the predictor–corrector updates.

Algorithm 1 uses $\mathrm{PC}_{t\to s}^{B}$ to denote a predictor–corrector update that advances block B from t to s and applies L Langevin corrections at s. In the continuous implementation, the predictor uses the probability-flow part of Eq. (25) and is deterministic. Each correction follows Eq. (32) with noise covariance $\eta g_{n_B}(s)^2 I$, where $\eta = (t-s)/L$ for $L > 0$. Here n_B is the observation count for that block, namely the size of group k for R^k and the total count for θ. Setting $L = 0$ skips corrections, which are also omitted on the final diffusion interval. Thus, no additional Gaussian reverse step is needed after these updates. The initialization laws q_{init} use Gaussian noise for θ and, in the BMP experiments, the receptor prior in § D.

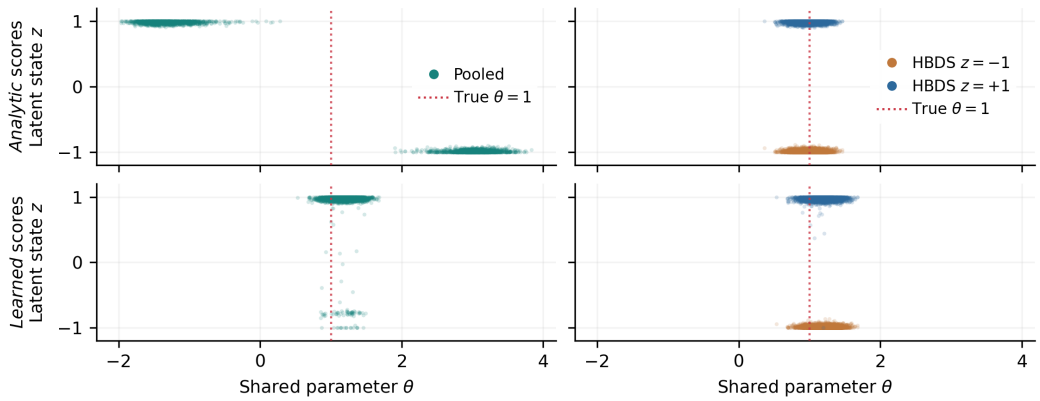

Figure 8: Joint-state diagnostic for the hierarchical toy with 20 observations across two groups and no ULA correctors. (*Left*) Pooled sampling with one shared latent. (*Right*) HBDS with separate group latents. Rows use analytic (top) and learned (bottom) scores; dotted lines mark the generating $\theta = 1$. This oracle diagnostic conditions HBDS latent updates on the true θ and parameter updates on the true group signs; pooled sampling receives no such information.

B.1 Toy simulator model example

We consider a hierarchical model with a discrete latent $k \in \{-1, +1\}$, a continuous latent $\theta \in \mathbb{R}$, and an observation $x \in \mathbb{R}$. The generative process is

$$k \sim \mathrm{Rad}\left(\tfrac{1}{2}\right), \qquad \text{(Rademacher: } \mathbb{P}[k=+1] = \mathbb{P}[k=-1] = \tfrac{1}{2}\text{)}, \tag{38}$$

$$\theta \sim \mathcal{N}(0,1), \tag{39}$$

$$x|\theta, k \sim \mathcal{N}\big(5k+\theta,\, 1\big), \quad \text{i.e., } x = 5k + \theta + \varepsilon,\ \varepsilon \sim \mathcal{N}(0,1) \text{ indep.} \tag{40}$$

The joint density factorizes as

$$p(\theta, k, x) = p(k)p(\theta)p(x|\theta,k) = \tfrac{1}{2}\,\mathcal{N}(\theta; 0, 1)\,\mathcal{N}\big(x; 5k+\theta, 1\big). \tag{41}$$

Thus, the likelihood is $p(x|\theta,k) = \mathcal{N}(x; 5k+\theta, 1)$, with prior $p(\theta) = \mathcal{N}(0,1)$ and $p(k) = \frac{1}{2}$. The corresponding posterior distributions are analytically tractable. The discrete latent variable has

$$p(k=+1|x) = \frac{\mathcal{N}(x;\,5,\,2)}{\mathcal{N}(x;\,5,\,2) + \mathcal{N}(x;\,-5,\,2)}, \qquad p(k=-1|x) = 1 - p(k=+1|x), \tag{42}$$

while the continuous latent variable conditioned on x and k follows

$$p(\theta|x,k) = \mathcal{N}\big(\theta;\, \tfrac{1}{2}(x-5k),\, \tfrac{1}{2}\big). \tag{43}$$

Together these define the exact posterior for a single observation,

$$p(\theta, k|x) = p(\theta|x,k)\, p(k|x), \tag{44}$$

which serves as the analytic ground truth for single-sample posterior evaluation.

Pooled versus grouped posterior structure. For K experimental groups, we introduce one group-specific discrete latent $z_g \in \{-1, +1\}$ per group while sharing the continuous parameter θ:

$$z_g \sim \mathrm{Rad}\left(\tfrac{1}{2}\right), \qquad \theta \sim \mathcal{N}(0,1), \qquad x_g \mid \theta, z_g \sim \mathcal{N}(5z_g + \theta, 1).$$

The hierarchical posterior is

$$p(\theta, z_{1:K} \mid x_{1:K}) \propto p(\theta) \prod_{g=1}^{K} p(z_g) p(x_g \mid \theta, z_g),$$

which allows each group to select its own latent mode. A pooled model instead ties all groups to a single latent z,

$$p_{\text{pool}}(\theta, z \mid x_{1:K}) \propto p(\theta)p(z)\prod_{g=1}^{K} p(x_g \mid \theta, z).$$

Thus, whenever different groups are better explained by different latent modes, the hierarchical posterior can represent the oracle structure while the pooled posterior cannot. For example, observations near $+5$ and -5 are naturally explained by different group-specific latents with $\theta \approx 0$, whereas the pooled model must force both groups to share the same latent explanation.

Learned-score illustration. The main-text comparison in Figure 4 used 20 observations, ten per group, generated with $\theta = 1$, signs $z_1 = -1$ and $z_2 = +1$, and unit observation-noise standard deviation. Both samplers reused a frozen single-observation score network, with 1,000 particles per sampling seed across seeds 0, 1, and 2, 200 predictor intervals, and no ULA correctors. HBDS conditioned its block updates on cumulative-VP Tweedie estimates, without access to the generating parameters or signs. The plot shows all saved paired particles without jitter or latent rounding; continuous latent states are sampler approximations to the discrete target.

B.2 Hierarchical SBI related works

Hierarchical posterior estimators. Hierarchical SBI explicitly distinguishes shared parameters from local latent variables. Heinrich et al. (2024) learn dataset-wide representations of event ensembles and separate global and local posterior estimators, supporting varying ensemble sizes. Habermann et al. (2025) introduce multilevel neural posterior estimation (ML-NPE), pairing hierarchical summary networks with global and local posterior estimators. Their framework supports covariates, shared parameters, and varying numbers of groups and observations per group within the training range, with validation through simulation-based calibration and comparisons to Stan. Charles et al. (2026) combine group-aware tokenization with flow matching, using a per-site likelihood surrogate to generate synthetic multi-site datasets for training a separate hierarchical posterior estimator. These approaches build the hierarchy into the learned inference model.

Hierarchical score models. Arruda et al. (2026) train global and local score estimators on individual groups, compose the global scores across groups, and then sample local parameters conditional on the sampled globals. This supports hierarchical inference without simulating the full hierarchy for each training example. HBDS instead reuses the conditional queries of one joint score model and alternates local and global updates within the reverse-diffusion process, rather than training separate estimators for the hierarchical posterior factorization. The distinction is conditional-model reuse and blockwise sampling, not hierarchical composition itself. Their compositional sampling modifications are discussed in § A.9.

Nested, design-dependent observations. In BMP, each of four cell lines has one receptor vector shared across 235 distinct ligand-dose observations, while biophysical parameters are shared across all cell lines (§ C). Assigning an independent local latent to every scalar observation would discard this within-cell-line sharing. Conversely, treating a cell line as one group requires representing its panel of design–response pairs within the group-level inference model. ML-NPE already accommodates nested observations and changing dataset sizes (Habermann et al., 2025); HBDS instead performs the aggregation through score composition at sampling time. It retains the single-observation representation and composes scores within receptor blocks and across groups for the shared parameters. Observation subsets can therefore change without training another hierarchical estimator, provided the underlying model has learned the required conditionals. The same sampler can also reuse the joint model adapted through likelihood fine-tuning (§ 3.2).

C BMP signaling pathway mathematical model

The BMP dataset contains $n = 940$ steady-state cell response measurements from $C = 4$ cell lines, each profiled under 235 distinct ligand-dose conditions. We write $y = (y_1, \ldots, y_n)$ for these observations. The parent and three receptor-knockdown lines form four groups, each with a

distinct receptor state shared across its ligand-dose experiments. These groups share the simulator's biochemical affinities and catalytic efficiencies, giving the hierarchical structure used by HBDS in § 3.5.

BMPs are homodimeric and heterodimeric ligands that transmit information by assembling signaling complexes with two classes of serine/threonine-kinase receptors: type I (A_i) and type II (B_k). In a parsimonious "one-step" mass–action kinetics model (Su et al., 2022), each ligand L_j binds one receptor of each class in a single equilibrium step

$$A_i + B_k + L_j \overset{K_{ijk}}{\rightleftharpoons} T_{ijk}, \quad \varepsilon_{ijk}\, T_{ijk} \longrightarrow S, \tag{45}$$

where K_{ijk} is an effective affinity for forming the active trimer T_{ijk} and ε_{ijk} is its catalytic efficiency in phosphorylating SMAD1/5/8, producing a transcriptional signal S. Interestingly, despite treating the 10 known ligands and $4 \times 3 = 12$ canonical receptors as effectively promiscuous, the model recovers dose–response data and explains how receptor context rewires downstream transcriptional programs during development or oncogenesis (Antebi et al., 2017; Su et al., 2022). Here, we model a subset of the known ligands and canonical receptors where $I = 3$, $J = 5$, and $K = 2$ denote the numbers of type-I receptors, ligands, and type-II receptors, respectively, so that $i = 1, \ldots, I$, $j = 1, \ldots, J$, and $k = 1, \ldots, K$. Table 2 summarizes how the SBI variables map onto this BMP model. The receptor concentrations A and B can also be treated as fixed design variables, as in perturbation-modeling settings where biological covariates are specified interventions (Dimitrov et al., 2026), giving $(L_j, A_i, B_k) \in \Xi^{J+I+K}$ and $(\varepsilon_{ijk}, K_{ijk}) \in \Theta_{>0}^{2IJK}$, but since receptor measurements are noisy, we treat them as latent variables throughout the experiments.

Table 2: BMP-to-SBI mapping and dimensions.

