ABC-D^2: GRADIENT-INFORMED INFERENCE FOR DISCRETE STOCHASTIC SIMULATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sequential Monte Carlo approximate Bayesian computation (SMC-ABC) enables likelihood-free Bayesian inference with expensive simulators, by distributing simulations across parallel workers. However, its random-walk moves can become ineffective as the parameter dimension grows and when the particle population is small. Gradient-informed proposals offer a potential solution, but stochastic simulators used in scientific contexts such as population genetics, epidemiology and chemical physics often use discrete distributions, which prevent conventional pathwise differentiation. We introduce ABC-D^2, a Metropolis-adjusted Langevin method that uses discrete stochastic automatic differentiation (D^2) to guide particle mutation in SMC-ABC, with unbiased and low-variance gradient estimates of an expected log-kernel ABC objective from a single simulated dataset and without resorting to relaxations or learned surrogates. We prove that drift along an unbiased gradient estimate improves the expected objective at the proposal over a random walk with the same covariance, to leading order, which a biased drift does not guarantee, and that fixing the drift's random numbers within each sweep keeps the Metropolis-Hastings correction exact. On Bernoulli generalised linear models, Gaussian mixture models and a multi-population SIR epidemic model, ABC-D^2 reduces posterior-mean error of random-walk SMC-ABC by up to two-thirds and matches the random walk's accuracy with at least an order of magnitude fewer particles. These results demonstrate the potential of discrete stochastic differentiation to improve the accuracy of SMC-ABC for discrete simulators in common scientific applications, where conventional gradient-based proposals cannot be deployed.

1 INTRODUCTION

Bayesian inference provides a framework to quantify uncertainty in unknown parameters by combining prior information with observed data. In many scientific models, however, evaluating the likelihood is computationally prohibitive or analytically intractable, even though generating data from the model is straightforward. Approximate Bayesian computation (ABC) addresses this setting by replacing likelihood calculation with simulation: parameters are retained when simulated data, or their summary statistics, are sufficiently close to the observed data (Sisson et al., 2018; Pritchard et al., 1999; Marjoram et al., 2003). This makes ABC applicable to a broad class of models, whether deterministic or stochastic, continuous or discrete. Discrete stochastic models are particularly common in scientific applications, including reaction events in stochastic chemical kinetics (Gillespie, 1977; Toni et al., 2009), disease transmission in epidemiology (McKinley et al., 2009; Kypraios et al., 2017), and birth-death or coalescent events in population genetics (Kingman, 1982; Beaumont et al., 2002). Although alternatively neural simulation-based inference (SBI) can amortise inference across datasets (Cranmer et al., 2020; Lueckmann et al., 2021), its accuracy may suffer when observation data is different from the simulations used for training, as reported in some biological applications (Wang et al., 2024). Under model misspecification experiments, SBI methods' accuracy and calibration often deteriorate, while ABC is comparatively robust on some tasks (Cannon et al., 2022), although ABC can converge to a pseudo-true parameter (Frazier et al., 2020). ABC thus remains widely used and well characterised amid rapidly evolving SBI methods.

ABC with Markov chain Monte Carlo (MCMC) embeds this simulation-and-comparison step within a Metropolis-Hastings chain (Marjoram et al., 2003). In its common random-walk form, it perturbs

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.

the current parameter, simulates the data at the proposed point and accepts only if the distance between the simulated and observed data, or their summary statistics, is sufficiently small. A common problem with ABC-MCMC is that the chain can mix slowly and be trapped in regions with low ABC probability (Sisson et al., 2007). These challenges are exacerbated as the parameter dimension of the model increases: each perturbed coordinate adds variance to the simulated data and lowers the acceptance rate, so random walk steps must shrink and the chain explores the parameter space diffusively (Roberts et al., 1997). If the observed data or summaries also grow in dimension, simulations must match the observations along more directions (Blum, 2010).

Gradient-informed proposals offer a natural way to cope with these limitations. Methods such as the Metropolis-adjusted Langevin algorithm (MALA) and Hamiltonian Monte Carlo (HMC) use the gradient of the log target, or its ABC approximation, to direct proposals toward regions with higher ABC probability (Roberts & Tweedie, 1996; Duane et al., 1987; Neal, 2011; Meeds et al., 2015). HMC type approaches can be costly when simulations are expensive, since each trajectory requires multiple sequential evaluations of the target and its gradient. Sequential Monte Carlo (SMC) offers a different trade-off, where it can update many particles in parallel with only short chains per SMC round (Sisson et al., 2007; Toni et al., 2009; Beaumont et al., 2009). When a simulator is smoothly differentiable with respect to its parameters, we can propagate gradients through the simulation (Graham & Storkey, 2017; Moreno et al., 2016). However, many simulators are not pathwise differentiable. For instance, a simulator incorporating a Bernoulli draw will have outcomes that change discontinuously as its success probability changes; the same implications arise for other discrete random primitives that appear throughout the simulator.

Several approaches estimate gradients of stochastic programs with discrete randomness. Continuous relaxations, such as the Gumbel-softmax (Jang et al., 2017; Maddison et al., 2017), replace discrete draws by differentiable approximations that generally introduce bias. Finite differences need extra simulations for every coordinate and are biased for a finite step, even when common random numbers reduce their variance (Meeds et al., 2015; Cao et al., 2024), and score-function estimators are unbiased but typically have high variance (Glynn, 1990; Williams, 1992). Arya et al. (2022) introduced stochastic automatic differentiation (stochastic AD), a composable alternative that produces unbiased, low-variance derivatives for programs that include discrete randomness. Alongside the usual derivative, it propagates weighted alternative outcomes of each discrete draw through the program and combines them into an unbiased single-sample estimate of the derivative of the expected output.

Here, we use stochastic AD to construct MALA proposals for ABC simulators with discrete randomness. Each gradient estimate costs additional simulations, which make a single long ABC-MCMC or HMC chain unattractive when simulators are costly. We therefore work within SMC-ABC, which propagates a population of weighted particles through ABC targets with decreasing tolerances, so gradient evaluations are distributed across particles and run in parallel. Specifically, we replace the random-walk rejuvenation of the adaptive SMC-ABC sampler of Del Moral et al. (2012) with MALA proposals whose drift is estimated by stochastic AD, and call the method ABC-D^2, where D^2 denotes differentiability and discreteness. Gradients help for a simple reason: a random walk perturbs the parameter equally in all directions, so its expected first-order change in the ABC objective is zero, whereas a drift along an unbiased gradient estimate moves proposals uphill on average. We make this precise with a local improvement guarantee over the random walk (Theorem 1) and show that fixing the drift's random numbers within each sweep keeps the Metropolis-Hastings (MH) correction exact (Proposition 1).

We compare ABC-D^2 with random-walk SMC-ABC on a Bernoulli generalised linear model (GLM), a shared-label Gaussian mixture model (GMM), and a multi-population SIR epidemic model with binomial observations, for 5 to 640 particles and up to 30 parameters. ABC-D^2 reduces the posterior-mean error in most settings, by up to two-thirds; in intermediate- and high-dimensional GLM and GMM settings, where the random walk stalls at high tolerances, it matches the random walk's accuracy with 16 to 64 times fewer particles.

Related work. Population and sequential Monte Carlo ABC samplers (Sisson et al., 2007; Toni et al., 2009; Beaumont et al., 2009) differ mainly in how they choose tolerances and mutation kernels. Filippi et al. (2013) find that locally adaptive kernels, and kernels built from Fisher information when it can be estimated, are particularly efficient. We keep a particle covariance proposal and add

a drift computed from the simulator. With regard to ABC in high dimensions, ABC accuracy is known to deteriorate as the dimension of the summaries grows (Blum, 2010), which has motivated low dimensional summaries (Fearnhead & Prangle, 2012), regression and copula approximations (Nott et al., 2014; Li et al., 2017), and parameter-wise updates (Kousathanas et al., 2016). These change the discrepancy or the posterior approximation, whereas we tackle a different problem: we change the proposal to speed up exploration of a high-dimensional parameter space, which is slow even when the summaries are relatively low-dimensional. Simulator gradients have been used for likelihood-free inference through stochastic finite differences of a synthetic likelihood in Hamiltonian ABC (Meeds et al., 2015), constrained Hamiltonian dynamics for differentiable generative models (Graham & Storkey, 2017) and joint scores extracted from the simulator to train neural estimators (Brehmer et al., 2020). These need finite differences, a differentiable generator, or the joint density of the simulator's latent variables, whereas stochastic AD differentiates the simulator program directly, without relaxations or learned surrogates. To our knowledge, none of these approaches guarantees that the gradient-informed proposals improve on a random walk: Theorem 1 gives such a guarantee, to leading order in step size, for any unbiased gradient estimate, including pathwise gradients of differentiable simulators. Within SMC, Langevin updates have used gradients from differentiable particle filters (Rosato et al., 2024) and Hamiltonian moves can be tuned adaptively (Buchholz et al., 2021). Neural simulation-based inference instead learns surrogate likelihoods or posteriors (Papamakarios & Murray, 2016; Greenberg et al., 2019; Cranmer et al., 2020; Lueckmann et al., 2021).

