SIAM/ASA J. Uncertainty Quantification
Vol. 14, No. 2, pp. 661–690

FLARE MCMC: Fidelity-Based Layer-Adaptive Recursive Proposals for MCMC*

Harini Venkatesan†, Christian Shelton†, Ming-Feng Ho‡, Simeon Bird§, and Mengxuan Wu†

Abstract. Markov chain Monte Carlo (MCMC) requires only the ability to evaluate the likelihood, making it a common technique for inference in complex models. However, it can have a slow mixing rate, requiring the generation of many samples to obtain good estimates and an overall high computational cost. FLARE MCMC is a multifidelity layered MCMC method that exploits lower-fidelity approximations of the true likelihood calculation to improve mixing and leads to overall faster performance. Such lower-fidelity likelihoods are commonly available in scientific and engineering applications where the model involves a simulation whose resolution or accuracy can be tuned. Our technique uses recursive, layered chains with simple layer tuning; it does not require the likelihood to take any specific form or have any particular internal mathematical structure. We demonstrate experimentally that FLARE MCMC achieves larger effective sample sizes for the same computational time across different scientific domains, including hydrology and cosmology.

Key words. Markov chain Monte Carlo, multilayered models, Bayesian inference, simulation-based inference, hydrology, cosmology

MSC codes. 62F15, 62M05, 65C05, 65C40, 85A35

DOI. 10.1137/25M1795194

1. Introduction.

Markov chain Monte Carlo (MCMC) is a workhorse of scientific and engineering computation. Most frequently, it is employed to compute the posterior distribution of model parameters based on observations. The calculated distributions (as represented by samples) give estimates that can be used in calibration and uncertainty quantification to aid in the generation of new scientific experiments, clarify the observability of the model, and resolve scientific theories.

Among the many MCMC algorithms, Metropolis–Hastings MCMC (MH-MCMC) is popular because of its ability to sample from almost any distribution while requiring only the ability to evaluate the model's likelihood given a parameter setting. Yet, this is also its weakness, as it has no additional knowledge of the problem setting to guide its sampling effectively. Therefore, its mixing time (speed of generating effective new samples) can be slow and the overall algorithm computationally burdensome.

*Received by the editors September 4, 2025; accepted for publication (in revised form) December 5, 2025; published electronically June 15, 2026.
https://doi.org/10.1137/25M1795194
Funding: The work of the authors was supported by National Science Foundation (NSF) grant IIS-2435579. The fourth author was supported by NASA-80NSSC21K1840.

†Department of Computer Science and Engineering, University of California, Riverside, Riverside, CA 92521 USA (hvenk001@ucr.edu, cshelton@cs.ucr.edu, mwu171@ucr.edu).
‡Leinweber Center for Theoretical Physics, University of Michigan, Ann Arbor, MI 48109 USA (mfho@umich.edu).
§Department of Physics and Astronomy, University of California, Riverside, Riverside, CA 92521 USA (sbird@ucr.edu).

Methods such as Hamiltonian Monte Carlo and its variants [17, 46, 36] speed up mixing by adding auxiliary momentum variables, allowing longer steps to reduce correlations between consecutive samples. Such methods require computing the gradient of the log target distribution with respect to the parameters—something that could be prohibitively expensive when the distribution is evaluated through lengthy simulation code. For instance, the cosmological simulation we use in our experimental results that aims to approximate the posterior density conditioned on the galaxy power spectrum from SDSS-III Baryon Oscillation Spectroscopic Survey (BOSS) Data [13, 4] cannot be modified to produce gradients. Due to the complexity and nondifferentiability of the forward cosmological simulation, gradients with respect to the model parameters are not available. Therefore, methods like autodifferentiation cannot be applied, and there is no analytic form for the gradients, prohibiting the use of gradient-based inference methods.

Fidelity-based layer-adaptive recursive (FLARE) MCMC speeds up the mixing time of MH-MCMC by exploiting lower-fidelity models of the same problem. Many engineering or scientific computational models can be run at multiple fidelities. FLARE MCMC exploits a set of computationally cheaper posterior calculations, each an approximation of the true posterior. Many posteriors involve solving a PDE, ODE, or integral. For these, coarsening the spatial or temporal grid leads to cheaper approximations. For those with constraint or optimization solvers, reducing the solvers' tolerances or maximum number of iterations can similarly lead to cheaper approximations. We further show a physics example where the underlying simulation can be coarsened by reducing the number of representative particles.

By recursively employing MCMC chains, we can use the coarser resolution models to guide the higher resolution MCMC chain. The result is a sampler for the target model that converges faster and generates more effective samples per computation time—even when considering the extra time necessary to employ the lower-fidelity computations.

2. Background.

Markov chain Monte Carlo is a class of algorithms designed to sample from a complicated target distribution by constructing an easy-to-simulate Markov chain such that the stationary distribution of the Markov chain is the target distribution. Commonly, this target distribution is the posterior distribution of a set of parameters, conditioned on observations. Let D be the observations and $\theta \in \Theta \subset \mathbb{R}^R$ be the parameters. Assuming a prior distribution on the parameters $p(\theta)$, the target posterior distribution of interest, $\pi(\theta \mid D)$, is obtained through Bayes' theorem,

$$\pi(\theta \mid D) = \frac{\mathcal{L}(D \mid \theta) p(\theta)}{p(D)} \propto \mathcal{L}(D \mid \theta) p(\theta), \tag{2.1}$$

where $\mathcal{L}(D \mid \theta)$ is the likelihood of the data, which in many scientific applications requires a lengthy simulation to evaluate. We only require the ability to evaluate $\pi(\theta \mid D)$ up to a constant of proportionality, and therefore the denominator of $p(D)$ is safely ignored. That $\pi(\theta \mid D)$ is a conditional distribution is largely irrelevant for MCMC, so we will just let $\pi(\theta)$ denote the distribution of interest (which is equal to $\pi(\theta \mid D)$ if the underlying distribution is a posterior, but it could be any distribution over θ).

2.1. Metropolis–Hastings MCMC.

We begin by focusing on the Metropolis–Hastings method for Markov Chain Monte Carlo (MH-MCMC) introduced by Hastings (1970) [30].

The $(i+1)$th sample, θ^{i+1}, is generated based on the previous sample in the chain, θ^i, in a two-step process. First, a proposed next state, $\tilde{\theta}^i$ is generated from a proposal distribution, $q(\tilde{\theta}^i|\theta^i)$. Then, $\tilde{\theta}^i$ is either accepted or rejected as θ^{i+1} according to a carefully constructed acceptance probability. If accepted, $\theta^{i+1}=\tilde{\theta}^i$; otherwise, $\theta^{i+1}=\theta^i$. Often, a normal distribution centered at θ^i is used as the proposal distribution $q(\tilde{\theta}^i|\theta^i)$, but almost any proposal distribution can be used, subject to mild conditions (for instance, that $q(\tilde{\theta}^i|\theta^i)$ is positive everywhere). With a chosen $q(\tilde{\theta}^i|\theta^i)$, the acceptance probability, $\mathcal{A}$, for the transition $\theta^i \to \tilde{\theta}^i$ is

$$\mathcal{A}(\theta^i \to \tilde{\theta}^i) = \min(1, r(\theta^i \to \tilde{\theta}^i)) \tag{2.2}$$

where

$$r(\theta^i \to \tilde{\theta}^i) = \frac{\pi(\tilde{\theta}^i)}{\pi(\theta^i)} \frac{q(\theta^i|\tilde{\theta}^i)}{q(\tilde{\theta}^i|\theta^i)} \,. \tag{2.3}$$

Although the standard MH–MCMC algorithm can be an easy way to sample from a posterior distribution, it requires sufficient samples to be an effective approximation of the posterior distribution. When the chain is slow to mix (due to a less-than-optimal proposal distribution), consecutive samples are highly dependent, and more samples must be taken to achieve a set representative of the true distribution. When the evaluation of $\pi(\theta^i)$ (necessary for the calculation of equation (2.3)) is computationally expensive, this is particularly problematic.