SBI symbol	BMP meaning	Dimension
ξ	Experimental design: ligand doses $L_{1:J}$	$J = 5$
θ	Latents: receptors $A_{1:I}, B_{1:K}$, affinities K_{ijk}, efficiencies ε_{ijk}	$I + K + 2IJK = 65$
y	Observation: steady-state SMAD response S	1

D CELL RECEPTOR HIERARCHICAL MODELING

We implement diffusion guidance by constructing a probabilistic prior over receptor expression informed by quantitative polymerase chain reaction (qPCR) measurements and then using the resulting (marginal) log-density as an energy function for guidance during sampling. The goal is to bias samples toward receptor configurations consistent with known KD conditions while explicitly accounting for the substantial uncertainty inherent to qPCR.

Indices and observations. Let $s \in \{0, 1, 2, 3\}$ index the four experimental cell-line conditions (one wild-type and three KD conditions). Let $j \in \{0, 1, 2, 3, 4\}$ index the five BMP-pathway receptors considered in this work (ACVR1, BMPR1A, ACVR2A, ACVR2B, BMPR2). For each condition s and receptor j we observe a positive relative qPCR readout $x_{s,j} > 0$, interpreted as a noisy proxy for receptor abundance (relative mRNA expression). Because qPCR variability is multiplicative, we model observations in log space, $y_{s,j} \triangleq \log x_{s,j}$.

Knockdown targets. For each condition s, the KD target index is $k_s \in \{-1, 0, 1, 2, 3, 4\}$, where $k_s = -1$ denotes no knockdown (wild-type) and $k_s = j$ denotes that receptor j is knocked down in condition s. We define an indicator

$$\mathbb{I}_{s,j} \triangleq \mathbb{I}[\, k_s = j \,],$$

which is 1 iff receptor j is the KD target in condition s and 0 otherwise.

Hierarchical model (baseline receptor distribution). We posit receptor-specific baseline log-means and log-standard-deviations that are partially pooled across receptors via global hyperparameters. Let μ_j denote the baseline log-mean for receptor j, and let σ_j denote the baseline

log-standard-deviation for receptor j. We introduce global hyperparameters (μ_0, κ) controlling the population distribution of receptor means, and (ℓ_0, η) controlling the population distribution of receptor log-variances. In standard hierarchical form,

$$\begin{aligned} \mu_0 &\sim \mathcal{N}(m_0, v_0), \\ \kappa &\sim \mathrm{HalfNormal}(\lambda_\kappa), \\ \mu_j \mid \mu_0, \kappa &\sim \mathcal{N}(\mu_0, \kappa^2), \qquad j = 0, \dots, 4, \\ \ell_0 &\sim \mathcal{N}(m_\ell, v_\ell), \\ \eta &\sim \mathrm{HalfNormal}(\lambda_\eta), \\ \log \sigma_j \mid \ell_0, \eta &\sim \mathcal{N}(\ell_0, \eta^2), \qquad j = 0, \dots, 4. \end{aligned}$$

This hierarchy induces partial pooling: when data are weak or noisy, $\{\mu_j, \sigma_j\}$ are shrunk toward the global centers (μ_0, ℓ_0), while still allowing receptor-specific deviations.

Condition-level variability. To account for inter-condition variability (e.g., biological differences between cell lines and experimental perturbations not explained by the KD), we add a condition-specific random effect $\delta_{s,j}$ for each receptor:

$$\begin{aligned} \tau &\sim \mathrm{HalfNormal}(\lambda_\tau), \\ \delta_{s,j} \mid \tau &\sim \mathcal{N}(0, \tau^2), \qquad s = 0, \dots, 3, \;\; j = 0, \dots, 4. \end{aligned}$$

This term captures the fact that receptor expression can vary appreciably across conditions even in the absence of knockdown.

Knockdown effect model (multiplicative fold-change). Knockdown acts multiplicatively in the original receptor domain, which corresponds to an additive shift in log space. We therefore model a KD fold-change via a log-shift $\nu_j \le 0$ and additional KD uncertainty $\omega_j \ge 0$:

$$\begin{aligned} \nu_j &\sim \mathcal{N}(m_{\nu,j}, v_{\nu,j}) \;\text{ with support on } (-\infty, 0], \\ \omega_j &\sim \mathrm{HalfNormal}(\lambda_{\omega,j}), \end{aligned}$$

and incorporate KD by modifying the conditional distribution of $y_{s,j}$. Specifically, the KD target receptor ($\mathbb{I}_{s,j} = 1$) receives a negative mean shift and an additional variance term:

$$y_{s,j} \mid \mu_j, \sigma_j, \delta_{s,j}, \nu_j, \omega_j \;\sim\; \mathcal{N}\Big(\mu_j + \delta_{s,j} + \mathbb{I}_{s,j}\,\nu_j, \;\; \sigma_j^2 + \mathbb{I}_{s,j}\,\omega_j^2\Big).$$

Non-target receptors ($\mathbb{I}_{s,j} = 0$) follow the baseline distribution.

Observation model in the original receptor domain. Let $r_{s,j} \triangleq \exp(y_{s,j})$ denote the receptor expression in the original (positive) domain. In our implementation we further impose a physically plausible upper bound $h > 0$ and use a truncated LogNormal likelihood with support $r_{s,j} \in (0, h)$:

$$r_{s,j} \mid \mu_j, \sigma_j, \delta_{s,j}, \nu_j, \omega_j \sim \mathrm{TruncLogNormal}\Big(\mu_j + \delta_{s,j} + \mathbb{I}_{s,j}\,\nu_j, \;\sqrt{\sigma_j^2 + \mathbb{I}_{s,j}\,\omega_j^2}; \; h\Big).$$

Equivalently, this is a LogNormal distribution in $r_{s,j}$ with truncation at h, which provides numerical stability and reflects the bounded receptor regime used in simulation.

Using the model for guidance. The hierarchical model defines a receptor prior density $p(r \mid s)$ for each condition s. During diffusion sampling, we compute a denoised receptor estimate and apply guidance by adding the gradient of the (marginal) log-prior,

$$g(r, s) \;\triangleq\; \nabla_r \log p(r \mid s),$$

scaled by a time-dependent factor (analogous to classifier-free guidance when used as a prior). In practice, marginalization over latent variables (e.g., $\mu_0, \kappa, \ell_0, \eta, \tau$) can be performed either by a tractable approximation (e.g., moment-matching) or by mixture density networks (Bishop, 1994); both yield a differentiable log-density suitable for guidance.

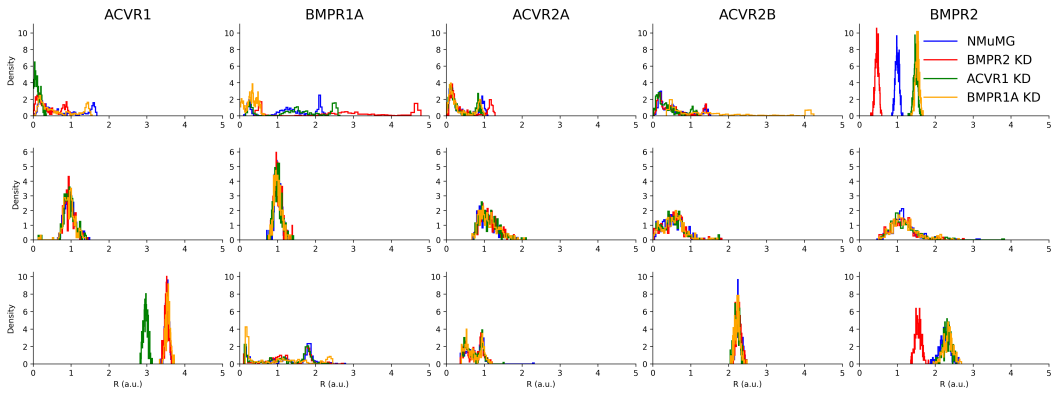

Figure 9: Comparison between BMP receptor posterior samples (columns) across LSR best-fit θ^{LSR} (*top*), applying a truncated lognormal prior to the BMP receptors' distributions (*middle*), and applying the marginal hierarchical receptor priors § D (*bottom*). Model identifiability in the θ^{LSR} parameter set can be seen in the multi-modal distributions of the θ^{LSR} fits as well as the case of BMPR2, where θ^{LSR} only works if the ACVR1 KD and BMPR1A KD conditions are $1.5\times$ the amount of the *parent* cell line. Imposing a diffusion prior on the receptors in the BMP simformer avoids this clash of parameters.

E Additional BMP experimental results

E.1 Calibrated receptor diffusion

The LSR receptor estimates in Figure 9 illustrate potential identifiability issues. Some knockdown (KD) conditions have higher fitted basal receptor levels than the parent cell line, contrary to the intended knockdown. Low predictive error alone therefore does not establish physical plausibility. The receptor priors in § D allow posterior sampling to incorporate these biological constraints.

LSR fit and receptor sensitivity. We compared FT posterior-predictive fit with the historical least-squares regression estimate θ^{LSR} and a sensitivity analysis in which its ACVR1 estimate was reduced by 10% (Table 3). The FT model in this comparison has three transformer layers and a feed-forward widening factor of two, with $\lambda = 5 \times 10^{-4}$. Its all-data row used HBDS on 940 observations, whereas each cell-line row used an independent pooled posterior conditioned on that cell's 235 observations. These rows are not subsets of a single all-data posterior. Each run used 250 draws, 100 diffusion steps, five Langevin corrections per step, and sampling seed 42. The all-data FT RMSE is 0.26. The historical LSR errors were retained without recomputation, and their all-data value is not the root-mean-square of the four cell-wise values, so they should not be treated as a consistently aggregated baseline until the original evaluation protocol is reconciled. The receptor perturbation illustrates local fit sensitivity.

E.2 BMP posterior-sampling ablations

We compared pooled sampling and HBDS on 940 BMP observations, with 235 observations from each of four cell lines. Pooled sampling infers the 60 shared biophysical parameters and one shared five-dimensional receptor vector, whereas HBDS maintains a separate receptor state for each cell line. Both samplers used the derived diffusion coefficient, 100 diffusion steps, five Langevin corrections per step, 250 posterior draws, and sampling seed 42. HBDS used three block-refinement rounds with Tweedie conditioning for both blocks. Pooled sampling used three resampling passes in one sequential round, which are not hierarchical block updates. These are single-seed results, with no between-seed uncertainty estimates.

Models and predictive metrics. The PT and FT models in Tables 4 and 5 have four transformer layers and a feed-forward widening factor of four. FT started from the matched PT model and used $\lambda = 5 \times 10^{-4}$ for the sampler comparison. The no-FiLM model is a separately pretrained and fine-tuned architectural ablation that retains mask-state embeddings, not the same model with FiLM

Table 3: RMSE comparison of historical LSR, ACVR1 -10% LSR ($\theta^{\mathrm{LSR}\dagger}$), and FT predictions. FT uses HBDS for all data and independent pooled inference for each cell line. Lower values are better.