Contributions. Our contributions are as follows:

1. We introduce ABC-D^2, an adaptive SMC-ABC sampler with a Langevin mutation kernel whose drift is an unbiased stochastic-AD estimate of the gradient of the expected log-kernel ABC objective (Section 2).
2. We show that an unbiased drift guarantees improvement in the expected objective at the proposal above that of a random walk, to leading order (Theorem 1). We also show that by fixing the drift's random numbers within each sweep, the proposal density can be made computable in both directions, so the MH correction leaves the ABC target invariant (Proposition 1).
3. On GLM, Gaussian mixture and multi-population SIR benchmarks with $5 - 30$ parameters and $5 - 640$ particles, ABC-D^2 reduces the posterior-mean error by up to 68%. In the GLM with $15 - 30$ and the GMM with $10 - 16$ parameters, 10 to 40 ABC-D^2 particles are as accurate as random walk SMC-ABC with 640 particles (within a 10% margin) and therefore require fewer simulations (Section 3). In the SIR model, 2 to 4 times fewer particles suffice ($d \geq 10$).

2 ABC WITH DISCRETE AUTOMATIC DIFFERENTIATION

In this work we develop ABC-D^2. We keep the adaptive SMC-ABC sampler of Del Moral et al. (2012), which moves each particle by an MH step with a Gaussian random-walk proposal (Section 2.2), and change only this proposal: ABC-D^2 drifts its mean along an unbiased estimate of the gradient of an expected log-kernel ABC objective (Sections 2.3 and 2.4). Proposals are still accepted with the usual hard ABC criterion, based on the fraction of M simulated datasets within the tolerance. The gradient therefore enters only through the proposal mean, so its quality governs efficiency, and validity is given by the MH correction together with the standard SMC reweighting. Figure 1 summarises one ABC-D^2 step.

2.1 SETTING AND NOTATION

Let $\theta \in \Theta \subseteq \mathbb{R}^d$ be the parameter, let p_0 be the prior density, and $Y = s(U;\theta) \sim p_\theta$ a simulated dataset, where p_θ is the intractable likelihood on the data space and $U \sim \nu$ is parameter-independent base randomness. A summary map S and discrepancy D then define the realised ABC distance (Pritchard et al., 1999; Beaumont et al., 2002) between Y and the observed data y:

$$\Delta(\theta, U) := D\big(S(s(U;\theta)), S(y)\big), \quad L_\varepsilon(\theta) = \Pr\{\Delta(\theta, U) \leq \varepsilon\}. \tag{1}$$

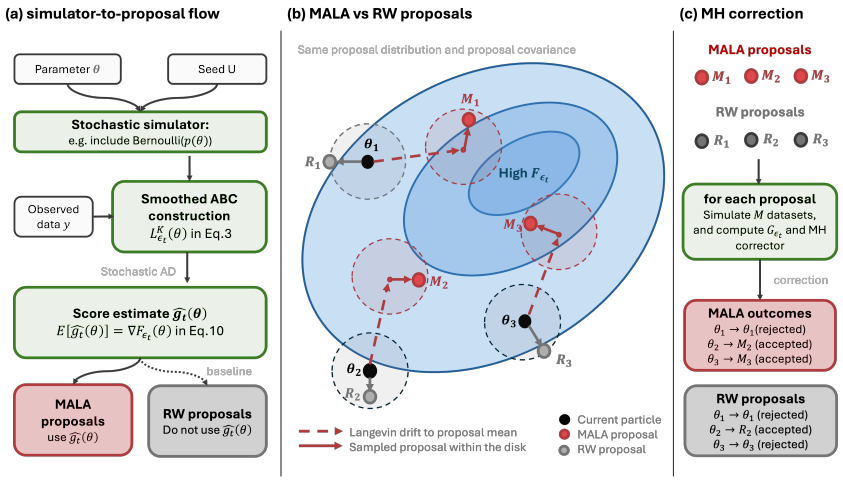

Figure 1: Schematic of the ABC-D^2 SMC-ABC sampler: (a) for each particle, stochastic AD estimates the gradient of the smoothed ABC log target, which we use to construct MALA proposals, with random-walk proposals shown for comparison; (b) ABC-D^2 with MALA proposes particles from the drifted mean, whereas random-walk perturbation proposals remain centred at the current particles. Intuitively, the drift guides particles toward regions of higher log kernel ABC target density. (c) The Metropolis-Hastings (MH) correction accepts or rejects each proposal using M simulated datasets.

The criterion is hard: for a given U, the simulated data either are or are not within distance ε of y, so ε is the ABC tolerance and $L_\varepsilon(\theta)$ is the probability over U of acceptance. The hard ABC posterior is $\pi_\varepsilon(\theta) \propto p_0(\theta)L_\varepsilon(\theta)$. Because the observed data y are fixed throughout, we suppress the conditioning on y and write $\pi_\varepsilon(\theta)$ for $\pi_\varepsilon(\theta|y)$. The simulator can contain discrete random primitives, which make $\Delta(\theta, U)$ jump as θ changes, so it is not pathwise differentiable.

Each particle carries M simulated datasets, and the sampler targets a joint distribution over the parameter and these datasets, the extended target is

$$\widetilde{\pi}_\varepsilon(\theta, Y^{1:M}) \propto p_0(\theta) \prod_{m=1}^{M} p_\theta(Y^m) G_\varepsilon(Y^{1:M}),$$
$$G_\varepsilon(Y^{1:M}) = \frac{1}{M} \sum_{m=1}^{M} \mathbf{1}\{D(S(Y^m), S(y)) \le \varepsilon\}, \tag{2}$$

where G_ε is the fraction of the M datasets within distance ε of the observed data. The θ marginal of equation 2 is the hard ABC posterior, because $\mathbb{E}[G_\varepsilon(Y^{1:M})|\theta] = L_\varepsilon(\theta)$ (Del Moral et al., 2012, Section 2.2). Separately, to define a proposal objective (Section 2.3), we use the smooth-kernel ABC likelihood (Wilkinson, 2013; Beaumont et al., 2002)

$$L^K_\varepsilon(\theta) = \mathbb{E}_U\left[K_\varepsilon\big(\Delta(\theta, U)\big)\right], \quad K_\varepsilon(r) = \exp\left(-\frac{r^2}{2\varepsilon^2}\right). \tag{3}$$

We use it only to construct proposals. A hard indicator is piecewise constant in the distance and gives no direction to a particle whose simulations are far from the tolerance, whereas the Gaussian kernel decreases smoothly with the distance.

2.2 The adaptive SMC-ABC method

The sampler of Del Moral et al. (2012) runs over sweeps $t = 0, 1, \ldots$ with a non-increasing tolerance schedule and N weighted particles $\{(\theta_t^{(i)}, Y_t^{(i),1:M}, W_t^{(i)})\}_{i=1}^N$, each storing the M realised

distances $\Delta_t^{(i),m}$ of its datasets. Sweep 0 draws N particles from the prior, simulates M datasets per particle, and sets $\varepsilon_0 = \infty$ with uniform weights. Each subsequent sweep performs three steps.