2.2. Related work. Our goal of accelerating MCMC sampling is shared by a large body of work. These approaches involve methods that couple chains (like simulated tempering), methods that aim to reduce the variance of estimators for a target using cheap approximations from multiple fidelities (like MLMC), and methods that use cheap models to build MCMC proposals.

Like FLARE MCMC, methods such as simulated tempering and coupled MCMC [56, 45, 5] use multiple chains. Samples are accepted or rejected by evaluating the energy of the process and adjusting the temperature of the model. Two chains are run in parallel at different temperatures, and the system swaps between different temperatures. Reversible jump MCMC [23, 3] also jumps between chains (of different dimensions). Although FLARE MCMC shares the notion of multiple chains, because it solves a different problem (to take advantage of simulations that are orders of magnitude cheaper to evaluate), the resulting structure is very different. Methods such as sequential MCMC or particle filtering [42, 16] use the notion of approximations of the target by a large number of samples called particles that are propagated across time using importance sampling. However, those are filtering frameworks and do not converge to a stationary distribution. Thus, though appearing related in its structure, FLARE MCMC is quite different from these methods.

A highly influential body of work focuses on reducing the variance of the final estimator for a target expectation in the multilevel Monte Carlo (MLMC) framework. Taking inspiration from the MLMC method [31] for high-dimensional, parameter-dependent integrals and MLMC path simulation [22], Hoang, Schwab, and Stuart [34] proposed a multilevel MCMC method that applies to Bayesian Inverse problems.

The core idea is to decompose a high-fidelity expectation into a telescoping sum using a hierarchy of computational models with increasing model resolution. This method achieves

computational speedup by estimating the low variance difference terms with a small number of samples, while the bulk of the computational effort is spent on the cheap, low-fidelity estimator. The authors also provide a rigorous complexity proof, showing how quickly posterior expectation might converge when running iterative samplers on sparse grids using telescopic expansion on the discretization error.

This MLMC framework has been extended in many directions. Multilevel sequential Monte Carlo samplers [7] and multilevel particle filters [39], along with previous work [35, 26, 25], extended MLMC to sequential Monte Carlo. Jasra et al. [38] extended MLMC to the problem of static parameter estimation in partially observed diffusions. Problems with multiple ways of discretizing were addressed by multi-index MCMC [29, 40]. These MLMC methods use samples from all chains in a telescoping estimator. They target the mean squared error (MSE) of a specific quantity of interest. In contrast, FLARE MCMC uses only samples from the finest chain (like the methods discussed below) and targets the chain's mixing time rather than MSE. Our theoretical analyses in this paper focus on the ergodicity and convergence rates for FLARE MCMC and are not specific to any particular problem domain.

Similarly to FLARE MCMC, several previous methods have shown that replacing the proposal with an approximation with generally high acceptance probability reduces the computational cost of the standard Metropolis–Hastings algorithm significantly. This idea was first proposed by Christen and Fox [9, 20] as a two-stage MCMC method that tests the original proposal using a cheap approximation to find moves in the chain that are more likely to be accepted. In other words, a candidate is accepted with the likelihood of the approximate model before it is evaluated with the more expensive model. In preconditioned MCMC using coarse-scale simulation proposed by Efendiev, Hou, and Luo [18], two stages are used to reduce the computational cost incurred in the fine fidelity by testing the coarse model based on a high-fidelity multiscale finite volume model. However, this only performs a single check with a cheap approximation and does not exploit it to run a full MCMC subchain.

Multilevel Markov chain Monte Carlo (MLMCMC) [14] achieves computational efficiency on the finer levels. If the coarse proposal from the approximation is rejected by the fine level, the coarse chain continues independently of the fine chain instead of recursively starting the next coarse chain from the current sample of the fine chain. MLMCMC uses a user-specified variable that is internal to the likelihood computation and shared across the levels (for instance, the predicted observations to be compared with the true observations through a noise model). The samples drawn from the coarse approximation are used to reduce the variance of this internal variable, achieving better proposals from the coarse fidelities.

Lykkegaard et al. [43] proposed adaptive multilevel delayed acceptance (MLDA), which adapted a recursive version of MLMCMC over multiple levels. Here, the coarse inner subchain used to generate subsequent proposals for the current chain is initiated from the current sample of the outer chain again instead of independently continuing the fine chain even if coarse proposal is rejected. MLDA also applies an adaptive error model (AEM) [41] to account for discrepancies between the different fidelities. It takes the two-level AEM from adaptive delayed acceptance Metropolis–Hastings [12, 11] and extends it by adding a telescoping sum of differences in the model output across multiple levels.

Several multilevel MCMC methods based on delayed rejection, in contrast to delayed acceptance, have also been proposed and are summarized by Peherstorfer, Willcox, and

Gunzburger [48]. Adaptive methods in multistage MCMC [57] have proposed using an independence sampler that is a good approximation for the posterior distribution in the first stage and random walk in the second stage to help with poor approximation by the independence sampler. Delayed rejection in MCMC [24] suggested using a normal distribution as the proposal in the first level and a normal distribution with the same mean but higher variance in the second level. Higdon, Lee, and Bi [32] proposed using multiple MCMC chains from low and high fidelities and coupling them using a product chain and "swapping" updates, allowing information to move between the two fidelity scales. An accelerated MCMC method using local approximations was developed by Conrad et al. [10] that uses local approximations of either the log-likelihood function or the forward model of different simulations in the Metropolis–Hastings kernel. Although these methods use approximations as proposals, they do not exploit layered or recursive MCMC chains.

Cai and Adams [8] proposed a multifidelity Monte Carlo method (MFMC) that uses randomized fidelities as the approximation for the target fidelity. The algorithm does not converge to the true posterior, but the resulting samples can be used to estimate expectations through a specific "sign-correction" formula. Our method follows a hierarchy of levels in its sampling while also sampling from the true posterior and provides a simpler alternative to previous multilevel methods.

Our multifidelity layered MCMC algorithm, FLARE MCMC, has a similar structure to MLDA in terms of the recursive layers and achieves a similar amount of effective samples across multiple chains of MLDA. However, our method for mitigating the differences between approximations is simpler in construction and implementation than that of MLDA, does not require the identification of any internal variables of the distribution to be sampled, and generates more effective samples in a shorter amount of time and with less computational cost. We demonstrate this on real-world large scientific problems. We also show theoretical convergence rates, give the optimal value for the number of inner steps M, and prove ergodicity of the adaptation in layer tuning.