Condition	θ^{LSR}	$\theta^{\mathrm{LSR}\dagger}$	FT
All data	0.08	1.02	0.26
NMuMG	0.08	0.08	0.48
BMPR1A KD	0.07	0.06	0.31
BMPR2 KD	0.09	0.06	0.13
ACVR1 KD	0.06	2.04	0.22

Table 4: Pooled/HBDS comparison for PT, FT, and separately trained FT without FiLM (¬FiLM), using 250 draws and one sampling seed.

			ℓ-C2ST	
Method	**RMSE ↓**	**Median ↓**	$\boldsymbol{\theta}$	$\boldsymbol{R}$
PT Pooled	0.58	1,594.55	0.22	0.06
FT Pooled	0.36	66.77	0.18	0.02
¬FiLM Pooled	12.30	4,538.82	0.13	0.03
PT HBDS	0.78	194.01	0.05	0.04
FT HBDS	**0.34**	**14.94**	0.02	0.07
¬FiLM HBDS	0.91	274.61	0.09	0.01

switched off at sampling time. It uses the same depth and width as the FiLM model. For posterior draw k, let $d_k = \|\mathbf{y}_k - \mathbf{y}_o\|_2$, where $\mathbf{y}_k$ is evaluated through the original mechanistic simulator over N observations. We report $\mathrm{RMSE} = \min_k d_k/\sqrt{N}$ and median distance $\mathrm{median}_k\, d_k$. Thus, RMSE measures the best predictive draw rather than the posterior mean, while median distance summarizes response-space error across draws.

Calibration. We report the logged local classifier two-sample test (ℓ-C2ST) (Linhart et al., 2023) statistics separately for parameters and receptors. The tests used an MLP with three folds and two ensemble members, classifier seed zero, and 100 permutation-based null trials. These are single-run statistics, not averages across independent sampling seeds.

Fine-tuning improves both predictive-fit metrics over its matched PT model under pooled sampling and HBDS (Table 4). For each FT architecture, HBDS gives lower errors than pooled sampling, and FT with FiLM has the lowest reported values among the completed comparisons. The three-layer results in Table 3 use a different model family and are not a controlled comparison of network size.

F Pretraining Details

Transformer architecture. The simformer uses value, ID, condition, and attention dimensions of 32, with three attention heads of size 32. The model used for the cell-line and LSR comparison in Table 3 has three layers and a feed-forward widening factor of two. The model/sampler and regularization ablations in Tables 4 and 5 instead used four layers and a widening factor of four. Rather than summing token embeddings (ID, condition, and value), we concatenate them prior to projection, which empirically produced more stable training and better downstream posterior quality.

We trained the simformer using 1,000 prior parameter draws evaluated across 940 experimental designs. For each draw $\theta \sim p(\theta)$ and design ξ, we generated expert-model outputs $y \sim p_{\text{eval}}(y \mid \theta, \xi)$ to form training examples (θ, ξ, y). While preliminary results suggest that increasing the number of prior parameter draws improves downstream posterior quality, we restricted this study to 1,000 draws for computational simplicity.

During training we employed two forms of masking. For *edge masking*, we applied masks corresponding to marginals, the full joint, and the expert model graph, each with equal probability ($\frac{1}{3}$ of the time). For *condition masking*, we used four different masks: (i) joint masking (noise all variables), (ii) posterior conditioning, (iii) likelihood conditioning, and (iv) random conditioning on 30% of tokens in arbitrary groupings. These were applied with probabilities of 20% for the joint, posterior, and likelihood masks, and 40% for the random conditioning. We found that heavier use of posterior and likelihood masks tended to produce overconfident models, whereas increasing the joint masking improved generalization.

Pretraining pipeline. We trained for 100 epochs with a batch size of 2,048. Evaluating these parameter draws across all 940 designs produced approximately $1{,}000 \times 940 = 940{,}000$ simulator outputs. While this represents a large training set, the dimensionality of the parameter space ($\theta \in \mathbb{R}^{60}$) implies a trade-off between the number of prior parameter draws available and the coverage of experimental conditions. In practice, this dataset size was sufficient for demonstrating the model's

capabilities within available compute, but there is room for future work studying scaling laws of the number of prior parameter draws and the number of parameters of the architecture.

Compute resources. All models were trained on NVIDIA A100 GPUs. Sampling and smaller ablation runs were also performed on NVIDIA A30 GPUs when memory constraints allowed.

G Fine-tuning and path-space details

MSE reward. We use negative MSE as the reward for agreement between conditional likelihood samples and observed responses. Concretely, for an observation y^{o} (under design ξ) and Monte Carlo samples $\{y_m\}_{m=1}^{M}$ drawn from the simformer diffusion sampler,

$$\{y_m\}_{m=1}^{M} \sim \pi^{\star}(\phi, \theta, R; \xi),$$

we define the per-condition reward

$$\mathcal{R}_{\mathrm{MSE}}(\phi, \theta, R; \xi) \;\coloneqq\; -\frac{1}{M} \sum_{m=1}^{M} \left\| y_m - y^{\mathrm{o}} \right\|_2^2.$$

Equivalently, the corresponding loss contribution is the Monte Carlo estimate of the expected MSE,

$$\widehat{\mathcal{L}}_{\mathrm{MSE}} \;=\; -\mathcal{R}_{\mathrm{MSE}}(\phi, \theta, R; \xi) \;=\; \frac{1}{M} \sum_{m=1}^{M} \left\| y_m - y^{\mathrm{o}} \right\|_2^2,$$

computed per experimental condition (per ξ) and aggregated across a minibatch during fine-tuning.

Time convention. Throughout the paper, diffusion time t increases from 0 to T during noising and decreases from T to 0 during sampling. To express the adjoint calculation in the increasing sampling clock used by Marion et al. (2025), set $s = T - t$ and $Z_s = Y_{T-s}$. Sampling then starts with noise at $Z_0 = Y_T$ and ends with the generated response $Z_T = Y_0$, consistently with § G.2.

Adjoint gradients. The diffusion sampler induces the mapping $\pi^{\star}(\phi, \theta, R; \xi)$ implicitly through the reverse-time SDE. In increasing sampling time, its drift is $\mu(s, z; \phi, \theta, R, \xi) = -b_\phi(z, T-s; \theta, R, \xi)$, where b_ϕ is the reverse-SDE drift. The sampling dynamics are

$$dZ_s = \mu(s, Z_s; \phi, \theta, R, \xi)\, ds + g(T-s)\, dW_s,$$

where W_s is a Wiener process in the sampling clock. The adjoint calculation retraces this sampled trajectory using $r = T - s$, starting at the generated response when $r = 0$. With column-vector gradients, the adjoint system of Marion et al. (2025, Appendix A.3, Eq. 14) becomes

$$\begin{aligned} A_0 &= \nabla_z \mathcal{R}(Z_T), & dA_r &= \left[\partial_z \mu(T-r, Z_{T-r}; \phi, \theta, R, \xi)\right]^{\top} A_r\, dr, \\ G_0^{\phi} &= 0, & dG_r^{\phi} &= \left[\partial_\phi \mu(T-r, Z_{T-r}; \phi, \theta, R, \xi)\right]^{\top} A_r\, dr. \end{aligned}$$

Table 5: BMP regularization and Langevin-correction sweep. Each setting used 250 HBDS draws and one sampling seed. L is the number of corrector steps per diffusion step. Bold marks the lowest error within each column.

	$L=0$		$L=1$		$L=3$		$L=5$	
λ	**RMSE** ↓	**Median** ↓	**RMSE** ↓	**Median** ↓	**RMSE** ↓	**Median** ↓	**RMSE** ↓	**Median** ↓
0	0.29	**12.16**	0.36	**12.96**	0.36	**12.97**	0.36	**12.97**
10^{-4}	0.30	28.15	**0.32**	89.55	**0.32**	90.63	**0.32**	90.87
5×10^{-4}	**0.28**	14.30	0.34	14.89	0.33	14.62	0.34	14.94
10^{-3}	0.34	29.97	0.36	41.20	0.36	41.36	0.36	41.86
10^{-2}	0.33	45.26	0.40	93.51	0.42	94.78	0.41	99.30
10^{-1}	0.30	52.17	0.52	157.55	0.53	153.48	0.48	159.86
1	0.30	47.00	0.40	130.73	0.41	134.90	0.41	139.27

Taking the expectation of G_T^ϕ over paths yields the endpoint reward gradient with respect to ϕ, which enters the minimized objective with a minus sign. The path penalty can be differentiated in the same way by augmenting the sampled state with its accumulated control energy (§ G.2). We initialized (θ, R) at a best-fit least-squares regression (LSR) solution $(\theta^{\mathrm{LSR}}, R^{\mathrm{LSR}})$, which provides a strong starting point for posterior predictive calibration. We note that the choice of conditioning is an important hyperparameter during fine-tuning, similar to the choice of a known pair in Wehenkel et al. (2025). It is also possible to simultaneously optimize these parameters, but doing so introduces new optimization dynamics that we leave for future work.

Regularization and corrector sweeps. We compared seven regularization strengths and $L \in \{0, 1, 3, 5\}$ Langevin corrections per diffusion step using the four-layer FiLM model (Table 5). Each setting used HBDS with 100 diffusion steps, three block-refinement rounds, Tweedie conditioning for both blocks, 250 posterior draws, and sampling seed 42. All settings retained the derived diffusion coefficient, including the predictor-only $L = 0$ case. At $L = 5$, $\lambda = 10^{-4}$ gives the lowest RMSE among positive penalties, while $\lambda = 5 \times 10^{-4}$ gives the lowest median distance. The unregularized model has the lowest median distance overall. In this single-seed sweep, omitting corrections yields lower errors than adding them for every λ, showing that more corrector steps need not improve predictive fit at a fixed discretization. This is a correction-count ablation, not a comparison with the ordinary VP coefficient.

G.1 Path-space formulation and Girsanov's theorem

We briefly state the change-of-measure result used in §§ 3.2 and 3.3. Let $\Omega = C([0, T]; \mathbb{R}^d)$ denote the space of continuous trajectories. Consider two SDEs with the same initial law and scalar diffusion coefficient,

$$dX_t = b_0(X_t, t)\, dt + g(t)\, dW_t, \qquad X_0 \sim p_0, \tag{46}$$

$$dX_t = b_1(X_t, t)\, dt + g(t)\, d\widetilde{W}_t, \qquad X_0 \sim p_0, \tag{47}$$

and let $\mathbb{P}_0$ and $\mathbb{P}_1$ denote their induced probability measures on Ω. Here W and $\widetilde{W}$ are Wiener processes under $\mathbb{P}_0$ and $\mathbb{P}_1$, respectively. Assume the SDEs are well posed, $g(t) > 0$, and the normalized drift change

$$u_t(x) := \frac{b_1(x, t) - b_0(x, t)}{g(t)}$$

is sufficiently integrable for the associated stochastic exponential to be a true martingale under $\mathbb{P}_0$ (Novikov's condition is a standard sufficient condition).