Step 1 (reweight). The new tolerance $\varepsilon_t < \varepsilon_{t-1}$ is set by the schedule below, and the weights are updated using only the stored distances:

$$W_t^{(i)} \propto W_{t-1}^{(i)} \frac{G_{\varepsilon_t}\big(\theta_{t-1}^{(i)}, U_{t-1}^{(i),1:M}\big)}{G_{\varepsilon_{t-1}}\big(\theta_{t-1}^{(i)}, U_{t-1}^{(i),1:M}\big)}, \tag{4}$$

with dead particles ($G_{\varepsilon_t} = 0$) kept at weight zero.

Step 2 (resample). When the effective sample size $\mathrm{ESS}_t = \left(\sum_i (W_t^{(i)})^2\right)^{-1}$, computed from the normalised weights, falls below $N_T = N/2$, resample systematically (Kitagawa, 1996) and reset the weights to $1/N$.

Step 3 (move). Each particle with positive weight is moved by one MH step that leaves $\widetilde{\pi}_{\varepsilon_t}$ in equation 2 invariant (Section 2.4).

Tolerance schedule. We use the adaptive ESS matching schedule of Del Moral et al. (2012, Section 3.1), where ε_t is set such that it solves

$$\mathrm{ESS}(\varepsilon_t) = \alpha \mathrm{ESS}(\varepsilon_{t-1}), \quad \alpha \in (0,1), \tag{5}$$

by bisection over $\varepsilon_t \in [\varepsilon^\star, \varepsilon_{t-1}]$, where $\mathrm{ESS}(\varepsilon)$ is the ESS of the weights equation 4 at candidate tolerance ε, which is a function of the stored distances. If the solution falls below the target tolerance $\varepsilon^\star$, then $\varepsilon_t \leftarrow \varepsilon^\star$. Runs terminate on reaching $\varepsilon^\star$, or for hard problems with no natural target, when the MH acceptance rate of Step 3 drops below 1.5%, which then determines the tolerance the algorithm can reach (Del Moral et al., 2012, Section 3.4).

2.3 The gradient estimator

Stochastic AD for discrete simulators Discrete components in simulators introduce jumps that standard pathwise differentiation cannot measure. We therefore turn to stochastic AD following Arya et al. (2022), to give an unbiased estimate of the gradient. Stochastic AD augments each intermediate variable of the simulator program with a stochastic triple $(X, \dot{X}, \{(w, X')\})$: the primal value X, an almost sure derivative $\dot{X}$ propagated as in ordinary forward mode AD (Baydin et al., 2018), and w weighted finite perturbations corresponding to alternate discrete outcomes X'.

Composing triples through deterministic operations obeys the usual product, quotient and chain rules for $\dot{X}$, and maps perturbations forward. Deterministic operations propagate both smooth derivatives and finite alternatives. At the output the derivative becomes

$$\widehat{\partial}_v f = \dot{f} + \sum_{(w,f')} w\big(f' - f\big), \tag{6}$$

a single sample estimate of $\partial_v \mathbb{E}[f(\theta)]$, unbiased under the stochastic derivative regularity and integrability conditions of Arya et al. (2022). When several perturbations arise, we keep one, chosen with probability proportional to its weight, and give it the total weight. This pruning strategy of Arya et al. (2022) keeps the estimate unbiased and needs one alternative evaluation per coordinate.

The differentiated objective At the staged tolerance ε_t we use one simulated dataset to form the random proposal objective

$$\begin{aligned} \widehat{\ell}_{\varepsilon_t}(\theta, U) &= \log p_0(\theta) + \log K_{\varepsilon_t}(\Delta(\theta, U)) \\ &= \log p_0(\theta) - \frac{\Delta(\theta, U)^2}{2\varepsilon_t^2}. \end{aligned} \tag{7}$$

To calculate $\widehat{g}_t$, stochastic AD is applied to the likelihood term and the prior gradient is evaluated analytically:

$$\widehat{g}_t(\theta) := \nabla \log p_0(\theta) + \widehat{\nabla} \log K_{\varepsilon_t}(\Delta(\theta, U)) \tag{8}$$

We therefore have

$$F_\varepsilon(\theta) := \mathbb{E}[\widehat{\ell}_\varepsilon(\theta, U)] = \log p_0(\theta) - \frac{\mathbb{E}[\Delta(\theta, U)^2]}{2\varepsilon^2}, \tag{9}$$

$$\mathbb{E}[\widehat{g}_t(\theta)] = \nabla F_{\varepsilon_t}(\theta). \tag{10}$$

By Jensen's inequality, $F_\varepsilon(\theta) \leq \log p_0(\theta) + \log L_\varepsilon^K(\theta)$. We differentiate the expected log-kernel rather than the logarithm of the smoothed likelihood in equation 3 because a single simulated dataset then gives an unbiased gradient. Averaging R independent replicates of the estimate in equation 8 can reduce variance while preserving unbiasedness, but we have found no significant improvements by using $R > 1$ in the experiments considered here.

2.4 The Langevin move kernel

Let $\widehat{g}_t(\theta)$ be the proposal objective gradient estimate, P_t twice the weighted empirical covariance of the particles (Beaumont et al., 2009), and $\Gamma_t = h_t^2 P_t$ with step size h_t. These quantities are fixed within a sweep. The proposal is

$$q_t(\theta' \mid \theta) = \mathcal{N}(\theta'; \mu_t(\theta), \Gamma_t), \quad \mu_t(\theta) = \theta + T_\delta\big(\tfrac{1}{2}\Gamma_t \widehat{g}_t(\theta)\big), \tag{11}$$

$$T_\delta(u) = u \cdot \min\left(1, \delta / \|u\|_{\Gamma_t^{-1}}\right), \qquad \|u\|_{\Gamma_t^{-1}} = \sqrt{u^\top \Gamma_t^{-1} u}, \tag{12}$$

Setting $T_\delta(0) = 0$, the mean displacement has Mahalanobis norm at most δ with respect to the proposal covariance Γ_t. When the truncation is inactive, this is preconditioned MALA (Roberts & Rosenthal, 1998). The truncation corresponds to the drift cap in the truncated Langevin algorithm (MALTA) of Roberts & Tweedie (1996). Setting $\widehat{g}_t(\theta) = 0$ gives the random walk baseline.

MH acceptance. A proposal $\theta' \sim q_t(\cdot \mid \theta)$ is first screened against the prior ($p_0(\theta') = 0$ rejects before simulation). Then M new datasets are simulated from $p_{\theta'}$ and accepted with probability

$$\min\left(1, \frac{p_0(\theta') G_{\varepsilon_t}(\theta', U'^{1:M}) q_t(\theta \mid \theta')}{p_0(\theta) G_{\varepsilon_t}(\theta, U^{1:M}) q_t(\theta' \mid \theta)}\right), \tag{13}$$

with the accepted datasets kept for accepted particles. If no replicate of θ' lands within ε_t the acceptance probability is zero and the drift at θ', which enters the reverse density $q_t(\theta|\theta')$, is not needed. The intractable simulator densities cancel from the ratio (Del Moral et al., 2012, Section 3.2).

Within-sweep determinism of the drift. For the Gaussian density in equation 11 to be the proposal actually evaluated in both directions, we use a fixed measurable drift within each sweep. If instead the drift were re-estimated with fresh randomness at every evaluation, the realised proposal's transition density would be an intractable mixture over the drift noise, and plugging one realisation into equation 13 yields an inexact, noisy MH chain (Alquier et al., 2016). We therefore generate all randomness used by the drift at θ in sweep t (simulator draws and stochastic AD perturbation draws, for all d coordinates) from a fixed seed calculated from t and θ (via a hash function) and store evaluations within the sweep. Within a sweep, $\widehat{g}_t$ is thus a fixed, deterministic measurable function of θ. Proposition 1 (Appendix A.2) shows that the extended target is invariant for every drift.

2.5 Unbiased gradients give better proposals

The following theorem compares the gradient proposal with a random walk at the same parameter and covariance. It gives a local guarantee of improvement in the expected proposal objective F_ε from equation 9.