3. FLARE MCMC: Fidelity-based layer-adaptive recursive proposals for MCMC.

We consider a series of models ordered by fidelity. For instance, we might have a model that evaluates a differential equation numerically as the main part of the likelihood calculation (simulating forward in time); the resolution of the spatial or temporal grid used to evaluate the model can be tuned to change its fidelity. The highest fidelity model is our "true" model, from whose posterior we wish to sample. FLARE MCMC draws samples from the true model. Its nested chains use the coarser fidelity models as cheap approximations of this finest fidelity model to speed up mixing.

Where the standard Metropolis–Hastings algorithm uses a distribution q that proposes the next sample, FLARE MCMC uses nested Markov chains as the proposal distribution. In a recursive fashion, each layer uses the result of another MCMC chain, with a coarser approximation as its proposal. The recent sample in the current chain is the starting sample in the nested chain. The coarser chain runs for M iterations, with each proposed sample evaluated by the likelihood of the cheaper layer. The last, Mth, sample of the coarser chain is proposed as the candidate for the next sample in the current chain. At the coarsest fidelity/layer, a standard proposal distribution is used, for instance, a normal distribution centered on the current point.

This avoids numerous expensive likelihood calculations in the fine fidelity that might end up rejected, and it allows the proposal to generate samples that are more likely to be accepted by the finest fidelity, since they were already accepted by an approximation. While the coarser chains have their own computational cost, they can often be orders of magnitude faster to evaluate, thus leading to an overall savings in the running time of the entire algorithm, as measured by the quality of the samples generated per computational time.

3.1. Algorithm specification. Let $\theta \in \mathbb{R}^R$ be the set of parameters (over which we are sampling), and let $j \in \{0, 1, \ldots, J\}$ be the fidelities ordered in a decreasingly complex fashion (0 is the "true" model, and J is the coarsest fidelity). We let $\pi_j(\theta)$ be the posterior distribution according to the jth fidelity model and let $q_j(\tilde{\theta}|\theta)$ be the proposal distribution for layer j. Here, θ_j^i is the ith sample in the current chain at layer j. The goal is to sample from $\pi_0(\theta)$.

In FLARE MCMC, the proposal distribution $q_j(\tilde{\theta}_j^i|\theta_j^i)$ for iteration i of a chain at layer j is another MCMC chain of M steps targeting the (coarser) posterior $\pi_{j+1}(\cdot)$, starting this nested chain at θ_j^i. The result of M steps using a chain with stationary distribution $\pi_{j+1}(\cdot)$ is the proposal for $\tilde{\theta}_j^i$: $q_j(\tilde{\theta}_j^i|\theta_j^i)$. More algorithmically, to generate $\tilde{\theta}_j^i$ from θ_j^i, we run the (coarser) MCMC algorithm at layer $j+1$. We start with $\theta_{j+1}^0 = \theta_j^i$ and continue the coarser MCMC sampler until θ_{j+1}^M. We then set $\tilde{\theta}_j^i = \theta_{j+1}^M$. At the coarsest layer, $q_J(\tilde{\theta}^i|\theta^i)$ is a standard simple proposal distribution. Figure 1 pictorially demonstrates this for $J = 2$ inner layers, each with $M = 2$ steps.

With the sampling scheme so defined, it remains to construct the acceptance probability for each layer: $\mathcal{A}_0, \mathcal{A}_1, \ldots, \mathcal{A}_J$. We follow a standard Metropolis–Hastings method for every layer, and therefore $\mathcal{A}_j = \min(1, r_j(\theta_j^i \to \tilde{\theta}_j^i))$. At the coarsest layer, the ratio $r_J(\theta_J^i \to \tilde{\theta}_J^i)$ is just as in equation (2.3) because q_J is a standard proposal distribution.

When $j < J$, the proposal distribution is from a Markov chain that obeys detailed balance. Therefore

$$\frac{q_{j+1}(\theta_j^i|\tilde{\theta}_j^i)}{q_{j+1}(\tilde{\theta}_j^i|\theta_j^i)} = \frac{\pi_{j+1}(\theta_j^i)}{\pi_{j+1}(\tilde{\theta}_j^i)}, \qquad 0 \le j < J, \tag{3.1}$$

Algorithm 3.1. FLARE-Chain(θ_j^0, n, j).

for $i = 0, \ldots, n-1$ **do**
  if $j = J$ **then** ▷ coarsest layer
    Sample $\tilde{\theta}_j^i$ from $q_j(\cdot|\theta_j^i)$
    Accept $\theta_j^{i+1} = \tilde{\theta}_j^i$ with probability $\mathcal{A}$ from equation (2.2)
    Otherwise, reject and $\theta_j^{i+1} = \theta_j^i$
  else
    $\theta_{j+1}^1, \ldots, \theta_{j+1}^M =$ FLARE-Chain$(\theta_j^i, M, j+1)$
    $\tilde{\theta}_j^i = \theta_{j+1}^M$
    Accept $\theta_j^{i+1} = \tilde{\theta}_j^i$ with probability $\mathcal{A}_j$ from equation (3.2)
    Otherwise, reject and $\theta_j^{i+1} = \theta_j^i$
return $\theta_j^1, \ldots, \theta_j^n$

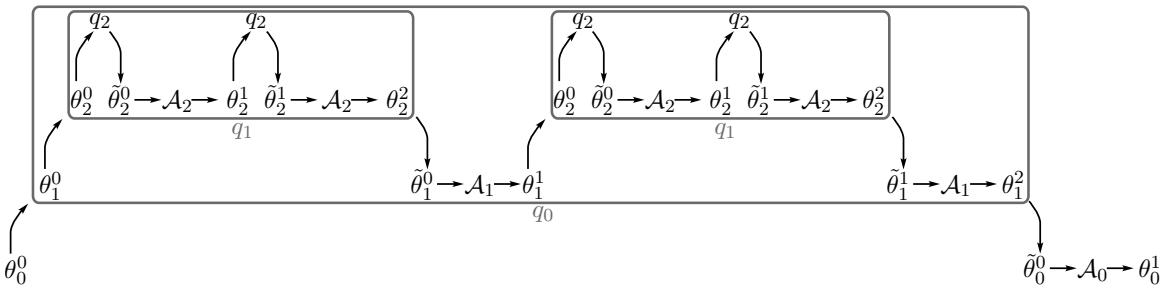


Figure 1. *One sampling step from the finest layer with two coarse fidelities and two iterations per nested chain. See the text of section 3.1.*

and thus

$$\mathcal{A}_j(\theta_j^i \to \tilde{\theta}_j^i) = \min\left(1, \frac{\pi_j(\tilde{\theta}_j^i)}{\pi_j(\theta_j^i)} \cdot \frac{\pi_{j+1}(\theta_j^i)}{\pi_{j+1}(\tilde{\theta}_j^i)}\right). \tag{3.2}$$

Note this equation does not depend on M (the number of steps for the coarser chain at layer $j+1$). While this chain almost certainly has not mixed for small M, the ratio $q_j(\theta_j^i|\tilde{\theta}_j^i)/q_j(\tilde{\theta}_j^i|\theta_j^i)$ is the same as if the chain had completely mixed and the proposed new state, $\tilde{\theta}_j^i$, were from the true posterior of the model at layer $j+1$. The values $\pi_{j+1}(\tilde{\theta}_j^i)$ and $\pi_{j+1}(\theta_j^i)$ were already calculated as part of the chain at layer $j+1$ and therefore do not take any additional computation time. FLARE MCMC is summarized in Algorithm 3.1. To gather N samples from the true posterior, the algorithm is called with FLARE-Chain$(\theta^0, N, J=0)$.