Under these assumptions, Girsanov's theorem gives $\mathbb{P}_1 \ll \mathbb{P}_0$ with Radon–Nikodym derivative

$$\frac{d\mathbb{P}_1}{d\mathbb{P}_0}(X) = \exp\left(\int_0^T u_t(X_t)^\top dW_t - \frac{1}{2} \int_0^T \|u_t(X_t)\|_2^2\, dt \right). \tag{48}$$

Consequently, provided the control energy is integrable under $\mathbb{P}_1$,

$$\mathrm{KL}(\mathbb{P}_1 \,\|\, \mathbb{P}_0) = \frac{1}{2} \mathbb{E}_{\mathbb{P}_1} \left[\int_0^T \|u_t(X_t)\|_2^2\, dt \right]. \tag{49}$$

For the PT and FT reverse SDEs in our setting, the shared starting law is the noise distribution at $t = T$ and the drift difference is $b_\phi - b_{\phi_0} = -g(t)^2(s_\phi^y - s_{\phi_0}^y)$. The normalized reverse-drift change is therefore $u_t = -g(t)(s_\phi^y - s_{\phi_0}^y)$. Rewriting the processes in increasing sampling time reverses this sign but leaves the squared control energy unchanged, yielding Eq. (3).

G.2 Stochastic optimal control connection

Likelihood fine-tuning as drift control. Fix $(\theta^{\mathrm{ref}}, \xi)$ and suppress these conditioning arguments. We retain the reverse-SDE convention in which diffusion time t decreases from T to 0, so sampling

ends at $t = 0$. The PT and FT samplers induce the path laws $\mathbb{P}_{\phi_0}$ and $\mathbb{P}_\phi$, respectively, through

$$\begin{aligned} dY_t^0 &= b_0(Y_t^0, t)\, dt + g(t)\, d\overleftarrow{W}_t, \\ dY_t^\phi &= \big[b_0(Y_t^\phi, t) + g(t) u_\phi(Y_t^\phi, t)\big]\, dt + g(t)\, d\overleftarrow{\widetilde{W}}_t, \\ u_\phi(y,t) &:= g(t)^{-1}\big[b_\phi(y,t) - b_0(y,t)\big] \\ &= -g(t)\big[s_\phi^y(y,t) - s_{\phi_0}^y(y,t)\big]. \end{aligned} \tag{50}$$

Here $b_\phi = b_0 + g u_\phi$ is the FT drift, while W and $\widetilde{W}$ distinguish Wiener processes under the PT and FT path laws, not parameter-dependent noise. Both processes share the diffusion coefficient $g(t)$ and a common noise law at $t = T$. Let $\pi_\phi = \pi^\star(\phi; \theta^{\mathrm{ref}}, \xi)$ denote the corresponding endpoint distribution. Extracting the generated response Y_0 is a measurable projection of the full trajectory, so the data-processing inequality and the control-energy identity in Eq. (3) give

$$\mathrm{KL}(\pi_\phi \| \pi_{\phi_0}) \le \mathrm{KL}(\mathbb{P}_\phi \| \mathbb{P}_{\phi_0}) = \frac{1}{2}\mathbb{E}_{\mathbb{P}_\phi}\int_0^T \|u_\phi(Y_t, t)\|_2^2\, dt,$$

under the assumptions in § G.1. Replacing the endpoint penalty in Eq. (2) by this path penalty gives the parameterized SOC cost

$$\begin{aligned} \mathcal{J}_\lambda(\phi) &:= -\mathbb{E}_{y\sim\pi_\phi}[R(y; y^o)] + \lambda\, \mathrm{KL}(\mathbb{P}_\phi \| \mathbb{P}_{\phi_0}) \\ &= \mathbb{E}_{\mathbb{P}_\phi}\left[-R(Y_0; y^o) + \frac{\lambda}{2}\int_0^T \|u_\phi(Y_t, t)\|_2^2\, dt\right], \end{aligned} \tag{51}$$

This is KL-regularized SOC within the score-network parameterization, with observation loss $-R$ as the sampling-endpoint cost and λ pricing control effort (Uehara et al., 2024b). Since the expected reward is unchanged and $\lambda \ge 0$, data processing gives $\mathcal{F}(\pi^\star(\phi)) \le \mathcal{J}_\lambda(\phi)$. The SOC cost is therefore a path-regularized upper bound on the endpoint objective, not an identical rewrite of it. We optimize this cost using Implicit Diffusion's adjoint sensitivities through the score-network parameters, without learning a separate control network or value function. Rewriting the sampler in increasing sampling time reverses the drift and control signs but leaves the quadratic cost unchanged.

Path-space trust regions. The same penalty admits a constrained interpretation,

$$\min_\phi\ -\mathbb{E}_{\mathbb{P}_\phi}[R(Y_0; y^o)] \qquad \text{subject to} \qquad \mathrm{KL}(\mathbb{P}_\phi \| \mathbb{P}_{\phi_0}) \le \rho, \tag{52}$$

where $\rho \ge 0$ is a divergence budget. Its Lagrangian is $\mathcal{J}_\lambda(\phi) - \lambda\rho$ with multiplier $\lambda \ge 0$, so fixing λ recovers the penalized optimization over ϕ. This motivates choosing the penalty through a divergence budget rather than treating it only as a tuning constant, in the spirit of path-space trust-region methods (Blessing et al., 2025). For multiple observed designs, constraints $\mathrm{KL}(\mathbb{P}_\phi^{\theta^{\mathrm{ref}},\xi} \| \mathbb{P}_{\phi_0}^{\theta^{\mathrm{ref}},\xi}) \le \rho(\xi)$ would introduce design-specific multipliers λ_ξ. Adjusting these against their budgets could regulate how strongly each condition is adapted, alongside a curriculum that selects which designs to revisit. This is a prospective extension, not an implemented adaptive optimizer or a claim of strong duality for the neural parameterization.

G.3 Fisher–Rao optimization analysis

Let $z = (\theta, \xi)$ collect the mechanistic parameters and experimental design. Under the regularity and local identifiability assumptions in § 3.3, the conditional reverse-SDE path laws $\{\mathbb{P}_\phi^z : z \in \Theta \times \Xi\}$ define a local statistical manifold. The path-space Fisher metric in Eq. (4) can be partitioned according to the two types of conditioning variables (Figure 10),

$$\mathcal{I}_\phi^{\mathrm{path}}(z) = \begin{bmatrix} I_{\theta\theta}(z) & I_{\theta\xi}(z) \\ I_{\xi\theta}(z) & I_{\xi\xi}(z) \end{bmatrix}, \tag{53}$$

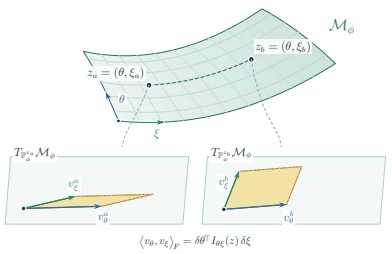

Figure 10: Joint parameter Fisher geometry.

Because the path-space Fisher metric may be singular in locally unidentifiable directions, we use $A^\dagger$ to denote the Moore–Penrose pseudoinverse, which for positive-semidefinite A inverts nonzero eigenvalues and leaves zero eigenvalues unchanged. We write $A^{\dagger/2} = (A^\dagger)^{1/2}$ for its positive-semidefinite pseudoinverse square root. Equivalently, all constructions may be restricted to the identifiable subspace.

For example, the parameter block of Eq. (4) is

$$I_{\theta\theta}(z) = \mathbb{E}_{Y\sim\mathbb{P}^z_\phi}\left[\int_0^T g(t)^2 J_\theta s^y_\phi(Y_t,t;z)^\top J_\theta s^y_\phi(Y_t,t;z)\,dt\right]. \tag{54}$$

As in § 3.3, a small perturbation $\delta z = (\delta\theta, \delta\xi)$ gives

$$\mathrm{KL}\left(\mathbb{P}^{z+\delta z}_\phi \middle\| \mathbb{P}^z_\phi\right) = \frac{1}{2}\delta z^\top \mathcal{I}^{\mathrm{path}}_\phi(z)\delta z + o(\|\delta z\|^2). \tag{55}$$

The block structure in Eq. (53) suggests several extensions of the present fixed-reference fine-tuning procedure.

Refining the mechanistic reference. The block $I_{\theta\theta}$ measures how perturbations of the mechanistic parameters change the conditional path law while holding the experimental design fixed. Thus, rather than fixing θ^{ref} throughout fine-tuning, one could refine it using the geometry induced by the learned likelihood. For an objective $L(\theta)$, a local natural-gradient update is

$$\delta\theta = -\eta\, I_{\theta\theta}(z)^\dagger \nabla_\theta L(\theta), \tag{56}$$

which favors parameter directions that improve the objective while inducing a comparatively small change in the conditional path law.

Using a single fixed θ^{ref} is only the simplest case. More generally, one could maintain a distribution over plausible mechanistic references, approximated by particles

$$q(\theta) \approx \sum_{m=1}^M w_m\, \delta_{\theta^{(m)}}. \tag{57}$$

These particles could be reweighted according to predictive agreement, transported using geometry-aware gradients such as Eq. (56), or updated with interacting-particle methods that preserve diversity. This would avoid forcing likelihood adaptation to be anchored to a single mechanistic explanation.

A distribution over reference parameters also yields an uncertainty-aware version of the adaptation diagnostic. In place of evaluating the path divergence $\mathcal{A}(\theta,\xi)$ from Eq. (3) at one θ^{ref}, define

$$\overline{\mathcal{A}}_q(\xi) = \mathbb{E}_{\theta\sim q}\left[\mathcal{A}(\theta,\xi)\right], \qquad V_{\mathcal{A},q}(\xi) = \mathrm{Var}_{\theta\sim q}\left[\mathcal{A}(\theta,\xi)\right]. \tag{58}$$

The first quantity identifies designs at which substantial adaptation is required across plausible mechanistic parameterizations, whereas the second identifies designs whose apparent misspecification depends strongly on parameter uncertainty. Here $V_{\mathcal{A},q}(\xi)$ quantifies how sensitive the inferred adaptation is to uncertainty in the mechanistic reference θ. This is distinct from Monte Carlo uncertainty arising from the finite trajectory samples used to estimate each $\mathcal{A}(\theta,\xi)$.

Design sensitivity and active simulation. The block $I_{\xi\xi}$ measures how strongly the conditional path law changes under local perturbations of the experimental design. For a small $\delta\xi$,

$$\mathrm{KL}\left(\mathbb{P}^{\theta,\xi+\delta\xi}_\phi \middle\| \mathbb{P}^{\theta,\xi}_\phi\right) \approx \frac{1}{2}\delta\xi^\top I_{\xi\xi}(\theta,\xi)\delta\xi. \tag{59}$$

Applied to the PT model, this geometry can identify regions of design space where the learned simulator surrogate changes rapidly and where denser simulation coverage may therefore be useful. Scalar summaries such as $\mathrm{tr}\, I_{\xi\xi}$ or $\log\det(I_{\xi\xi}+\epsilon I)$ could be used to prioritize additional simulator calls in an active acquisition scheme rather than expanding the pretraining set uniformly. Related work uses diffusion models to propose new simulation parameters based on surrogate loss or uncertainty for online PDE-surrogate training (Cesar et al., 2026). Our proposed acquisition signal instead comes from local path-space sensitivity.