Theorem 1. *Fix $\varepsilon > 0$, $\theta \in \mathbb{R}^d$, $P \succ 0$ and $\delta > 0$. Suppose $F = F_\varepsilon : \mathbb{R}^d \to \mathbb{R}$ has globally Lipschitz gradient, and $\widehat{g}$ is unbiased for $\nabla F(\theta)$ with finite second moment. Let θ_D be the proposal in equation 11 with covariance $h^2 P$ and Gaussian noise independent of $\hat{g}$ and let θ_{RW} be the random walk with the same covariance. Holding $\theta, P, \varepsilon, \delta$ fixed, then*

$$\mathbb{E}[F(\theta_D) - F(\theta_{RW})] = \frac{h^2}{2} \|\nabla F(\theta)\|_P^2 + O\left(h^3\right) \tag{14}$$

where $\|u\|_P^2 = u^T P u$. If $\nabla F(\theta) \neq 0$, the gain is strictly positive for all sufficiently small $h > 0$.

Appendix A.1 gives an explicit remainder bound and proof. Unbiasedness makes the leading term in equation 14 a squared gradient norm. Writing $g = \nabla F(\theta)$, a bias $b = \mathbb{E}[\widehat{g}] - g$ changes the leading term to be $h^2 g^T P(g + b)/2$, which is negative whenever $g^T P b < -\|g\|_P^2$. Biased estimators, such as those from continuous relaxations, therefore carry no such guarantee.

3 Experiments

3.1 Models, settings and accuracy measures

Gaussian mixture model For observations $i = 1, \cdots, n$, we generate a binary component c_i as $c_i \sim \text{Bernoulli}(\text{logit}^{-1}(\theta_0))$. Conditional on this c_i, observation $x_i \in \mathbb{R}^{d-1}$ follows $x_i | c_i, \theta \sim \mathcal{N}((c_i - \frac{1}{2})g, (1 + c_i)^2 \cdot I_{d-1})$ where $g_j = \exp(\theta_j)$ for $j = 1, \cdots, d-1$. The governing parameter is then $\theta = (\theta_0, \theta_1, \cdots, \theta_{d-1}) \in \mathbb{R}^d$ where θ_0 is the logit of the mixture weight and θ_j is the log gap for coordinate j. For each observation i, the sampled binary component c_i is shared across all $d - 1$ coordinates, inducing dependence between coordinates.

Bernoulli GLM For each trial $i = 1, \cdots, n$, we have $y_i | \theta \sim \text{Bernoulli}(p_i)$ with $p_i = \text{logit}^{-1}(x_i^T \theta) = (1 + \exp(-x_i^T \theta))^{-1}$, where $y_i \in \{0, 1\}$ and x_i is the ith row of the given design matrix X. Hence, $\theta = (\theta_0, \theta_1, \cdots, \theta_{d-1}) \in \mathbb{R}^d$, for a given model dimension d, where θ_0 is the intercept. Appendix B gives the priors, the true parameters and reference posteriors.

Multi-Population SIR model We extend the SIR task of SBIBM (Lueckmann et al., 2021) to $d - 1$ regions with their own transmission rates β_k and a shared recovery rate γ, so that $\theta = (\log \beta_1, \ldots, \log \beta_{d-1}, \log \gamma)$. In each region, a deterministic SIR epidemic in a population of 10^6 with one initial infection reaches the peak prevalence $p_k = 1 - r_k - r_k \log(s_0/r_k)$, when $r_k = \gamma/\beta_k$ is below the initial susceptible fraction s_0 (otherwise p_k is the initial prevalence). A survey of 80 individuals at the peak gives the observed count $y_k \sim \text{Binomial}(80, p_k)$.

Settings We compare ABC-D^2 with the random-walk mutation in the same adaptive SMC-ABC sampler for $N \in \{5, 10, 20, 40, 80, 160, 320, 640\}$ particles, three fixed observed datasets per dimension and 25 paired repeats per dataset that start both methods from the same prior draws. The GLM has 100 binary observations, $d \in \{5, 10, 15, 20, 25, 30\}$ and a Gaussian prior, and compares the sufficient summary $X^T Y$ by Euclidean distance, following the released SBIBM implementation (Lueckmann et al., 2021). The GMM has $d \in \{6, 8, 10, 12, 14, 16\}$ parameters and 128 observations in $\mathbb{R}^{d-1}$, compared by the maximum mean discrepancy (MMD) between the simulated and observed samples (Gretton et al., 2012). The SIR model has $d \in \{5, 10, 15, 20, 25, 30\}$ and compares the $d-1$ counts by Euclidean distance. All samplers use one MH move per particle per sweep, twice the weighted covariance (eigenvalues floored at 10^{-10}), $h = 1$, $\alpha = 0.9$, and $R = 1$. For the GLM and GMM we use $M = 5$ and $\delta = 2$ and 1 respectively, and stop at the 1.5% acceptance floor or after 120 sweeps. As single binomial counts give noisy acceptance estimates, for the SIR model we use $M = 1,000$ and $\delta = 2$ and run without an acceptance floor to the target tolerance $\varepsilon^* = q_{0.1}$, where $q_{0.1}$ is the 10% quantile of the distance of the data simulated at the true parameter, or at most 800 sweeps.

Accuracy measures We compare weighted particle means with the reference posteriors computed from the exact likelihood of the observed data, which do not depend on the ABC discrepancy: long MCMC runs for the GLM and GMM, and numerical quadrature for the SIR model (Appendix B). For reference mean μ_j and standard deviations σ_j, the error of a particle mean $\widehat{\mu}$ is $\{d^{-1} \sum_{j=1}^{d} [(\widehat{\mu_j} -$

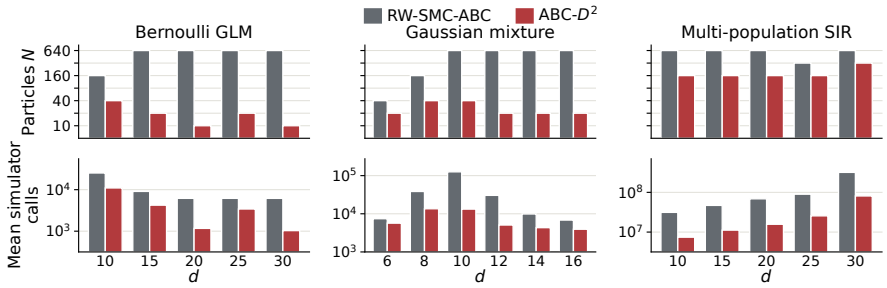

Figure 2: ABC-D^2 matches the accuracy of random walk SMC-ABC with up to 64 times fewer particles. Top: at each d, the random-walk population N (grey) and the smallest ABC-D^2 population (red) that keeps the mean final error within 10% of that of the random walk (Section 3.2), for the pair with the largest ratio. Bottom: their mean simulator calls.

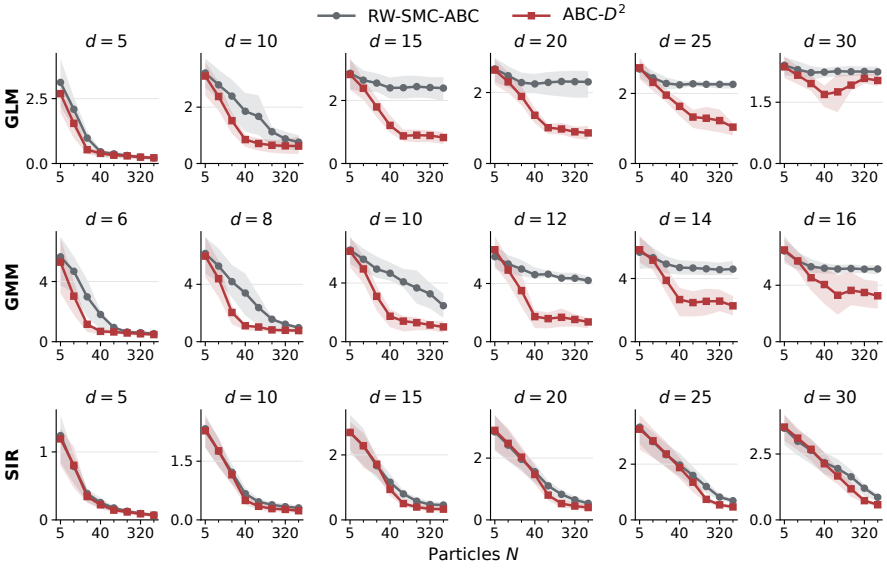

Figure 3: Posterior-mean RMSE against the particle population N for the Bernoulli GLM (top), Gaussian mixture (middle) and multi-population SIR model (bottom). Lines are means over the three datasets with 25 repeats each. Bands are the interquartile range over the 75 runs.