3.2. Convergence rate. We show a convergence rate for FLARE MCMC. We measure the distance to the stationary distribution in terms of total variation distance as follows.

Definition 3.1 (Strasser (1985) [54]). *The total variation distance between two probability measures ν_1 and ν_2 is defined as*

$$\|\nu_1 - \nu_2\| = \sup_A |\nu_1(A) - \nu_2(A)|.$$

The minorization condition of Markov chains, used by Roberts and Rosenthal [50], provides a means of bounding the convergence rate. For any Markov chain with a one-step transition probability of $p(\theta^i \to \theta^{i+1})$, we let $p^n(\theta^i \to \theta^{i+n})$ denote the corresponding n-step transition probability. Formally, the definition of the minorization condition is stated below.

Definition 3.2 (Roberts and Rosenthal (2004) [50]). *A Markov chain on Θ satisfies the minorization condition if there exists an $\epsilon > 0$, a positive integer n, and a probability measure $\nu(.)$ such that*

$$p^n(\theta^0 \to \theta^n) \geq \epsilon\nu(\theta^n) \qquad \forall \theta^0, \theta^n \in \Theta. \tag{3.3}$$

With respect to the stationary distribution, the probability of transitioning from θ to θ' can be minorized by a lower bound such that $p_j^1(\theta \to \theta') \geq \epsilon\pi_j(\theta')$.

Lemma 3.3. *Assume the minorization condition holds at the innermost level ($j = J$): $p_J^1(\theta_J^i \to \theta_J^{i+1}) \geq \xi_J \cdot \pi_J(\theta_J^{i+1})$ for some $\xi_J > 0$. Then, there exists a minorized lower bound on levels $j < J$ such that*

$$p_j^1(\theta_j^i \to \theta_j^{i+1}) \geq \xi_j \cdot \pi_j(\theta_j^{i+1}) \qquad \forall \theta_j^i, \theta_j^{i+1}, \tag{3.4}$$

where $\xi_j = (1 - (1 - \xi_{j+1})^M) \cdot \min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$.

We call ξ_j the minorization constant for level j. This satisfies the necessary minorization condition of Theorem 8 in [50]. This allows us to get a quantitative bound on the distance to the stationary distribution of every level as stated in Theorem 3.4. The proof for Lemma 3.3 can be found in Appendix A.

Theorem 3.4. *Let $p_j^n(\theta_j^0 \to \cdot)$ be the distribution for layer j with an invariant target probability $\pi_j(\cdot)$. FLARE MCMC is uniformly ergodic and converges as $\|p_j^n(\theta_j^0 \to \cdot) - \pi_j(\cdot)\| \leq (1-\xi_j)^n$, where $\xi_j = (1 - (1 - \xi_{j+1})^M) \cdot \min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$ as in Lemma 3.3. Most critically, this theorem holds for layer $j = 0$.*

Proof. The results follow from the minorization condition established in Lemma 3.3. ∎

Although the theorem above establishes the convergence of the chain to the invariant target distribution, it illustrates several aspects about the inner chains. In particular, ergodicity requires that the chain be able to reach all regions of the target's support. Therefore, the coarser approximations' supports must be supersets of the finer ones. The coupling strength, ξ_j, in turn depends on the coupling strength of the coarser approximations' chains, $\xi_k, k > j$. Thus, the effects of the mixing times of the inner chains on the outer chain are captured in this theorem.

The similarities of the approximations are captured in the $\min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$ terms. Thus, the theorem also quantifies the effects of similarities and dissimilarities among the approximations on the total convergence rate. If the modes of the target distribution are preserved across coarsening, then we would expect these terms to be larger and therefore the outer chain to mix faster. Ideally, the minimum in this term could be replaced with an expectation, thus turning it into the Kullback–Liebler (KL) divergence between adjacent layers. We have not yet determined whether this is possible.

3.3. Optimal number of inner steps M.

To understand how to select the number of inner steps for each layer M_j, we derive a theoretical expression for optimal M_j that balances the decoupling rate of the chains and the cost of likelihood evaluations. The following lemma provides an analytical expression for the value of M_j that maximizes the cost-aware decoupling rate. The proof for Lemma 3.5 can be found in Appendix B.

Lemma 3.5. *For layers $0 \leq j < J$, suppose the minorization condition holds such that $p_j^1(\theta_j^i \to \theta_j^{i+1}) \geq \xi_j \cdot \pi_j(\theta_j^{i+1})$, where $\xi_j = (1 - (1 - \xi_{j+1})^{M_j}) \cdot \min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$ is the minorization constant, and $M_j \geq 0$ is the number of inner steps. Let the total cost per step at level j be $B_j = b_j + M_j \cdot B_{j+1}$, where $b_j > 0$ is the cost of evaluating the likelihood at level j, and $B_{j+1} > 0$ is the cost of a single evaluation of the inner layer $j+1$. Define the cost-aware computational decoupling rate as*

$$f(M_j) = \frac{\left(1-(1-\xi_{j+1})^{M_j}\right)\cdot \min_{\theta}\left(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)}\right)}{B_j}. \tag{3.5}$$

Then the real-valued maximizer M_j^* of $f(M_j)$ is

$$M_j^* = -\frac{1}{\Upsilon} W_{-1}\left(-e^{-\Upsilon\mu}\right) - \mu, \tag{3.6}$$

where $\Upsilon = -\log(1-\xi_{j+1})$, $\mu = \frac{b_j}{B_{j+1}} + \frac{1}{\Upsilon}$, and W_{-1} is -1 branch of the Lambert W function.

However, this expression is not directly usable in practice, since it depends on the unknown coupling minorization constant ξ_{j+1}, which is generally unknown and difficult to approximate in MCMC settings. The optimizer M_j^* is a real-valued quantity, whereas in practice the number of inner steps must be an integer. Instead, we have found empirical values for M that offer the best trade-off between computation time and sampling efficiency in the experiment section 4. Nevertheless, this lemma provides a theoretical benchmark for the optimal trade-off between computational cost of likelihoods and effective mixing across layers.

3.4. Layer tuning. The algorithm above uses the coarser fidelities to guide the finer ones. Early in the chain, this is useful for quickly driving the samples toward high-probability regions. However, this mismatch between the fidelities can cause problems later because it can steer the chain away from high-probability regions in the fine fidelity model that do not overlap with high-probability regions of the coarse fidelity model. Figure 2 demonstrates an example of such a partial, but not complete, overlap in one of our examples. To combat this, we present a simple modification that requires neither estimation of any internal variables of the probability models nor estimation of means or variances from multiple chains.