For the FT model, the same quantity instead describes sensitivity of the adapted likelihood over experimental conditions. Designs with large $I_{\xi\xi}$ may merit greater attention in a fine-tuning curriculum, especially when they also exhibit high predictive error or large PT–FT divergence. Importantly, sensitivity is not itself evidence of misspecification: additional PT simulations improve fidelity to the expert simulator, whereas $\mathcal{A}(\theta, \xi)$ measures how strongly fine-tuning moves away from that simulator toward observations.

The same geometry may also support experimental-design objectives aimed at learning mechanistic parameters. At a candidate design ξ, large $I_{\theta\theta}(\theta, \xi)$ indicates that the predictive path law is locally sensitive to changes in θ. Standard optimal-design summaries, for example

$$\log\det\left(I_{\theta\theta}(\theta, \xi) + \epsilon I\right), \tag{60}$$

could therefore be used prospectively to identify experimental conditions that are locally informative about the mechanistic parameterization.

Parameter–design coupling and local identifiability. The cross block $I_{\theta\xi}$ measures the Fisher inner product between path-law changes induced by parameter and design perturbations. A normalized coupling operator is

$$C(\theta, \xi) = I_{\theta\theta}^{\dagger/2} I_{\theta\xi} I_{\xi\xi}^{\dagger/2}. \tag{61}$$

Its singular vectors identify pairs of parameter and design directions that produce similar first-order changes in the conditional path law. Large singular values indicate local parameter–design confounding: a change in the mechanistic parameters can be partially mimicked by moving the experimental condition. Small singular values indicate more geometrically distinct effects.

This coupling can be removed from the parameter block through the Schur complement

$$I_{\theta|\xi}^{\mathrm{eff}} = I_{\theta\theta} - I_{\theta\xi} I_{\xi\xi}^{\dagger} I_{\xi\theta}. \tag{62}$$

Geometrically, $I_{\theta|\xi}^{\mathrm{eff}}$ measures the component of parameter-induced path-law sensitivity that remains after projecting away directions that can be reproduced locally by perturbing the design. It therefore provides a prospective local identifiability diagnostic: small eigenvalues correspond to parameter combinations whose effects are difficult to distinguish from changes in experimental conditions. Conversely, designs for which

$$\log\det\left(I_{\theta|\xi}^{\mathrm{eff}}(\theta, \xi) + \epsilon I\right) \tag{63}$$

is large would be locally informative about mechanistic parameters while reducing redundancy with design-induced variation.

Future work. Although we do not develop these extensions here, they suggest promising ways to improve both PT and FT models in future SBI applications. For example, path-space geometry could guide refinement or particle transport of θ^{ref} and incorporate mechanistic uncertainty into adaptation diagnostics (Eq. (58)). It could also inform simulation acquisition for pretraining, complementing diffusion-based active sampling (Cesar et al., 2026), while supporting design-adaptive fine-tuning curricula and local parameter–design identifiability analysis (Eqs. (61) and (62)).

G.4 Related work on simulator misspecification in SBI

Simulation-based inference (SBI) treats the expert model simulator as an implicit, or intractable, likelihood model from which one can sample, $y \sim p(y|\theta)$, but whose density cannot be evaluated. Starting with samples from a prior $\theta \sim p(\theta)$, synthetic pairs (y, θ) are generated to fit a neural density estimator for the likelihood $p_\phi(y|\theta)$ (Lueckmann et al., 2019; Papamakarios et al., 2019), posterior $p_\phi(\theta|y)$ (Greenberg et al., 2019; Lueckmann et al., 2017; Papamakarios & Murray, 2016; Sharrock et al., 2024), or likelihood-to-evidence ratio (Durkan et al., 2020; Miller et al., 2022) using observed data y_o, enabling parameter inference without explicit likelihoods. Thus, SBI is well-suited for biological simulators that use stochastic differential equations or simulation by convex optimization (Dirks et al., 2007; Su et al., 2022). SBI relies on a simulator that accurately reflects the true, and unknown, data-generating process of the observed data, i.e., $p^\star(y_o) \coloneqq \int p^\star(\theta) p^\star(y_o|\theta) d\theta$, which is typically unavailable and can be the cause of poor performance of SBI models (Cannon et al., 2022; Schmitt et al., 2024). This has resulted in a variety of techniques to improve robustness of

Table 6: Representative approaches to reward- and constraint-based generative model adaptation. Both reward types are scalar. "Reward values only" requires evaluations $R(y)$ but no sample derivatives, whereas "Differentiable reward/loss" uses gradients such as $\nabla_y R(y)$. Reward and loss differ only by sign. "Sampler gradients" describes differentiation through sampling, including adjoint sensitivities, not derivatives used to construct fixed regression targets. Auxiliary objects exclude supplied reward models and the adapted drift/control. The Implicit Diffusion row describes its finite-horizon formulation. Regularization mechanisms and optional ablations are distinguished in the text.

Method	**Feedback / target**	**Update mechanism**	**Sampler gradients**	**Auxiliary object**	**Reference regularization**
DPOK (Fan et al., 2023)	Reward values only	Policy gradient	No	None	Discrete path KL
DDPO (Black et al., 2024)	Reward values only	Policy gradient	No	None	No explicit PT KL
ReFL (Xu et al., 2023)	Differentiable reward/loss	Intermediate clean-prediction gradient	Partial	None	Pretraining denoising loss
AlignProp (Prabhudesai et al., 2023)	Differentiable reward/loss	Rollout backpropagation	Yes	None	No explicit PT KL
DRaFT-K (Clark et al., 2024)	Differentiable reward/loss	Truncated rollout backpropagation	Last K steps	None	LoRA weight decay
ELEGANT (Uehara et al., 2024b)	Differentiable reward/loss	SOC objective via rollout gradients	Yes	Value function and initial-law sampler	Drift-control cost and initial-law KL
Adjoint Matching (Domingo i Enrich et al., 2025)	Differentiable reward/loss	Regression on fixed adjoint targets	No	None	SDE path KL (SOC)
Adjoint Sampling (Havens et al., 2025)	Differentiable unnormalized energy	Adjoint regression with reference bridges	No	None	Control cost, energy target
Implicit Diffusion (Marion et al., 2025)	Differentiable reward/loss	Adjoint sensitivity	Yes	None	Optional SDE path KL
CGM-relax / CGM-reward (Smith et al., 2026)	Moment constraints	Likelihood-ratio gradients	No	Tilt coefficients (reward)	KL to PT (path KL for SDEs)
Value Matching (Jensen et al., 2026)	Non-differentiable reward allowed	Online value-function learning	No	Value function	Control-based

SBI inference to model misspecification (Gao et al., 2023; Huang et al., 2023a; Mishra et al., 2026; Ward et al., 2022; Wehenkel et al., 2025) but there is a gap in methods that *transfer* improvement in predictive accuracy to posterior inference.

G.5 Related work on diffusion fine-tuning

Diffusion fine-tuning is a rapidly growing field with diverse optimization and regularization strategies. We review a selection of approaches relevant to our setting, summarized in Table 6, rather than attempting an exhaustive survey.

Reward optimization and gradient estimators. Reward-based diffusion fine-tuning encompasses policy-gradient and direct-differentiation methods, with different requirements on reward access (Uehara et al., 2024a). DPOK (Fan et al., 2023) and DDPO (Black et al., 2024) treat denoising as a sequential decision process, allowing optimization without differentiating the reward. DPOK explicitly regularizes deviation from the PT model through a KL penalty. When rewards are differentiable, ReFL (Xu et al., 2023) backpropagates feedback from a predicted clean sample at an intermediate denoising state, while AlignProp (Prabhudesai et al., 2023) and DRaFT (Clark et al., 2024) differentiate through the sampling trajectory. AlignProp uses low-rank adaptation and gradient checkpointing to reduce memory requirements. DRaFT-K instead restricts backpropagation to the last K denoising steps, and DRaFT-LV reduces gradient variance for $K = 1$. These approaches trade access to reward gradients and the cost of differentiating long trajectories against the sampling variance of policy-gradient estimates.

Stochastic optimal control. SOC treats fine-tuning as learning a change to the PT drift that improves terminal reward while penalizing the accumulated control effort. With a shared initial law and diffusion coefficient, Girsanov's theorem expresses path-space KL as a quadratic cost of this drift change (§ G.1). This gives a trajectory-level notion of staying close to the PT model, rather than a penalty on network weights. ELEGANT (Uehara et al., 2024b) uses this formulation to optimize both the denoising drift and the initial noise distribution, regularizing both changes. Its implementation estimates a value function and learns an auxiliary process for sampling the adapted initial distribution. Adjoint Matching (Domingo i Enrich et al., 2025) takes a different route, fitting the control to targets obtained from a simplified backward adjoint equation. Trajectories and targets are held fixed during each regression update, avoiding differentiation through the controlled rollout, although reward and base-drift derivatives are still needed to construct the targets. Its reward fine-tuning formulation uses a memoryless noise schedule to recover the intended reward-tilted output distribution without separately learning the initial noise law. Adjoint Sampling (Havens et al., 2025) builds on this regression approach for unnormalized energy targets, using reference bridges to reuse terminal samples and energy gradients across multiple parameter updates to amortize expensive sampling and energy evaluations. These options differ in the auxiliary quantities they learn, their sampling requirements, and whether the target is a reward-adapted PT model or an explicitly specified energy density.

Implicit Diffusion and sampler gradients. Implicit Diffusion (Marion et al., 2025) starts from an objective on the distribution produced by a parameterized sampler and develops joint sampling and parameter-update schemes. For finite-horizon diffusion models, adjoint equations propagate the terminal loss sensitivity backward through the dynamics to estimate parameter gradients. These are gradients through sampling, as in DRaFT, rather than the fixed-target regression updates of Adjoint Matching. Our implementation uses checkpointed backpropagation through the discretized reverse-SDE rollout, which computes a discrete adjoint while trading recomputation for memory. DRaFT-K restricts backpropagation to the last K steps, and DRaFT-LV additionally averages over noise to reduce variance (Clark et al., 2024). Implicit Diffusion also develops joint sampling and optimization updates, so this shared gradient principle does not make the algorithms identical. The gradient method and regularizer remain separate choices, and neither checkpointing nor adjoint sensitivities inherently impose a path-space KL penalty.