$\mu_j)/\sigma_j]^2\}^{1/2}$. The curves average the 75 run errors equally across the three datasets, with shading showing their interquartile range.

3.2 An order of magnitude fewer particles at equal accuracy

In most dimensions of the GLM and GMM, ABC-D^2 reaches the accuracy of the random walk with an order of magnitude fewer particles and fewer simulations (Figure 2). At fixed N, ABC-D^2 uses more simulations than the random walk, because the drift requires simulations and its runs continue to lower tolerances: at $N = 640$ it uses 25 and 7 times as many simulator calls for the 15-dimensional GLM and the 12-dimensional GMM. The relevant comparison is therefore at equal accuracy, which we declare when a one-sided simultaneous 95% upper bound places the mean error of ABC-D^2 within 10% of that of the random walk. For $d \geq 15$ in the GLM and $d \geq 10$ in the GMM, ABC-D^2 with 10 to 40 particles is as accurate as the random walk with 640, a 16 to 64 fold reduction, with 1.7 to 9.5 times fewer simulator calls (Figure 2). In the SIR model, where the random-walk error keeps decreasing with N, the reductions are smaller: four-fold for $10 \leq d \leq 20$ and two fold for $d = 25$ and 30, with 3.5 to 4.4 times fewer simulator calls.

3.3 ACCURACY ACROSS POPULATION SIZES

For the GLM (Figure 3, top), the drift helps most in intermediate and high dimensions. At $N = 640$ the mean error decreases from 2.39 to 0.82 at $d = 15$ (66%), from 2.30 to 0.86 at $d = 20$ (63%) and from 2.26 to 1.04 at $d = 25$ (54%). At $d = 25$ the random walk error changes by less than 0.06 between $N = 40$ and 640, while the ABC-D^2 with $N = 40$ is already more accurate than the random walk with $N = 640$. At $d = 10$ the reduction peaks at $N = 80$ (58%) and falls to 19% at $N = 640$ as the random walk catches up. At $d = 5$ the drift helps small populations (46% at $N = 20$), and for $N \geq 160$ the methods differ by less than 5%. At $d = 30$ the reduction peaks at $N = 40$ (24%) and falls to 9% at $N = 640$.

For the GMM (middle row), the largest mean error reductions occur at $d = 10$ and 12, from 4.08 to 1.40 at $N = 80$ (66%), and from 4.21 to 1.36 at $N = 640$ (68%), respectively. At $d = 14$ and 16 the reductions at $N = 640$ are 51% and 37%. For $d \geq 12$ the random walk error changes by at most 0.4 between $N = 40$ and $N = 640$, whereas ABC-D^2 keeps improving. At $d = 6$ and 8 the gains peak at $N = 40$ (62% and 67%) and shrink to 11% and 21% at $N = 640$, as both samplers approach the reference.

The SIR results (bottom row) follow the same pattern with smaller gains. ABC-D^2 has lower mean error, in all 30 settings with $N \geq 40$. At $N = 640$ the reduction is 6% at $d = 5$ and $24 - 33\%$ for $d \geq 10$, and the largest is 39% ($d = 30$, $N = 320$). For $N \leq 20$ and $d \geq 15$, most runs of both samplers exhaust the sweep budget and the methods differ by less than 5%. With larger populations ABC-D^2 reaches the target more often.

Why the gains diminish for large d. The terminal tolerances (Appendix C) mirror the errors: in the 30-dimensional GLM both samplers stop at similar, high tolerances (55.5 and 61.1 at $N = 640$), unlike at $d = 15$ (15.2 against 51.8). One plausible reason for this is that, with drift displacement capped at $\delta \leq 2$ proposal standard deviations, drift becomes a small part of each move compared to the Gaussian noise, which has a norm that scales as $\sqrt{d}$. This can potentially be addressed by scaling δ with d. Another plausible reason is that the growing summaries make the Euclidean distance and the MMD less informative (Blum, 2010; Ramdas et al., 2015).

4 DISCUSSION AND CONCLUSION

We introduced ABC-D^2, which brings gradient information to SMC-ABC for simulators with discrete randomness by estimating a Langevin drift with stochastic AD. Unbiasedness guarantees a local improvement of the expected proposal objective over a random walk (Theorem 1) and fixing the drift's random numbers within each sweep keeps the MH correction exact (Proposition 1). On Bernoulli GLMs, Gaussian mixtures and a multi-population SIR model, the drift reduced the posterior-mean error by up to 66%, 68% and 39%, and where the random walk stalled at high tolerances, 10 to 40 ABC-D^2 particles matched 640 random walk particles with fewer simulations.

Limitations and future work The gains shrink in the 30-dimensional GLM, where both samplers stop at high tolerances. Dimension-dependent step sizes, or tuning h_t to a target acceptance rate may help. All benchmarks use synthetic data, applying ABC-D^2 to real epidemiological data, genetic or chemical-kinetic data including Gillespie-type reaction networks is a natural next step. Another direction is to extend the application of ABC-D^2 to hierarchical models with multiple subjects having subject-specific parameters and shared population-level parameters. Following Gibbs ABC (Turner & Van Zandt, 2014), we would use block-wise rejuvenation on the vector parameters: (1) update each subject's parameter block in turn while keeping the population and other subjects' parameters fixed, and (2) update the population parameters conditional on the subject-level parameters. For each subject-level update, stochastic AD could estimate the gradient of the conditional smoothed ABC log target and use it to construct an MALA proposal that guides the update toward regions of higher conditional ABC density. We would update the population-level parameters in a separate conditional step, without using gradient information.

AI USE STATEMENT

In this work, we have not used generative AI for any of the tasks with required disclosure, including tools for research ideation or execution, to draft sections of the paper, for generating synthetic datasets, and for proving mathematical claims. We have used generative AI tools for the tasks with recommended disclosure, including for retrieval and discovery of literature on SMC-ABC, gradient estimation for discrete stochastic programs, and differentiable simulators. We also used these tools to edit and organise the source code used for numerical simulations. We have reviewed the AI-assisted code and the list of literature cited in the manuscript. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work develops a general inference method and evaluates it on synthetic data only. It involves no human subjects or personal data, and we are not aware of any specific ethical concerns.

REPRODUCIBILITY STATEMENT

Section 2 and Algorithm 1 specify the sampler, and Appendix A gives the proofs and their assumptions. Section 3 and Appendix B give the models, priors and observed data, sampler settings and reference posteriors. Code to reproduce all the experiments and figures will be made publicly available.

REFERENCES

Pierre Alquier, Nial Friel, Richard Everitt, and Aidan Boland. Noisy Monte Carlo: convergence of Markov chains with approximate transition kernels. *Statistics and Computing*, 26(1–2):29–47, 2016. doi: 10.1007/s11222-014-9521-x.

Gaurav Arya, Moritz Schauer, Frank Schäfer, and Christopher Rackauckas. Automatic differentiation of programs with discrete randomness. In *Advances in Neural Information Processing Systems*, volume 35, pp. 10435–10447, 2022. doi: 10.52202/068431-0758.

Atılım Güneş Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018. URL https://jmlr.org/papers/v18/17-468.html.

Mark A. Beaumont, Wenyang Zhang, and David J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002. doi: 10.1093/genetics/162.4.2025.

Mark A. Beaumont, Jean-Marie Cornuet, Jean-Michel Marin, and Christian P. Robert. Adaptive approximate Bayesian computation. *Biometrika*, 96(4):983–990, 2009. doi: 10.1093/biomet/asp052.

Michael G. B. Blum. Approximate Bayesian computation: a nonparametric perspective. *Journal of the American Statistical Association*, 105(491):1178–1187, 2010. doi: 10.1198/jasa.2010.tm09448.

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

Alexander Buchholz, Nicolas Chopin, and Pierre E. Jacob. Adaptive tuning of Hamiltonian Monte Carlo within sequential Monte Carlo. *Bayesian Analysis*, 16(3):745–771, 2021. doi: 10.1214/20-BA1222.

Patrick Cannon, Daniel Ward, and Sebastian M. Schmon. Investigating the impact of model misspecification in neural simulation-based inference, 2022. URL https://arxiv.org/abs/2209.01845.