Recall $\pi_j(\theta_j)$ is known up to a normalizing constant: $\pi_j(\theta_j) = \tilde{\pi}_j(\theta_j)/Z_j$ where $\tilde{\pi}_j(\theta_j)$ is the unnormalized distribution, and Z_j is the normalizing constant. We modify the target distributions for coarser chains (and thus the proposal distributions for all $j > 0$) as

$$\psi_j(\theta_j) = (\tilde{\pi}_j(\theta_j) + \omega_j)/\zeta_j(\omega_j) \qquad \forall\ 0 < j \le J, \tag{3.7}$$

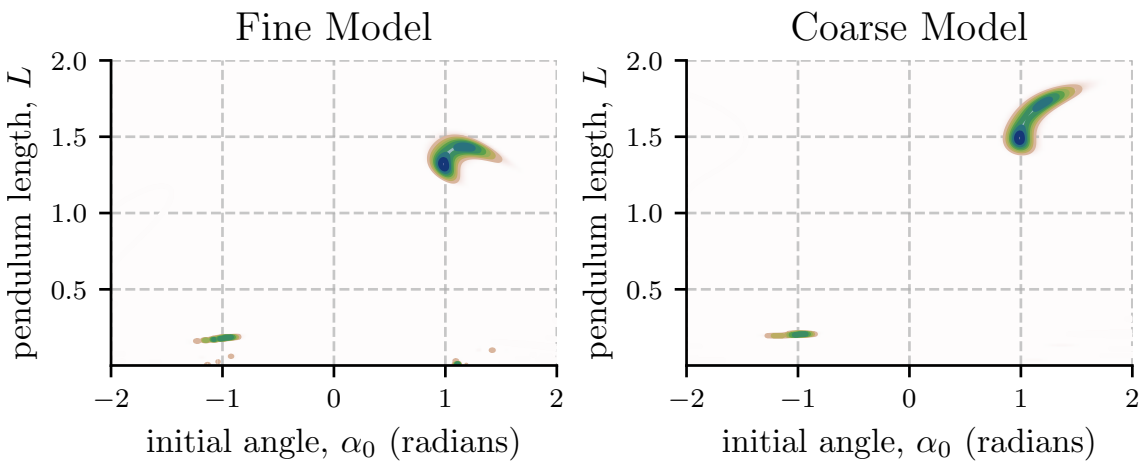


Figure 2. *Posterior of different fidelities; the coarse model is the small angle approximation. See section 4.1.*

where $\zeta_j(\omega_j)$ is the normalizing constant of this new distribution which depends on ω_j.[1]

We now use $\psi_j(\theta_j)$ in place of $\pi_j(\theta_j)$ in equation (3.2), therefore modifying the acceptance ratio for all $0<j\leq J$ as

$$\mathcal{A}_j(\theta_j^i \to \tilde{\theta}_j^i) = \min\left(1, \frac{\psi_j(\tilde{\theta}_j^i)}{\psi_j(\theta_j^i)} \cdot \frac{\psi_{j+1}(\theta_{j+1}^i)}{\psi_{j+1}(\tilde{\theta}_{j+1}^i)}\right) . \tag{3.8}$$

For the finest layer, things remain the same (or, alternatively, $\omega_0 = 0$), because we do not want to change the distribution of the overall sampler.

This effectively mixes the stationary distribution of the jth layer with a uniform distribution (we have added a constant to the posterior and then renormalized), encouraging the proposal to explore more widely than the coarser layer would normally. While this method is unsophisticated, we found it simpler to implement and compute than other options and just as effective.

Instead of leaving $(\omega_1, \omega_2, \ldots, \omega_J)$ as hyperparameters, we use gradient descent to adapt them over the course of the sampling. We adjust ω_{j+1} to minimize the KL divergence between layers ψ_j and ψ_{j+1}:

$$\mathrm{KL}(\psi_j \| \psi_{j+1}) = \mathop{\mathbb{E}}_{\theta\sim\psi_j}[\ln(\psi_j)] - \mathop{\mathbb{E}}_{\theta\sim\psi_j}[\ln(\psi_{j+1})] . \tag{3.9}$$

This tries to make the coarser (approximating) distribution ψ_{j+1} more similar to the distribution ψ_j. Because the first term does not depend on ω_{j+1}, the objective function is to maximize

$$H_{j+1} = \mathop{\mathbb{E}}_{\theta\sim\psi_j}[\ln(\psi_{j+1})] . \tag{3.10}$$

Using equation (3.7), we have

$$\begin{aligned} \frac{\partial}{\partial\omega_{j+1}} H_{j+1} &= \frac{\partial}{\partial\omega_{j+1}} \left(\mathop{\mathbb{E}}_{\theta\sim\psi_j}[\ln(\tilde{\pi}_{j+1}(\theta) + \omega_{j+1})] - \ln\zeta_{j+1}(\omega_{j+1}) \right) \\ &= \mathop{\mathbb{E}}_{\theta\sim\psi_j}\left[\frac{\partial}{\partial\omega_{j+1}} \ln(\tilde{\pi}_{j+1}(\theta) + \omega_{j+1})\right] - \mathop{\mathbb{E}}_{\theta\sim\psi_{j+1}}\left[\frac{\partial}{\partial\omega_{j+1}} \ln(\tilde{\pi}_{j+1}(\theta) + \omega_{j+1})\right], \end{aligned} \tag{3.11}$$

where the second step replaces the derivative of the log-partition function with the expected derivative of the log-probability.

The first term is an expectation with respect to the distribution at the lower layer j. We assume that the lower layer is mixed, and therefore the starting state for the chain at layer $j+1$ is a sample drawn from ψ_j. The second term is an expectation with respect to the distribution at this layer, $j+1$. We let the sample at the *end* of this chain after M steps approximate a sample from this distribution. This is similar to the approximation employed by M-step contrastive divergence [33]. Although this is not guaranteed to converge [55], in

[1] We assume the domain of θ, Θ, is of finite volume.

Algorithm 3.2. FLARE-with-layer-tuning(θ_j^0, n, j).

for $i = 0, \ldots, n-1$ **do**
  if $j = J$ **then** ▷ coarsest layer
    Sample $\tilde{\theta}_j^i$ from $q_j(\cdot|\theta_j^i)$
    Accept $\theta_j^{i+1} = \tilde{\theta}_j^i$ with probability $\mathcal{A}$ from equation (2.2)
    Otherwise, reject and $\theta_j^{i+1} = \theta_j^i$
  else
    $\theta_{j+1}^1, \ldots, \theta_{j+1}^M =$ FLARE-with-layer-tuning(θ_j^i, M, $j+1$)
    $\tilde{\theta}_j^i = \theta_{j+1}^M$
    Update gradient of ω_{j+1} using $\frac{\partial}{\partial \omega_{j+1}} H_{j+1}$ from equation (3.12)
    Accept $\theta_j^{i+1} = \tilde{\theta}_j^i$ with probability $\mathcal{A}_j$ from equation (3.8)
    Otherwise, reject and $\theta_j^{i+1} = \theta_j^i$
return $\theta_j^1, \ldots, \theta_j^n$

practice we have found it to work well. Thus, the total derivative for the gradient ascent update is

$$\frac{\partial}{\partial \omega_{j+1}} H_{j+1} \approx \frac{1}{\tilde{\pi}_{j+1}(\theta_{j+1}^0) + \omega_{j+1}} - \frac{1}{\tilde{\pi}_{j+1}(\theta_{j+1}^M) + \omega_{j+1}}. \tag{3.12}$$

Note that these denominators are calculated during the MCMC chain, and therefore the derivative requires very little extra computation.