Reference preservation. The penalties in Table 6 act on different objects. DRaFT uses AdamW weight decay on its LoRA factors, shrinking the learned weight update toward the PT model (Clark et al., 2024, Appendix A.1). This is parameter-space regularization, not a KL between sampling laws or literally a squared distance between all FT and PT network weights. Its separate KL ablation penalizes PT–FT noise-prediction differences only at the final denoising step (Clark et al., 2024, Appendix B.8). In contrast, DPOK sums expected transition KLs over the discrete denoising chain, yielding a path KL that upper-bounds endpoint KL (Fan et al., 2023). ReFL retains a pretraining denoising loss rather than a PT-distribution KL (Xu et al., 2023). For continuous SDEs with shared diffusion and initial laws, path KL is the integrated drift-control energy in Eq. (49). This is the optional regularizer we adopt with Implicit Diffusion, while ELEGANT additionally penalizes its changed initial noise law. Adjoint Sampling instead uses control costs to reach an energy-defined target, not to preserve a PT data distribution.

Feedback and distributional constraints. Beyond the gradient estimator, the reliability and availability of feedback determine which adaptations are useful. Uehara et al. (2024c) study online fine-tuning with limited reward queries, while BRAID (Uehara et al., 2024d) uses conservative reward estimates to limit overoptimization outside the support of offline data. Calibrating Generative Models (CGM) (Smith et al., 2026) seeks the KL-closest model satisfying moment constraints. CGM-relax penalizes constraint violations, whereas CGM-reward fits exponential-tilt coefficients that define a reward. Both use likelihood-ratio gradients with sampled outputs held fixed, requiring evaluable density ratios and parameter gradients rather than differentiable constraint functions or rollout backpropagation. Their continuous-diffusion formulation uses path-space KL via Girsanov's theorem. Flow Density Control (De Santi et al., 2025) extends optimization to broader utilities and divergences through mirror-flow updates, and Value Matching (Jensen et al., 2026) studies gradient-free reward-guided flow adaptation. These works motivate considering both the computational cost of fine-tuning and whether its objective remains informative away from the observed data.

Why implicit diffusion for SBI? The challenge in our SBI setting is to match a finite set of observed design–response pairs (ξ, y_o) without access to the true data-generating process for additional samples or queries at unmeasured designs. We adapt the conditional likelihood at a simulator-aligned θ^{ref}, using discrepancy from these observations as the learning signal and the PT path law as a regularizing reference. This aims to correct simulator mismatch while retaining the mechanistically interpretable parameter–response relationships learned during pretraining. Because the required correction varies with ξ, choosing which designs to revisit, how to weight their losses, and how strongly to regularize raises a *curriculum* problem (Bengio et al., 2009). Our toy illustrates the tension where updates that correct the misspecified region can degrade regions that were already accurate (§ G.8). We use Implicit Diffusion (Marion et al., 2025) to obtain adjoint gradients of these endpoint losses through the reverse-SDE sampler without learning a separate value function or introducing a control-regression objective, allowing design selection and weighting to enter the loss directly. This leaves the question of where and how strongly to fine-tune, which we can explore by interpreting path divergence $\mathcal{A}(\theta, \xi)$ alongside predictive error. By showing where adaptation improves predictions and where it degrades them, these diagnostics motivate adaptive curricula and design-specific trust regions, with regularization weights λ_ξ adjusted during fine-tuning to control departure from the PT path law (Blessing et al., 2025). Adapting these weights to design-specific divergence budgets would let us regulate likelihood correction at θ^{ref} without imposing the same penalty across all designs. Although this optimization acts on likelihood samples rather than posterior-predictive draws, the shared FiLM-modulated embeddings allow the correction to transfer to posterior queries (§ G.6). Adjoint Matching and Adjoint Sampling (Domingo i Enrich et al., 2025; Havens et al., 2025) offer further routes worth investigating, but adapting their objectives and training procedures to this conditional likelihood setting remains future work.

G.6 Embedding-based transfer learning

Fine-tuning a joint simformer through its likelihood requires the learned correction to transfer to posterior queries, but the original architecture of Gloeckler et al. (2024) zeros condition inputs for latent tokens and leaves this transfer largely to the transformer. We instead apply token-aware Feature-wise Linear Modulation (FiLM) (Perez et al., 2018) before the transformer. For token i, let e_i^{val}, e_i^{id}, and e_i^{cond} denote its value, identity, and latent-or-conditioned state embeddings, and define $b_i = [e_i^{\mathrm{val}}, e_i^{\mathrm{id}}]$. A linear projection of $[e_i^{\mathrm{id}}, e_i^{\mathrm{cond}}]$ produces $[\tilde{\gamma}_i, \tilde{\beta}_i]$, which we bound as $\gamma_i = 1 + \varepsilon \tanh(\tilde{\gamma}_i)$ and $\beta_i = \varepsilon \tanh(\tilde{\beta}_i)$ before forming the transformer input $z_i = [\gamma_i \odot b_i + \beta_i, e_i^{\mathrm{cond}}]$. Thus, the condition state both enters the transformer explicitly and modulates the value–identity representation, while ε limits how far latent and conditioned embeddings can separate; likelihood fine-tuning can therefore update a representation that posterior queries reuse.

G.7 FMCPE-LSR baseline

Protocol. FMCPE first trains a neural posterior estimator (NPE) on simulator-generated pairs and then uses flow matching to correct its predictions with a smaller calibration set (Ruhlmann et al., 2026). For BMP, we used the authors' Lampe/Zuko NPE implementation and treated the 4,816 retained LSR fits as pseudo-calibration parameters. They are estimates from the observed BMP dataset, not ground-truth parameters from a high-fidelity data-generating process. The complete

sweep contains 48 runs: four nested observation budgets (235, 470, 705, and 940 measurements), four nested LSR budgets (1,204, 2,408, 3,612, and 4,816 fits), and three seeds (33, 43, and 53). Observation subsets were balanced across cell lines. For each condition, we trained a dedicated NPE and correction flows, drew 1,024 posterior samples, and evaluated the first 250 through the original one-step BMP simulator over all 940 measurements.

For the simformer comparison, we evaluated the FT model and the PT model from which it was initialized, each with sampling seeds 42, 43 and 44. Every run used 250 joint draws of the 60 biophysical parameters and 20 cell-line receptor concentrations, 20 predictor steps, five corrector steps, three EM steps, hierarchical receptor sampling, and Tweedie conditioning in both parameter blocks. We computed the predictive metrics using the original BMP simulator over all 940 measurements.

Shared predictive metrics. Let $\hat{\mathbf{y}}^{(s)}$ be the original simulator's response vector for posterior draw s, let $\mathbf{y}^o$ contain the 940 observed responses, and let $r_y = \max_i y_i^o - \min_i y_i^o$. We report

$$\mathrm{RMSE}_{\min} = \min_s \frac{1}{r_y}\sqrt{\frac{1}{940}\sum_{i=1}^{940}\left(\hat{y}_i^{(s)} - y_i^o\right)^2}, \qquad d_{\mathrm{med}} = \mathrm{median}_s \left\|\hat{\mathbf{y}}^{(s)} - \mathbf{y}^o\right\|_2 .$$

The first metric evaluates the best predictive draw; the second describes a typical predictive draw. This same RMSE definition is used for every method, with the FMCPE entries taken from its saved 250-draw prediction banks. Both metrics are in response space.

Results. Table 7 reports the predictive errors for FMCPE-LSR and the PT and FT models.

G.8 Design-dependent misspecification toy

Analytic model. The simulator and observation model in § 4.3 share the prior $\theta \sim \mathcal{N}(0, I_2)$ and noise standard deviation $\sigma = 0.2$, but differ by the additive discrepancy

$$h(\xi) = \exp\left(-\frac{\|\xi - c\|_2^2}{2w^2}\right), \qquad c = (1, -0.5)^\top, \quad w = 0.4, \quad \xi \in [-2, 2]^2.$$

Table 7: Predictive errors through the original BMP simulator on all 940 measurements, using 250 parameter draws per run. Values are mean ± SE over three FMCPE training seeds or three sampling seeds of each fixed PT or FT simformer. FMCPE ℓ-C2ST uses separate shared-normalization correction refits and a joint 80-dimensional parameter target.

Method	Observations	LSR fits	RMSE ↓	Median distance ↓	ℓ-C2ST ↓
FMCPE-LSR	235	1,204	0.242 ± 0.004	203.661 ± 38.307	0.198 ± 0.023
FMCPE-LSR	235	2,408	0.231 ± 0.026	203.451 ± 70.019	0.150 ± 0.039
FMCPE-LSR	235	3,612	0.184 ± 0.010	77.887 ± 19.521	0.194 ± 0.050
FMCPE-LSR	235	4,816	0.168 ± 0.011	136.460 ± 77.855	0.157 ± 0.049
FMCPE-LSR	470	1,204	0.256 ± 0.008	283.935 ± 26.484	0.137 ± 0.020
FMCPE-LSR	470	2,408	0.202 ± 0.006	108.803 ± 19.193	0.081 ± 0.013
FMCPE-LSR	470	3,612	0.195 ± 0.003	154.748 ± 46.537	0.095 ± 0.019
FMCPE-LSR	470	4,816	0.177 ± 0.003	31.505 ± 10.206	0.065 ± 0.004
FMCPE-LSR	705	1,204	0.246 ± 0.003	296.322 ± 88.535	0.074 ± 0.027
FMCPE-LSR	705	2,408	0.197 ± 0.014	161.322 ± 41.237	0.099 ± 0.022
FMCPE-LSR	705	3,612	0.193 ± 0.002	98.697 ± 18.042	0.124 ± 0.046
FMCPE-LSR	705	4,816	0.168 ± 0.007	50.355 ± 16.256	0.072 ± 0.036
FMCPE-LSR	940	1,204	0.270 ± 0.012	203.950 ± 20.409	0.083 ± 0.017
FMCPE-LSR	940	2,408	0.210 ± 0.025	188.117 ± 73.589	0.117 ± 0.029
FMCPE-LSR	940	3,612	0.186 ± 0.023	62.011 ± 26.766	0.129 ± 0.010
FMCPE-LSR	940	4,816	$\mathbf{0.175 \pm 0.005}$	31.051 ± 8.460	0.086 ± 0.017
PT BMP Simformer	940	0	0.252 ± 0.003	11.111 ± 0.116	0.157 ± 0.010
FT BMP Simformer	940	1	0.245 ± 0.003	$\mathbf{10.376 \pm 0.058}$	0.189 ± 0.019

Pretraining drew designs uniformly over this square. Each observation panel contains $n = 10$ responses generated at $\theta^\star = (0.8, -1.2)^\top$; the localization run used a separate, manually specified design panel. Fine-tuning conditioned on an LSR estimate fitted under the simulator model, not on $\theta^\star$. We distinguish the misspecified region $h(\xi) \geq 0.1$ from its complement.

The true posterior is available analytically. Let X have rows $\xi_i^\top$, and let $\mathbf{y}^o$ and $\mathbf{h}$ collect the observations and discrepancies $h(\xi_i)$. Then

$$p^\star(\theta \mid \mathbf{y}^o, X) = \mathcal{N}(\mu_\star, \Sigma), \qquad \Sigma = (I_2 + \sigma^{-2} X^\top X)^{-1}, \qquad \mu_\star = \sigma^{-2} \Sigma X^\top (\mathbf{y}^o - \mathbf{h}).$$

The simulator-implied posterior has the same covariance but omits $\mathbf{h}$ from its mean. This gives an exact reference for evaluating posterior correction.