Xuefei Cao, Shijia Wang, and Yongdao Zhou. An adaptive approximate Bayesian computation MCMC with global-local proposals. *arXiv preprint arXiv:2412.15644*, 2024. doi: 10.48550/arXiv.2412.15644.

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

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, 22(5):1009–1020, 2012. doi: 10.1007/s11222-011-9271-y.

Simon Duane, A. D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987. doi: 10.1016/0370-2693(87)91197-X.

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

Sarah Filippi, Chris P. Barnes, Julien Cornebise, and Michael P. H. Stumpf. On optimality of kernels for approximate Bayesian computation using sequential Monte Carlo. *Statistical Applications in Genetics and Molecular Biology*, 12(1):87–107, 2013. doi: 10.1515/sagmb-2012-0069.

David T. Frazier, Christian P. Robert, and Judith Rousseau. Model misspecification in approximate Bayesian computation: consequences and diagnostics. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(2):421–444, 2020. doi: 10.1111/rssb.12356.

Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977. doi: 10.1021/j100540a008.

Peter W. Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84, 1990. doi: 10.1145/84537.84552.

Matthew M. Graham and Amos J. Storkey. Asymptotically exact inference in differentiable generative models. *Electronic Journal of Statistics*, 11(2):5105–5164, 2017. doi: 10.1214/17-EJS1340SI.

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

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL https://www.jmlr.org/papers/v13/gretton12a.html.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. doi: 10.1093/biomet/57.1.97.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkE3y85ee.

J. F. C. Kingman. The coalescent. *Stochastic Processes and their Applications*, 13(3):235–248, 1982. doi: 10.1016/0304-4149(82)90011-4.

Genshiro Kitagawa. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *Journal of Computational and Graphical Statistics*, 5(1):1–25, 1996. doi: 10.1080/10618600.1996.10474692.

Athanasios Kousathanas, Christoph Leuenberger, Jonas Helfer, Mathieu Quinodoz, Matthieu Foll, and Daniel Wegmann. Likelihood-free inference in high-dimensional models. *Genetics*, 203(2):893–904, 2016. doi: 10.1534/genetics.116.187567.

Theodore Kypraios, Peter Neal, and Dennis Prangle. A tutorial introduction to Bayesian inference for stochastic epidemic models using approximate Bayesian computation. *Mathematical Biosciences*, 287:42–53, 2017. doi: 10.1016/j.mbs.2016.07.001.

J. Li, D. J. Nott, Y. Fan, and S. A. Sisson. Extending approximate Bayesian computation methods to high dimensions via a Gaussian copula model. *Computational Statistics & Data Analysis*, 106: 77–89, 2017. doi: 10.1016/j.csda.2016.07.005.

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

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete distribution: a continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=S1jE5L5gl.

Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100(26):15324–15328, 2003. doi: 10.1073/pnas.0306899100.

Trevelyan McKinley, Alex R. Cook, and Robert Deardon. Inference in epidemic models without likelihoods. *The International Journal of Biostatistics*, 5(1):Article 24, 2009. doi: 10.2202/1557-4679.1171.

Edward Meeds, Robert Leenders, and Max Welling. Hamiltonian ABC. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 582–591. AUAI Press, 2015. URL https://auai.org/uai2015/proceedings/papers/266.pdf.

Alexander Moreno, Tameem Adel, Edward Meeds, James M. Rehg, and Max Welling. Automatic variational abc. *arXiv preprint*, 2016. doi: https://doi.org/10.48550/arXiv.1606.08549.

Radford M. Neal. MCMC using Hamiltonian dynamics. In Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng (eds.), *Handbook of Markov Chain Monte Carlo*, pp. 113–162. Chapman and Hall/CRC, 2011. doi: 10.1201/b10905-6.

D. J. Nott, Y. Fan, L. Marshall, and S. A. Sisson. Approximate Bayesian computation and Bayes' linear analysis: toward high-dimensional ABC. *Journal of Computational and Graphical Statistics*, 23(1):65–86, 2014. doi: 10.1080/10618600.2012.751874.

George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with Bayesian conditional density estimation. In *Advances in Neural Information Processing Systems*, volume 29, pp. 1028–1036, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/6aca97005c68f1206823815f66102863-Abstract.html.

J. K. Pritchard, M. T. Seielstad, A. Pérez-Lezaun, and M. W. Feldman. Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molecular Biology and Evolution*, 16 (12):1791–1798, 1999. doi: 10.1093/oxfordjournals.molbev.a026091.

Aaditya Ramdas, Sashank Jakkam Reddi, Barnabás Póczos, Aarti Singh, and Larry Wasserman. On the decreasing power of kernel and distance based nonparametric hypothesis tests in high dimensions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015. doi: 10.1609/aaai.v29i1.9692.

G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120, 1997. doi: 10.1214/aoap/1034625254.

Gareth O. Roberts and Jeffrey S. Rosenthal. Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society: Series B*, 60(1):255–268, 1998. doi: 10.1111/1467-9868.00123.

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.

Conor Rosato, Joshua Murphy, Alessandro Varsi, Paul Horridge, and Simon Maskell. Enhanced SMC^2: Leveraging gradient information from differentiable particle filters within Langevin proposals. In *2024 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 1–8. IEEE, 2024. doi: 10.1109/MFI62651.2024.10705779.

S. A. Sisson, Y. Fan, and Mark M. Tanaka. Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 104(6):1760–1765, 2007. doi: 10.1073/pnas.0607208104.

Scott A. Sisson, Yanan Fan, and Mark A. Beaumont (eds.). *Handbook of Approximate Bayesian Computation*. Chapman and Hall/CRC, Boca Raton, FL, 2018. ISBN 978-1-315-11719-5. doi: 10.1201/9781315117195.

Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael P. H. Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009. doi: 10.1098/rsif.2008.0172.

Brandon M. Turner and Trisha Van Zandt. Hierarchical approximate bayesian computation. *Psychometrika*, 79(2):185–209, 2014. doi: 10.1007/s11336-013-9381-x.

Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-normalization, folding, and localization: an improved $\widehat{R}$ for assessing convergence of MCMC (with discussion). *Bayesian Analysis*, 16(2):667–718, 2021. doi: 10.1214/20-BA1221.

Xiaoyu Wang, Ryan P. Kelly, Adrianne L. Jenner, David J. Warne, and Christopher Drovandi. A comprehensive guide to simulation-based inference in computational biology, 2024. URL `https://arxiv.org/abs/2409.19675`.

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

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992. doi: 10.1007/BF00992696.

A Algorithms and Proofs

Algorithm 1 gives one sweep of ABC-D^2 in full. Each particle stores the M distances of its current datasets, simulated at initialisation or when its last move was accepted, so reweighting, tolerance selection and resampling need no new simulations. With $M_g > 1$ datasets per drift replicate, the objective in equation 7 is replaced by the logarithm of the average of the M_g kernels. All our experiments use $M_g = R = 1$ however, as increasing the number of replicates is costly and we have found no significant gain for the drift calculation.

Algorithm 1 One sweep of the adaptive SMC-ABC sampler of Del Moral et al. (2012) with the $ABC - D^2$ move.

Input: particles $\{(\theta^{(i)}, \Delta^{(i),1:M}, W^{(i)})\}_{i=1}^{N}$, previous ε_{t-1}, ESS factor α, target $\varepsilon^\star$, acceptor replicates M, drift replicates R with M_g datasets each, step h, truncation δ.
Reweight: stage ε_t by bisection on equation 5 over $[\varepsilon^\star, \varepsilon_{t-1}]$, clamped at the target; update $W^{(i)}$ by equation 4.
Resample systematically if ESS $< N/2$; reset $W^{(i)} = 1/N$.
Fit covariance $P_t = 2\widehat{\Sigma}_t$; set $\Gamma_t = h^2 P_t$; fix the drift for sweep t at tolerance ε_t
for each particle i with $W^{(i)} > 0$ **do**
  for each replicate $r = 1, \dots, R$ **do**
    Draw M_g seeds $U_{r,1:M_g}$ from $(t, r, \theta^{(i)})$
    For each coordinate $v = 1, \dots, d$, replay $U_{r,1:M_g}$ and evaluate equation 6 on equation 7
  end for
  $\widehat{g}_t(\theta^{(i)}) \leftarrow R^{-1} \sum_r \widehat{\nabla \ell}_{M_g,r}$; store for reuse
  $\mu \leftarrow \theta^{(i)} + T_\delta\big(\frac{1}{2}\Gamma_t \widehat{g}_t(\theta^{(i)})\big)$ by equation 12; draw $\theta' \sim \mathcal{N}(\mu, \Gamma_t)$
  if $p_0(\theta') = 0$, reject; **else** simulate M fresh datasets at θ'; **if** $G_{\varepsilon_t}(\theta', \cdot) = 0$, reject
  else accept $(\theta', U'^{1:M})$ with the MH probability equation 13, the reverse density using the (stored, deterministic) drift at θ'
end for
Stop the run if $\varepsilon_t = \varepsilon^\star$, the sweep's MH acceptance rate is below 1.5%, or the sweep cap is reached; **else** proceed to sweep $t + 1$.