A single ω_j is kept for each layer and maintained across subchains at that layer. We use a learning rate of 10^{-3} to adjust ω_j for all experiments. An update is made on layer j once after each M-step subchain.

To make the innermost Gaussian proposal more robust, we adaptively update the covariance of the proposal distribution as initially proposed in the AM algorithm [27]. We use the history of chains from the coarsest layer $\theta_J^0, \theta_J^1, \ldots, \theta_J^t$ to update the covariance for the inner most proposal distribution. By using all previous states of the coarsest layer, the proposal distribution quickly adapts using the accepted samples. This rapid start of adaptation ensures good mixing in the inner most layer which gives higher quality candidate samples for the finer chains. We show the recursive algorithm with layer tuning adaptation added in Algorithm 3.2. Here we update the gradient after M steps of each layer and use it in the acceptance probability with ω mixed in as a uniform distribution to the target distribution.

3.5. Ergodicity of layer tuning.

We show that FLARE MCMC with adaptive tuning of the proposals at each layer is ergodic. This can be shown with diminishing adaptation and simultaneous uniform ergodicity.

Lemma 3.6. *For layers $0 \le j < J$, let $\gamma_j \in \Gamma_j$ be the adaptations for the proposal at layer j or the chain at level $j+1$, i.e., $\gamma_j = \omega_{j+1} \leftrightarrow \psi_{j+1}(\theta) \leftrightarrow p_{j+1}(\theta \to \cdot)$ where $\Gamma_j \in \mathbb{R}$ and ψ_{j+1} is the target at layer $j+1$ with the layer tuning adaptation added. Let $p_{j,\gamma_j}(\theta \to \cdot)$ denote the transition distribution of chain at level j using adaptation γ_j, starting in state θ. Assume for*

all $j, \omega_j \in [\underline{\omega}, \overline{\omega}]$ *for some* $0 < \underline{\omega} < \overline{\omega}$, *and, at the innermost layer, there exists a minorization constant* $\xi_J > 0$ *such that* $\|p^M_{J,\gamma_J}(\theta \to \cdot) - \psi_J(\cdot)\| \leq (1 - \xi_J)^M$. *Then, we have the following:*

(a) *Simultaneous uniform ergodicity: For all* $\tau > 0$. *there exists* $n = n(\tau) \in \mathbb{N}$ *such that*

$$\left\|p^n_{J,\gamma_J}(\theta \to \cdot) - \psi_j(\cdot)\right\| \leq \tau \tag{3.13}$$

for all $\theta \in \Theta$ *and* $\gamma_j \in \Gamma_j$.

(b) *Diminishing adaptation: The amount of adaptation diminishes in probability with the number of steps* t *in the adaptation as*

$$\lim_{t\to\infty} \sup_{\theta} \left\|p_{j,\gamma_j^t}(\theta \to \cdot) - p_{j,\gamma_j^{t+1}}(\theta \to \cdot)\right\| = 0. \tag{3.14}$$

The proof for Lemma 3.6 can be found in Appendix C.

Theorem 3.7. *FLARE MCMC with an adaptive layer tuning parameter is ergodic.*

Proof. We use Lemma 3.6 to show the conditions necessary in Theorem 1 of Roberts and Rosenthal [51]. This shows that the adaptive algorithm is ergodic. ∎

4. Experiments. We measure the efficiency of the MCMC methods tested using the effective sample size (ESS) [49] estimated across multiple chains as

$$N_{ESS} = (N \cdot K) / \left(1 + 2 \sum_{k=1}^{2m+1} \rho(k)\right), \tag{4.1}$$

where N is the number of samples, K is the number of chains, $\rho(k)$ is the lag-k correlation, and m is the largest value such that $\rho(2m) + \rho(2m+1) > 0$. We compute ESS for each parameter for the "bulk" (entire distribution) and "tail" (largest and smallest 5% of the samples) of the distributions.

We compare FLARE MCMC with standard Metropolis–Hastings (with proposal adaptation introduced by Haario, Saksman, and Tamminen [27]) and other multifidelity MCMC methods: multilevel delayed acceptance MCMC (MLDA) [43], MLDA with adaptive error model (AEM) [12, 43], multilevel MCMC (MLMCMC) [14, 44], and Multifidelity Monte Carlo (MFMC) [8]. We give the following detail about the different methods used for comparison:

1. MLDA without any adaptation: Introduced by Lykkegaard et al. [43], this method uses recursive chains of approximations as proposals. However, there is no adaptation being done to "correct" the approximations. Even the authors note that without adaptation, the chains do not mix well and have poor effective sample sizes.
2. MLDA with AEM: Extended by Lykkegaard et al. in the same paper [43], this method uses a similar structure to that above. They also use adaptive error model (AEM) as a way to deal with the discrepancies between the different layers, as it uses a telescoping sum of differences in the mean of the approximations. They demonstrate with the subsurface flow model that MLDA with AEM leads to good mixing and high ESS. We demonstrate similar results for two of our experiments. Our subsurface flow model experiment uses the same fidelities as set up by the original authors; however, we run it to collect more samples using a higher number of chains.

3. Multilevel MCMC (MLMCMC): This method was proposed by Dodwell et al. [14] and was then applied to MLDA. A quantity of interest, Q, is proposed that is related to the parameters of the model. The samples drawn from the posterior are used to reduce the variance of Q. Since in MLDA, samples are drawn from both the "true" posterior as well as the approximations, and the samples from the approximate levels are used to reduce the variance of Q. They state that it thus requires fewer samples to achieve the same variance. Using a telescopic sum, the difference in Q estimates between levels is used to correct Q with respect to the next coarser level. For the pendulum model, Q is the mean of the outputs. For the subsurface flow experiment used by the original authors of MLDA [43], Q is the hydraulic head at some fixed point $(x, y) = (0.5, 0.45)$; that is, the model PDE is solved at these points at each level using samples from the coarser approximate level.
4. Multi-Fidelity Monte Carlo (MFMC): This method was proposed by Cai and Adams [8]. It uses a continuum of models with increasing fidelity and has a single Markov chain with a random choice of the fidelity, K, at each step. The fidelity K is part of the sampled state-space and therefore also part of the proposal distribution and acceptance probability. We map K to a reasonable range of fidelities for each experiment. For the pendulum model, we map K to the error tolerance of the integrator, ϵ, as $\epsilon = e^{K/10} + 10^{-6}$. For the subsurface flow experiment, we let the grid resolution be equal to $10K$ (K is the sampled fidelity of this method) in order to map the resolution to the fidelity range expected by the algorithm's implementation. The samples from this method are not from the true posterior; rather, they can be corrected to estimate an expectation (like the mean). Therefore, we do not plot the evolution of effective samples with respect to time in the results, as they are not samples from the true posterior.

For the MLDA-based methods, we use the authors' implementations in the open-source probabilistic programming package PyMC3 [52] by Lykkegaard et al. [43]. For MFMC, we use the author-provided implementation.