The toy exposes a trade-off hidden by posterior metrics alone: adaptation can move the posterior toward the truth while degrading a likelihood that was already accurate away from the discrepancy. More aggressive full-network fine-tuning reduces posterior W_2 further to 0.03, but raises correct-region likelihood RMSE to 0.32.

Design-localized error and adaptation. Figure 5 uses a separate full-network fine-tuning run with joint two-dimensional RBF design embeddings and $\lambda = 0.1$. We estimated the PT and FT likelihood means using 16 draws per design at the same fixed LSR reference and evaluated their absolute errors against the known noise-free response at $\theta^\star$. The displayed e_{PT} and e_{FT} maps share a color scale, and their annotations average these pointwise errors over all 1,681 design-grid points. This run is distinct from the posterior benchmark above and supplies the $\lambda = 0.1$ setting in the regularization sweep below.

Mean absolute error decreases from 0.31 to 0.20 inside the misspecified region ($h \geq 0.1$), but increases from 0.11 to 0.15 outside it. Path divergence has a Pearson correlation of 0.75 with h, and its mean inside the misspecified region is 4.73 times that outside. Thus, concentrated adaptation can coexist with both local correction and deterioration elsewhere; path divergence alone does not determine whether predictions improve. This analysis used one training seed, one dataset of 10 observations, and one generating parameter vector, so it is a controlled diagnostic rather than a multi-seed benchmark.

Regularization trade-off. Figure 11 varies λ for the joint-RBF model while holding the PT model, observation panel, LSR reference, and remaining training settings fixed. Strong regularization reduces response adjustment and path divergence, retaining more of the PT model, but also weakens correction near the omitted bump. Increasing λ from 1 to 10 reduces RMSE outside the misspecified region by about 3%, while increasing it inside from 0.25 to 0.29. None of the tested FT models improves full-grid RMSE over PT. Here prediction error compares the FT mean at θ^{LSR} with the noiseless true response at $\theta^\star$, so it includes reference-parameter error. This trade-off motivates a design-dependent penalty $\lambda(\xi)$ that preserves well-modeled regions while permitting correction where needed; testing that extension remains future work.

Sampling-time autoguidance. Inspired by autoguidance (Karras et al., 2024), we tested score blending as a sampling-time alternative to changing the training penalty. Using the frozen PT and FT models from Figure 5, we sampled with $s_w^y = s_{\mathrm{PT}}^y + w(s_{\mathrm{FT}}^y - s_{\mathrm{PT}}^y)$ at the same LSR reference, where $w = 0$ recovers pretraining and $w = 1$ recovers fine-tuning. Here the endpoints differ through

Table 8: Fine-tuning on the analytic design-misspecification toy. Posterior errors are measured against the exact true posterior; likelihood RMSE is evaluated in correctly specified and misspecified regions of the design space. The improvement factor is the PT error divided by the FT error, so values above $1\times$ favor fine-tuning and values below $1\times$ indicate degradation.

Metric	PT	FT	Improvement (PT/FT)
Posterior $W_2 \downarrow$	0.09	0.07	$1.35\times$
Posterior mean error $\downarrow$	0.08	0.03	$2.88\times$
Correct-region likelihood RMSE $\downarrow$	0.05	0.26	$0.18\times$
Misspecified-region likelihood RMSE $\downarrow$	0.45	0.30	$1.49\times$

misspecification correction, rather than training quality alone. Figure 12 shows that intermediate weights can retain correction near the discrepancy while attenuating some errors elsewhere. With 64 response draws per design, 100 reverse steps, and matched sampling noise across weights, $w = 0.5$ gives full-grid mean absolute error 0.12, compared with 0.14 for PT and 0.15 for FT. The full sweep also tested $w \in \{1.25, 1.5, 2\}$, which increase full-grid error beyond FT. The lower row estimates PT-relative path divergence along each weight's own reverse trajectories, using 64 paths per design rather than rescaling the FT map; this discretized estimate excludes the final Tweedie step. The best weight was selected retrospectively in this single-seed comparison, and improvement is not uniform across designs. These results suggest a sampling-time control for over-adaptation.

We interpret path divergence alongside the pointwise prediction errors $e_{\mathrm{PT}}(\xi)$ and $e_{\mathrm{FT}}(\xi)$ defined in § 4.3.

Table 9: Interpretation of design-localized fine-tuning. Path divergence $\mathcal{A}(\theta, \xi)$ measures departure from the PT model at fixed θ, while predictive error change is evaluated against observations at the same design ξ.

$\mathcal{A}(\theta, \xi)$	**Error improves**	**Error does not improve**
Small	Adequate simulator or mild correction	Under-correction
Large	Corrected misspecification	Over-correction or forgetting

Table 9 relates the magnitude of adaptation to its predictive benefit at each design. Large path divergence with reduced error is consistent with misspecification correction, whereas large divergence without improvement suggests unnecessary adaptation or forgetting. Small divergence with persistent error indicates under-correction, while small divergence where predictions are already accurate suggests that little correction is needed.

G.9 BMP MISSPECIFICATION EVALUATION

LSR cost groups. The LSR ensemble was generated by fitting the expert simulator to the same BMP observations across four cell lines in $6{,}000$ bounded least-squares runs with different random initializations. Each run minimized squared prediction error and yielded a candidate parameter set, providing alternative fits to the same data. We used the $4{,}816$ fits retained in the original filtered bank provided by Klumpe et al. (2022). Each fit contains 60 biophysical parameters and five receptor multiplicative factors, giving 65 features. We transformed these features to model-normalized coordinates, z-scored each feature across fits, and computed PCA (Figure 13). To group solutions by parameter structure and fit quality, we applied k-means with nine clusters to standardized PC1,

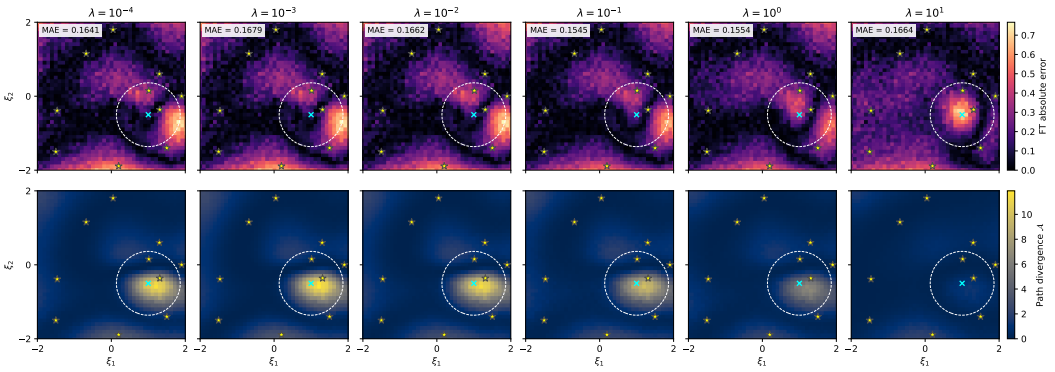

Figure 11: Regularization sweep on the joint-RBF misspecification toy. Columns increase λ from 10^{-4} to 10. (*Top*) FT absolute prediction error $e_{\mathrm{FT}}(\xi)$, with full-grid mean absolute error annotated. (*Bottom*) Path divergence $\mathcal{A}(\xi)$. Stars mark observed designs; dashed contours enclose $h(\xi) \geq 0.1$. Color scales are shared within rows. The sweep used one training seed and 16 Monte Carlo draws per design to estimate response means.

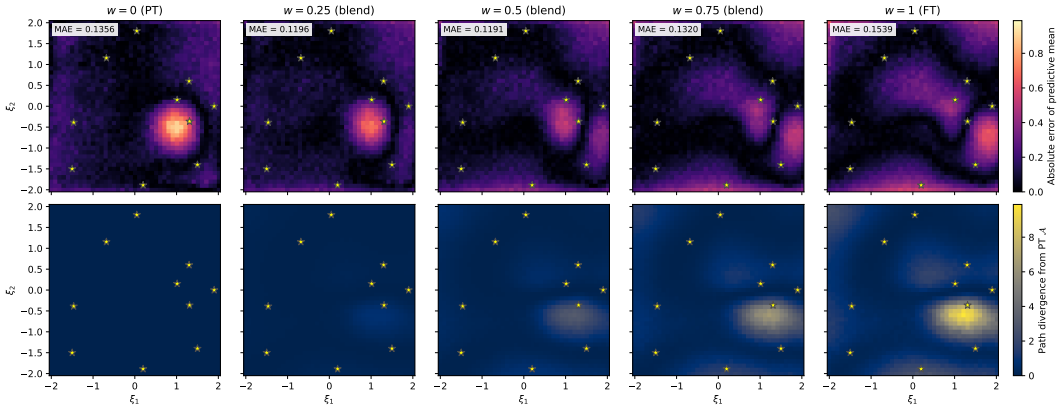

Figure 12: PT–FT score blending on the misspecification toy. Columns vary the sampling weight w from PT (0) to FT (1), without retraining. (*Top*) Absolute error of the predictive mean against the noise-free response, with full-grid mean absolute error annotated. (*Bottom*) Estimated path divergence from PT. Stars mark observed designs; color scales are shared within rows. Intermediate weights reduce average error while limiting departure from PT.

PC2, and $\log_{10} C$, where C is the total LSR fitting cost, multiplying the standardized cost coordinate by eight. We ordered clusters by their mean log-cost and merged clusters 1–3, 4–6, and 7–9 into low (L), middle (M), and high (H) cost groups, respectively. These labels summarize global fit quality, not performance at every design. For the local group assignments in Figures 6 and 14, we computed MSE across member fits at each design and selected the group with the lowest value. Thus, a globally higher-cost group can still fit a particular experimental condition best, as explicitly shown in Figure 14A.

Ligand-competition and titration conditions. The three response curves in Figure 6 probe competition for receptors and dose-dependent signaling. The ligand-competition axis transitions from high BMP4 and low BMP10 to high BMP10 and low BMP4. The NMuMG response changes along this axis, whereas the BMPR2-knockdown line remains close to zero. The BMP4 titration in NMuMG provides a dose–response comparison that also includes a decrease in observed signal at the highest concentration. We held the mechanistic parameters at θ^{LSR} and sampled the FT likelihood, isolating response correction from changes in inferred parameters. The FT model improves over the simulator at this reference throughout the displayed BMPR2-knockdown series and at lower BMP4 doses, while the NMuMG competition series and higher-dose titration retain regions where LSR predicts better. These conditional-likelihood comparisons are distinct from the aggregate HBDS posterior-predictive benchmark in Table 3.

Design-level adaptation. We relate the design-conditional path divergence in Eq. (3) to the LSR cost groups defined above. Panel A of Figure 14 shows which group has the lowest MSE at each experimental condition. Although the low-cost group dominates, other groups fit better at particular ligand conditions and cell lines. These preferences describe relative fit among groups of parameter estimates, rather than establishing simulator misspecification on their own.