A.1 Proof of Theorem 1

Explicit remainder bound. Under the assumptions of Theorem 1, write $g = \nabla F(\theta), a = \|g\|_P$ and $m_2 = \mathbb{E}\,\|\widehat{g}\|_P^2 < \infty$. Since P is fixed and positive definite, global Lipschitz continuity of the gradient gives a finite L such that, for all $x, y \in \mathbb{R}^d$,

$$\|\nabla F(x) - \nabla F(y)\|_P \le L\,\|x - y\|_{P^{-1}}\,. \tag{15}$$

For $h > 0$, let $\Gamma = h^2 P$ and $Z \sim \mathcal{N}(0, I_d)$, independent of $\widehat{g}$:

$$\theta_D = \theta + T_\delta(\tfrac{1}{2}h^2 P\widehat{g}) + hP^{1/2}Z, \quad \theta_{RW} = \theta + hP^{1/2}Z, \tag{16}$$

where T_δ uses the Γ^{-1} norm in equation 12. The expected proposal guarantee satisfies for every $h > 0$,

$$\left|\mathbb{E}[F(\theta_D) - F(\theta_{RW})] - \frac{h^2 a^2}{2}\right| \le h^3\left(\frac{a m_2}{16\delta} + \frac{L}{2}\sqrt{d m_2}\right) + \frac{L h^4 m_2}{8} \tag{17}$$

Proof. Use the symmetric positive-definite square root of P and write

$$f(x) = F(\theta + P^{1/2}x), \quad X = P^{1/2}\widehat{g}, \quad \mu = P^{1/2}g. \tag{18}$$

Then $\nabla f(0) = \mu, \mathbb{E}[X] = \mu$, $\|\mu\| = a$, and $\mathbb{E}\|X\|^2 = m_2$. The Lipschitz continuity of ∇F makes ∇f also globally L-Lipschitz, and so the expectations considered are finite.

The Γ^{-1} norm of the uncapped mean displacement is $h\,\|X\|/2$. Hence the displacement in P normalised coordinates is

$$c_h := P^{-1/2} T_\delta(\tfrac{1}{2}h^2 P\hat{g}) = \frac{h^2}{2} X \min\left(1, \frac{2\delta}{h\,\|X\|}\right), \tag{19}$$

with $c_h = 0$ for $X = 0$. Thus, $F(\theta_D) = f(hZ + c_h)$ and $F(\theta_{RW}) = f(hZ)$.

For $r \geq 0$ and $b > 0$, $(r - b)_+ \leq r^2/(4b)$. When $r \geq b$ this follows from $(r - 2b)^2 \geq 0$, and otherwise the left side is zero. Applying it with $b = 2\delta/h$ gives

$$\mathbb{E}\left\|c_h - \frac{h^2}{2}X\right\| = \frac{h^2}{2}\mathbb{E}\left(\|X\| - \frac{2\delta}{h}\right)_+ \leq \frac{h^3 m_2}{16\delta}, \tag{20}$$

$$\mathbb{E}\|c_h\|^2 \leq \frac{h^4}{4}\mathbb{E}\|X\|^2 = \frac{h^4 m_2}{4}. \tag{21}$$

Unbiasedness now gives the leading term:

$$\left|\mu^T\mathbb{E}[c_h] - \frac{h^2 a^2}{2}\right| = \left|\mu^T\mathbb{E}\left[c_h - \frac{h^2}{2}X\right]\right| \leq \frac{h^3 a m_2}{16\delta} \tag{22}$$

By the Lipschitz bound,

$$\left|f(hZ + c_h) - f(hZ) - \mu^T c_h\right| = \left|\int_0^1 [\nabla f(hZ + sc_h) - \nabla f(0)]^T c_h ds\right| \tag{23}$$

$$\leq Lh\|Z\|\|c_h\| + \frac{L}{2}\|c_h\|^2. \tag{24}$$

Using $\mathbb{E}\|Z\|^2 = d$, the Cauchy-Schwartz inequality, and equation 21, we obtain

$$\mathbb{E}\left|f(hZ + c_h) - f(hZ) - \mu^T c_h\right| \leq \frac{Lh^3}{2}\sqrt{dm_2} + \frac{Lh^4 m_2}{8} \tag{25}$$

Since $\mathbb{E}[F(\theta_D) - F(\theta_{RW})] = \mu^T\mathbb{E}[c_h] + \mathbb{E}[f(hZ + c_h) - f(hZ) - \mu^T c_h]$, combining this inequality with equation 22 proves equation 17. The right-hand side of equation 17 is $O(h^3)$, which gives equation 14. Dividing equation 17 by h^2 shows that the gain is strictly positive for sufficiently small h when $a > 0$. □

Where bias enters If instead the gradient estimate has bias $b = \mathbb{E}[\widehat{g}] - g \neq 0$, then equation 22 changes. Its leading term becomes $\frac{h^2}{2}g^T P(g + b)$, so

$$\mathbb{E}[F(\theta_D) - F(\theta_{RW})] = \frac{h^2}{2}g^T P(g + b) + O(h^3). \tag{26}$$

Thus the leading order proposal improvement can be negative, meaning that a biased estimator of the gradient does not guarantee improvement.

A.2 Invariance of the Fixed Drift Mutation

Proposition 1. *Let $\varepsilon > 0, \Gamma \succ 0, \delta > 0$ and let $b : \Theta \to \mathbb{R}^d$ be a finite measurable drift. Let q_b be the Gaussian proposal in equation 11, with $\widehat{g}_t = b$ and covariance Γ. Consider the proposal $\theta' \sim q_b(\cdot|\theta)$ and its simulated datasets $Y'^{1:M}$ sampled independently from $p_{\theta'}$. Accept using equation 13, evaluating both Gaussian densities with this same drift and covariance. The resulting kernel is reversible with respect to $\widetilde{\pi}_\varepsilon$, whose θ marginal is the ABC posterior π_ε.*

Proof. Since b is finite and measurable and T_δ is continuous, the mean $\theta + T_\delta(\Gamma b(\theta)/2)$ is finite and measurable. Together with $\Gamma \succ 0$, this defines an evaluable Gaussian proposal q_b that is fixed within the sweep. In our construction, deterministic seeding ensure that proposal generation and both density evaluations use the same drift.

Now apply the extended target ABC MH construction of Del Moral et al. (2012, Section 3.2) with parameter proposal q_b and M sampled replicates from the simulator. Their simulator density cancellation gives exactly equation 13. Retaining the state on rejection therefore gives reversibility by the standard MH result (Hastings, 1970). The θ marginal is π_ε because $\mathbb{E}[G_\varepsilon|\theta] = L_\varepsilon(\theta)$, as in Del Moral et al. (2012, Section 2.2). □

B EXPERIMENTAL DETAILS

Priors and observed data For the GLM, the design matrix has an intercept column and lags $0, \dots, d-2$ of one fixed white-noise stimulus, as in SBIBM, and the prior is the SBIBM smoothness prior extended to d dimensions: $\theta_0 \sim \mathcal{N}(0, 2)$ independently of $(\theta_1, \dots, \theta_{d-1}) \sim \mathcal{N}(0, (A^T A)^{-1})$, where $A = B^2 + \mathrm{diag}\{(k/(d-1))^{1/2}\}_{k=0}^{d-2}$ and B is the $(d-1) \times (d-1)$ first difference matrix. Each observed GLM dataset draws its true parameter from this prior. For the GMM, the priors are independent uniforms, $\theta_0 \sim \mathcal{U}(-2.5, 2.5)$ and $\theta_j \sim \mathcal{U}(\log 0.5, \log 5)$. Both samplers run in logistic coordinates on this box, and errors are computed in the original coordinates. The observed GMM datasets use mixture weight 0.25 and gaps g_j evenly spaced from 1 to 3 across $d-1$ coordinates. For the SIR model, the priors are the SBIBM priors, $\log(\beta_k) \sim \mathcal{N}(\log 0.4, 0.5^2)$ and $\log \gamma \sim \mathcal{N}(\log 0.125, 0.2^2)$, independently. The true β_1 and γ are those of the first three SBIBM observations, the other β_k are drawn from the prior, and the observed counts are new binomial surveys. Errors are computed for rates β_k and γ.