These implementations have significant computational overhead compared with our implementation of FLARE MCMC (see supplementary material (supplement.zip [local/web 5.13KB])). Therefore, we only measure the time taken in likelihood computation (which is the same code for all methods).

We present three different experimental posterior sampling problems across different scientific domains: a simple pendulum, a hydrology simulation that was used by prior methods as a benchmark, and a cosmology simulation that stresses computational limits.

Because of the computational expense of the cosmology simulation, we are not able to collect a sufficient number of samples to get a reliable estimate of the effective sample sizes. Instead, we compare our estimates to those in the cosmology literature.

For each experimental domain, we construct three fidelities by adjusting the relevant simulation parameter. In all cases, we measure our abilities to sample from the highest fidelity ($j = 0$). For methods labeled "(single)," there is a single higher fidelity layer ($J = 1$). For methods labeled "(double)," there are two higher fidelity layers ($J = 2$): the one from the "(single)" experiments and one that is even more coarse. For the coarsest fidelity, a Gaussian proposal distribution is used with an adaptive covariance matrix.

In the pendulum and cosmology experiments, this normal distribution is reflected to keep parameters within their respective ranges. FLARE MCMC can be extended beyond $J = 2$ layers. However, just two layers improves over the standard MCMC and other multilevel methods significantly. Layers' costs should be roughly orders-of-magnitude different in computational costs. For these examples, $J=2$ is the limit of how many layers can practically be constructed with orders-of-magnitude different computational costs.

4.1. Simple pendulum. The equation of motion for a pendulum of length L, mass M, and initial angle α_0 is $\ddot{\alpha} = -(g/L)\sin\alpha$. Our goal is to sample from the posterior of the distribution the two parameters $\theta = (L, \alpha)$ conditioned on the observations of α at three irregularly spaced times during the motion: $\alpha(1) = -0.85$, $\alpha(2.3) = 0.9$, $\alpha(5.0) = 0.95$. Observations of these angles are assumed to be corrupted by Gaussian noise with known standard deviation: $\sigma = 0.1$. Different fidelities correspond to adjusting the error tolerance of an adaptive Runge–Kutta 4(5) ODE integrator [15] (10^{-3} or 10^{-6} in our experiments) with stepsize control and dense output [28]. As a separate coarsest layer of approximation, we use the small angle approximation (which does not hold for the observations), $\sin(\alpha) \approx \alpha$, reducing the equation of motion to a simple harmonic motion which can be solved analytically as $\alpha(t) = \alpha_0 \cos(t\sqrt{g/L})$. The difference between the finest fidelity posterior and this small angle approximation is shown in Figure 2.

Results. To judge the importance of setting M, we evaluated our method across different values of M with fixed total computation time. The effective sample size (ESS) is plotted as a function of M for the two competing multilevel methods in Figure 3. We can see that beyond $M = 5$, the increased computation time from running longer inner subchains leads to a decrease in overall sampling efficiency, indicating that $M = 5$ offers the best trade-off between computation time and sampling efficiency. Therefore, the layered subchains were run for $M = 5$ steps.

We ran 10 chains of the finest fidelity for all methods. We replicated this experiment (of 10 chains) 50 times. Table 1(a) summarizes the mean effective sample size per second (ESS/s) and the average mean of the parameters across all 500 chains with the standard deviation along with the acceptance rates for every layer.

Figure 4 shows ESS (across all 10 chains) as a function of computation time for each method, with the total number of samples (N) generated in 1000 s. The standard deviations are plotted as (barely visible) vertical bars. For the sake of readability, we have separated our plots to show how each method performs with one level of nesting (single) and two levels of nesting (double).

We note that the MFMC method obtains significantly more samples in the same time budget. This result arises from its randomized fidelity selection, which collects samples at low-fidelity evaluations more frequently than other methods. Since the samples from this method are not from the true posterior, we do not list the effective samples in the table. However, for comparison, we note that the average bulk and tail ESS/s for 50 runs of 10 chains measured for parameters $[\alpha, l]$ are $[24.80, 20.46]$ and $[26.52, 30.61]$ for MFMC.

All methods are able to improve by using more fidelities. Our FLARE MCMC method is consistently and significantly better than the other methods (including the best one, MLDA

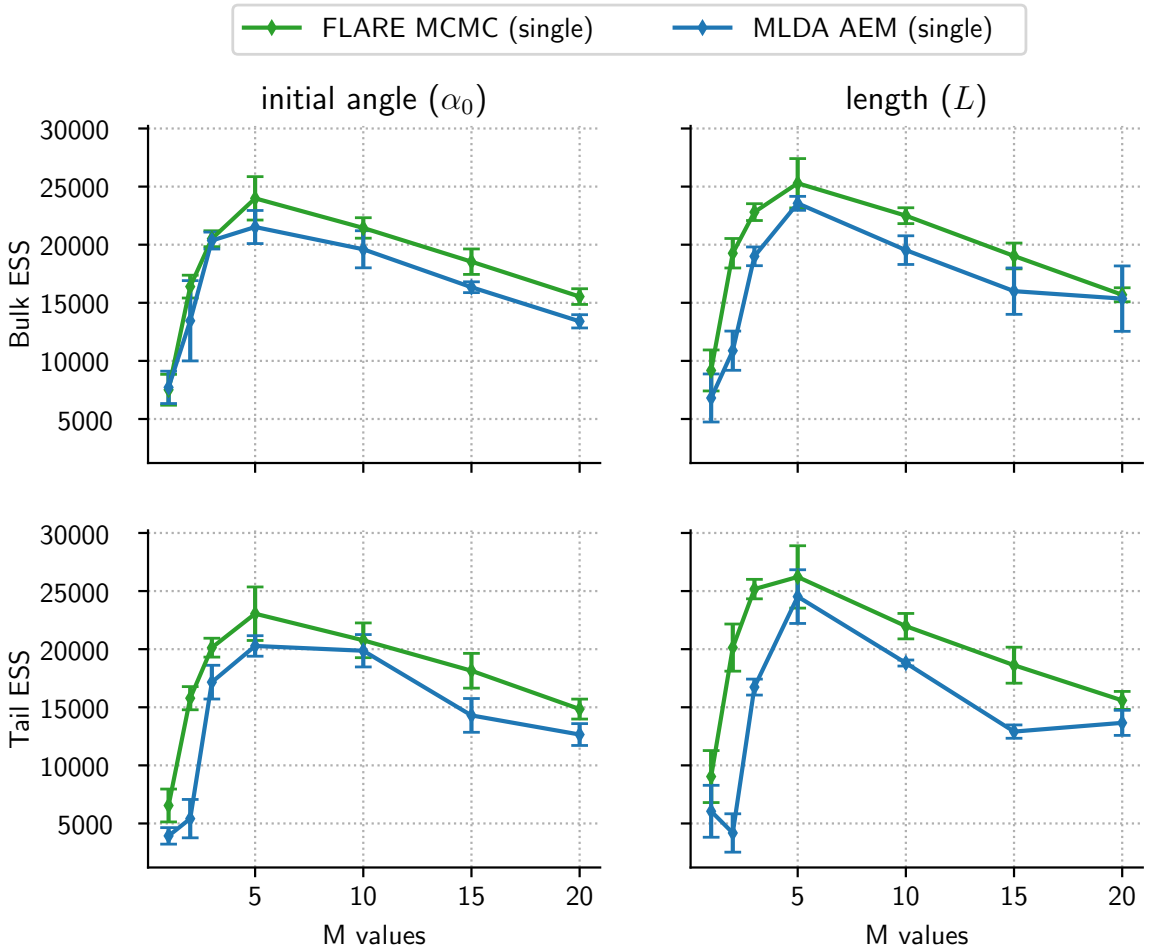


Figure 3. *ESS as a function of M for two single-layered ($J = 1$) methods, with fixed computation time of 500 seconds, each run across 50 different chains.*