Comparing panels A and B reveals a qualitative correspondence between some departures from the dominant LSR group and elevated path divergence $\mathcal{A}(\theta, \xi)$. Panels C and D then distinguish adaptation from predictive improvement by comparing FT responses with the selected LSR anchor and the PT bootstrap median, respectively. Fine-tuning improves over the PT model across many conditions, but its comparison with LSR remains mixed. Nevertheless, improvements over LSR occur both at conditions favoring alternative parameter groups and at some conditions where the dominant group remains preferred. The green-boxed examples connect these design-level diagnostics to the response curves in Figure 6.

Together, these results provide a proof of concept for design-localized likelihood correction where fine-tuning can improve predictions beyond the selected mechanistic fit on particular conditions, while path divergence identifies where the likelihood changes. The remaining regions where LSR performs

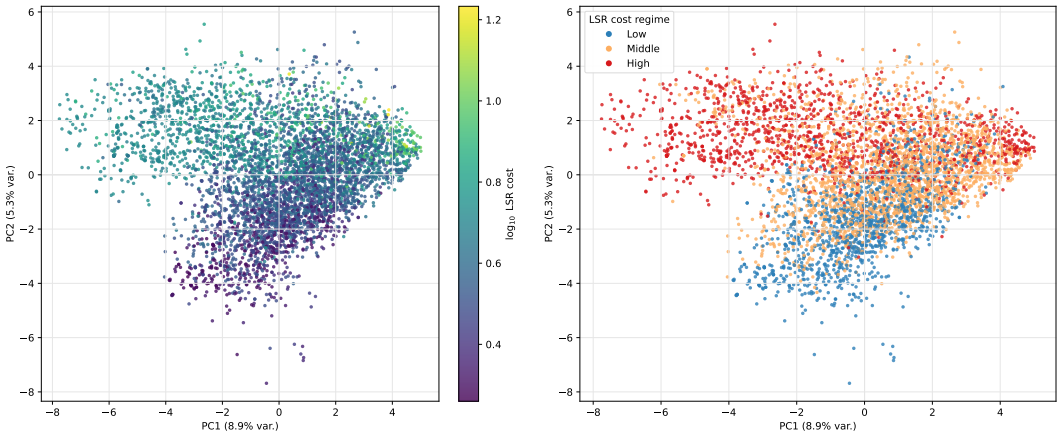

Figure 13: LSR solution structure and cost groups across $4{,}816$ retained fits. (*Left*) PCA coordinates colored by $\log_{10}$ total LSR cost. (*Right*) The same projection colored by low, middle, and high cost groups, containing $1{,}296$, $2{,}245$, and $1{,}275$ fits. PC1 and PC2 explain 8.9% and 5.3% of the standardized feature variance. Grouping also uses fitting cost, so overlap in this two-dimensional projection is expected.

better also show that the current fine-tuning procedure does not fully exploit this potential, or, can be improved. This motivates the design-adaptive curricula and divergence-aware regularization discussed in §§ G.2 and G.3, targeting residual error while preserving successful corrections. Establishing whether these extensions yield broader gains requires further evaluation, which we leave to future work.

Posterior transfer without collapse. Although likelihood fine-tuning is anchored at a fixed reference θ^{LSR}, the resulting posterior does not collapse onto this point estimate. Figure 15 shows that FT posterior marginals generally shift toward the reference while retaining substantial spread, indicating that the likelihood-side correction transfers to posterior queries without reducing inference to the fine-tuning anchor.

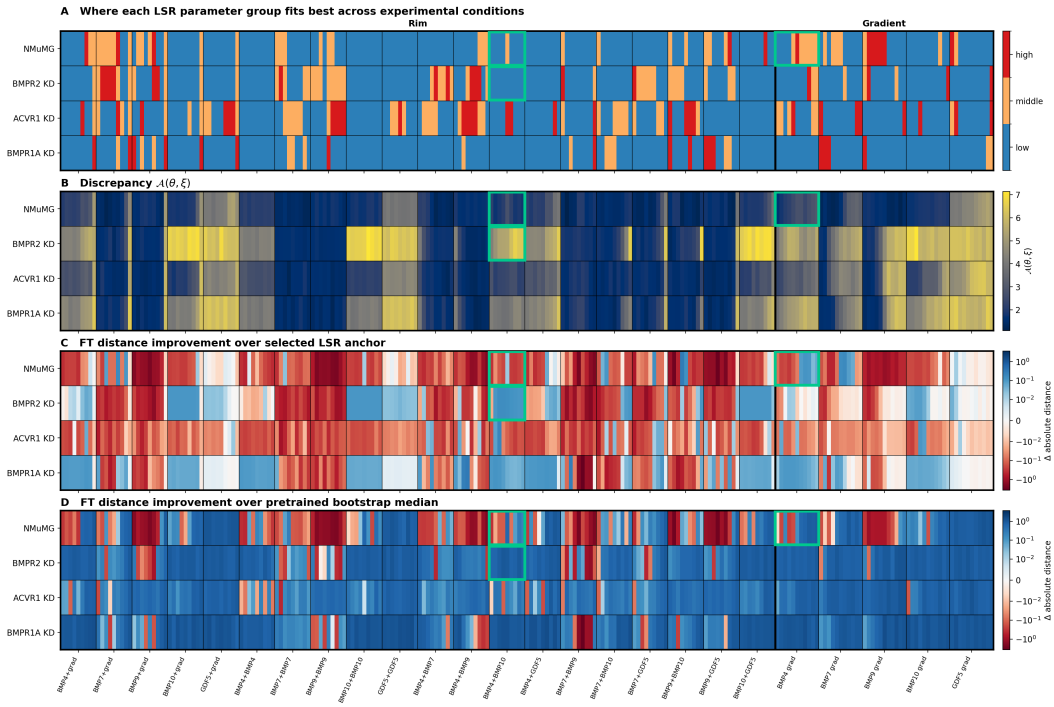

Figure 14: Design-localized adaptation in the BMP model across four cell lines (rows) and experimental conditions (columns). Thin black lines separate titration series; the thick black line separates ligand-competition ("Rim") from gradient series ("Gradient"). Green boxes mark the conditions illustrated in Figure 6. (*A*) Best-fitting LSR parameter group; colors identify groups, not error magnitudes. (*B*) Path divergence $\mathcal{A}(\theta, \xi)$, with yellow indicating greater adaptation. (*C*) Fine-tuning distance improvement relative to the selected LSR anchor. (*D*) Fine-tuning distance improvement relative to the PT bootstrap median. In C and D, blue indicates reduced distance to observations and red indicates increased distance relative to the comparator.

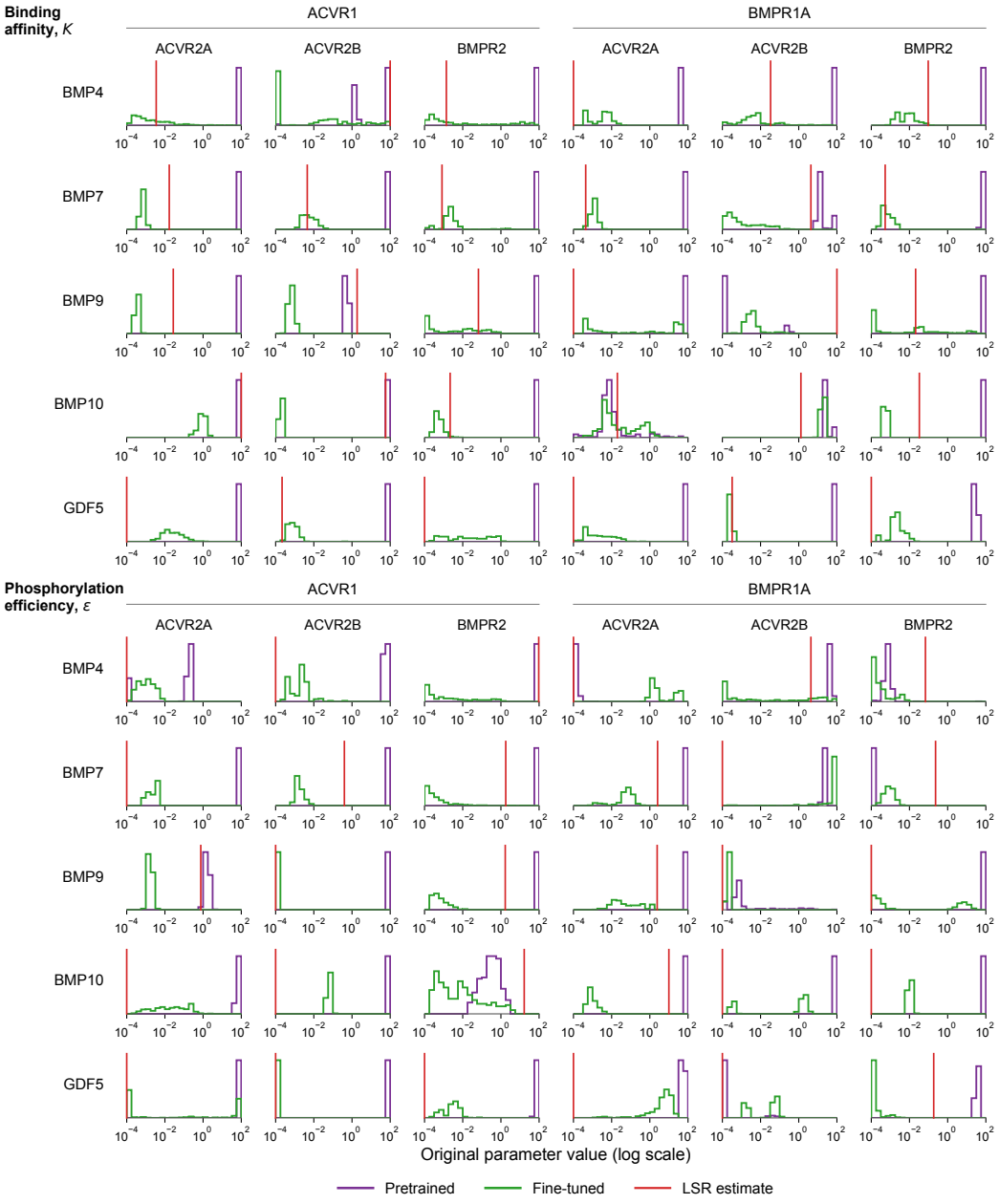

Figure 15: BMP posterior alignment with the LSR reference. Purple and green show PT and FT marginals; red lines mark the fitted LSR reference. The upper and lower blocks show 30 binding affinities K and 30 phosphorylation efficiencies ε, with ligand rows and receptor-complex columns. Type-I receptors label the column groups above; type-II receptors label individual columns beneath them, identifying each receptor pair. FT marginals generally move closer to the reference while retaining spread rather than collapsing to it.

H AI USE STATEMENT

AI tools assisted with grammar, organization, TikZ figures, and experiment management. The authors take responsibility for the final content.