Reference posteriors. For the GLM and GMM, four preconditioned MALA chains on the exact likelihood are extended in blocks of $50,000$ iterations until, for every coordinate, its square and the squared norm, the rank-normalised and folded split-$\hat{R}$ (Vehtari et al., 2021) is at most 1.01 and the bulk and tail effective sample sizes are at least $10,000$ and $5,000$. Two of the four chains start from a Laplace approximation at the posterior mode and two from the prior. For the SIR model, the posterior factorises over regions given γ, so the reference moments are computed by numerical quadrature over γ and conditionally on γ, over each β_k.

C TERMINAL TOLERANCES

Figures 4 to 6 show the tolerance ε_t at which each run stops, with the same aggregation as Figure 3.

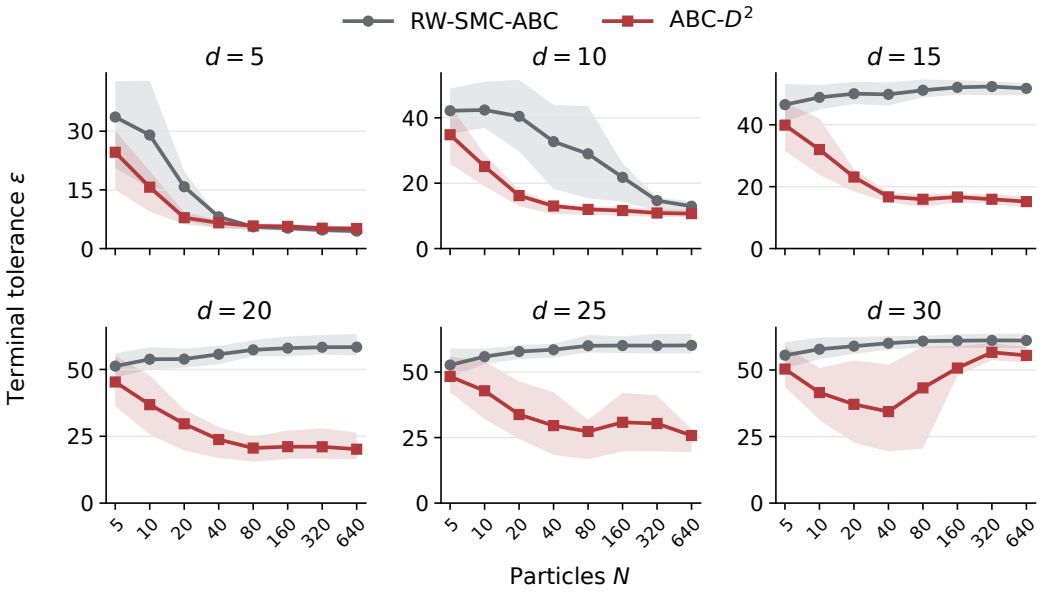

Figure 4: Bernoulli GLM: terminal tolerance ε at which each run stops, across particle populations and parameter dimensions. Lines are means over three fixed datasets with 25 repeats each, with bands showing interquartile ranges.

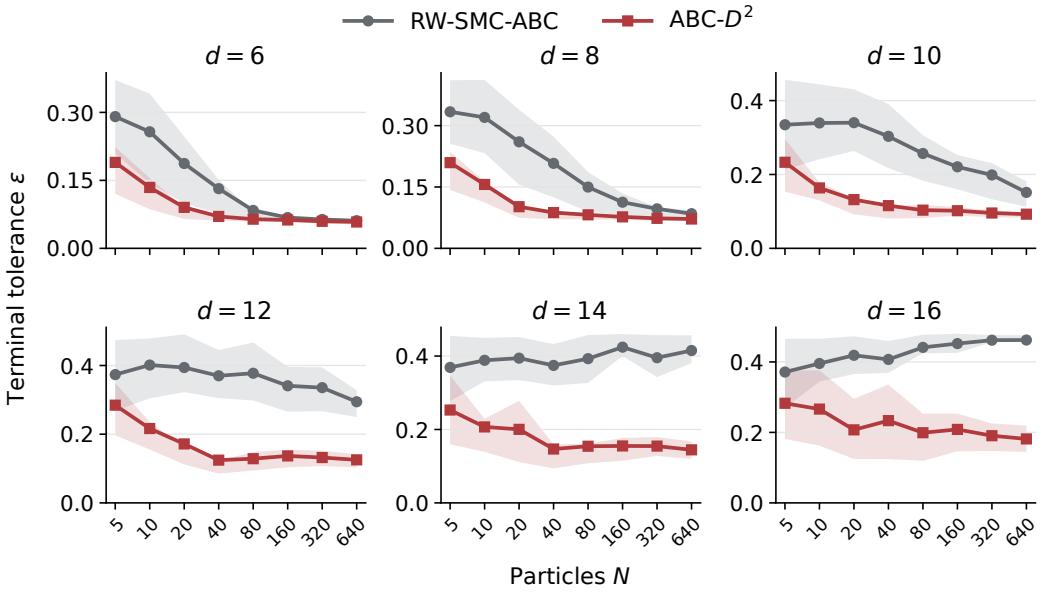

Figure 5: Gaussian mixture: terminal tolerance ε (MMD scale) with the same aggregation and shading as Figure 4.

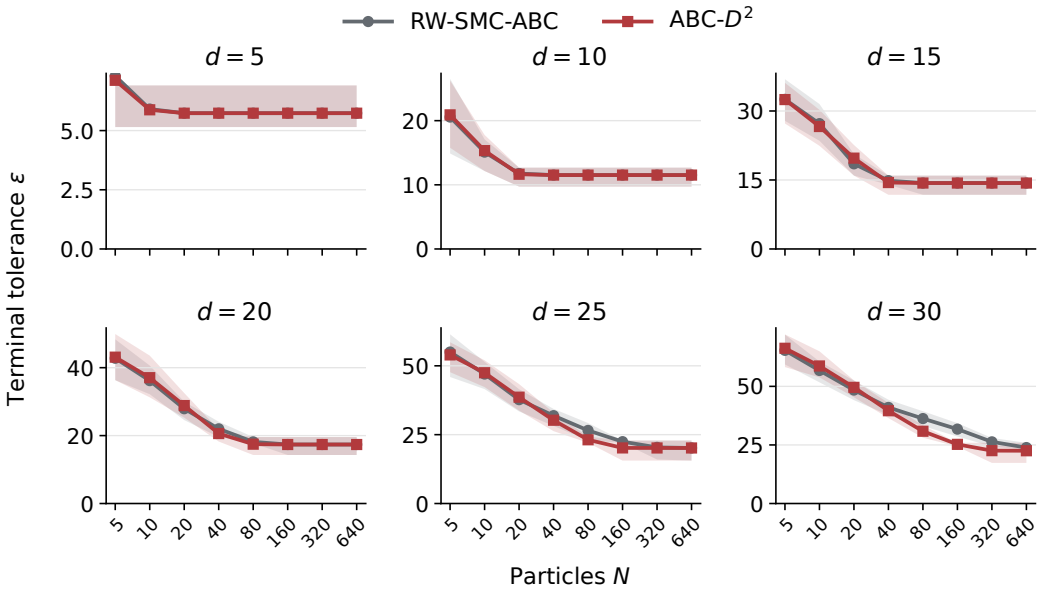

Figure 6: Multi-Population SIR: terminal tolerance ε with the same aggregation and shading as Figure 4.