with AEM as shown in Table 1) in terms of ESS/s in both the bulk and tail of the distribution. The acceptance rates for the multilevel methods indicate that while the coarsest level accepts about one-third of the samples (consistent with Gelman, Roberts, and Gilks [21]), the proposed sample from that level is accepted by the finer levels frequently since it was already accepted by an approximate coarse level. The mean of the parameter across different runs of the standard MCMC has a higher standard deviation compared to the other multilevel methods, suggesting that some runs of MCMC do a poor job at finding the modes in the posterior. FLARE MCMC produces more samples in the same amount of time compared to its competing method MLDA, reflecting that our sampler requires fewer likelihood evaluations per step.

Figure 5 shows the distributional distance to the true distribution as a function of number of samples, averaged over 500 chains. The true mean and standard deviation are unknown for all the experiments in the paper, and measuring the distance between a multidimensional distribution which can be evaluated (only up to a normalizing constant) and a distribution represented by samples is nontrivial. However, we have analyzed the 1-dimensional marginals

Table 1

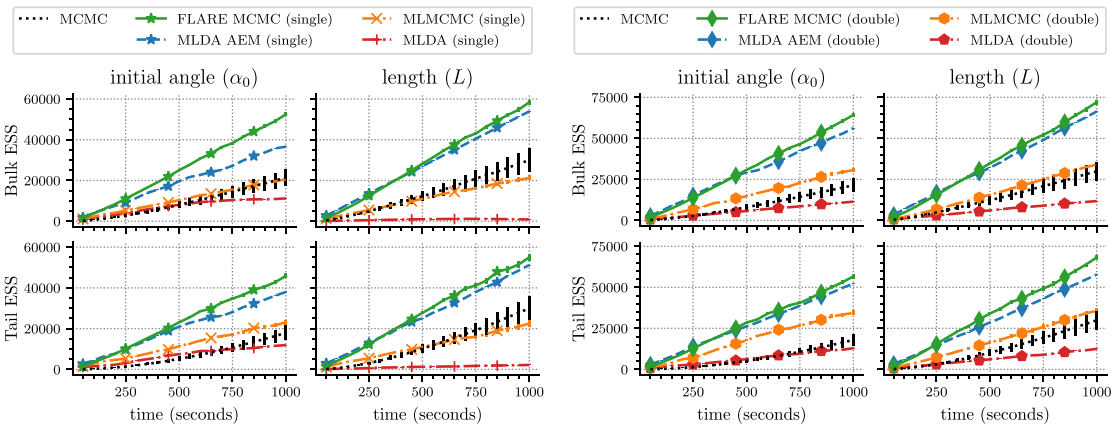
Mean ESS for bulk and tail distributions across 50 *runs for* 10 *chains: we list the mean value of each parameter with standard error across all* 500 *total runs. Average acceptance rates are listed per layer, and the total samples are based on the average cost of likelihood evaluations per sample per method. Each chain is run for a total of* 1000 *seconds.*

	α				L				Acceptance rate			total samples
	bulk ESS/s	tail ESS/s	mean	sd	bulk ESS/s	tail ESS/s	mean	sd	$j=0$	$j=1$	$j=2$	
MCMC	21.43	29.94	0.953	0.492	17.94	29.94	1.265	0.355	0.29			100000
FLARE MCMC (single)	**52.47**	**58.42**	1.081	0.019	**45.85**	**54.95**	1.372	0.016	0.98	0.24		60000
AEM MLDA (single)	39.41	53.91	1.064	0.062	42.20	52.24	1.371	0.051	0.98	0.27		50000
MLMCMC (single)	25.03	34.43	1.044	0.291	21.54	32.99	1.351	0.167	0.92	0.32		44500
MLDA (single)	11.64	1.49	1.059	0.041	15.36	7.73	1.361	0.022	0.91	0.31		45000
FLARE MCMC (double)	**64.26**	**72.10**	1.086	0.001	**56.68**	**68.20**	1.374	0.009	0.99	0.86	0.29	35000
AEM MLDA (double)	56.92	65.53	1.085	0.006	50.95	58.51	1.375	0.003	0.98	0.89	0.3	25000
MLMCMC (double)	33.82	37.05	1.049	0.015	33.09	39.94	1.361	0.015	0.90	0.81	0.28	25500
MLDA (double)	10.01	7.54	1.053	0.035	4.96	12.99	1.362	0.024	0.86	0.72	0.24	30000
MFMC			1.075	0.057			1.348	0.132	0.46			370000

(a) Simple pendulum

	θ_1				θ_2				θ_3				Acceptance rates			Total samples
	bulk ESS/s	tail ESS/s	mean	sd	bulk ESS/s	tail ESS/s	mean	sd	bulk ESS/s	tail ESS/s	mean	sd	$j=0$	$j=1$	$j=2$	
MCMC	5.35	7.81	-0.457	0.0037	5.49	8.04	0.466	0.0036	5.63	8.19	0.076	0.0034	0.27			10000
FLARE MCMC (single)	**8.74**	**11.99**	-0.460	0.0030	**8.58**	**11.81**	0.467	0.0036	**8.63**	**11.58**	0.076	0.0028	0.98	0.26		8365
AEM MLDA (single)	7.77	4.67	-0.459	0.0032	6.96	4.29	0.466	0.0033	8.16	6.24	0.076	0.0030	0.99	0.29		4100
MLMCMC (single)	4.35	5.86	-0.460	0.0028	4.37	3.27	0.465	0.0031	5.29	6.49	0.077	0.0021	0.93	0.32		8000
MLDA (single)	3.65	1.52	-0.463	0.0035	4.78	1.56	0.490	0.0036	3.33	3.02	0.075	0.0031	0.87	0.34		8250
FLARE MCMC (double)	**15.49**	**21.35**	-0.460	0.0026	**15.141**	**19.69**	0.469	0.0026	**15.07**	**19.76**	0.077	0.0023	0.99	0.95	0.3	6500
AEM MLDA (double)	13.29	14.70	-0.460	0.0035	12.32	13.22	0.468	0.0035	12.70	14.45	0.077	0.0033	0.99	0.93	0.24	2285
MLMCMC (double)	8.46	10.49	-0.459	0.0027	8.04	9.13	0.468	0.0030	7.47	5.81	0.077	0.0024	0.92	0.89	0.31	6000
MLDA (double)	4.48	5.05	-0.461	0.0036	5.33	6.60	0.469	0.0033	5.28	7.35	0.076	0.0030	0.93	0.91	0.27	6350
MFMC			-0.475	0.026			0.418	0.041			0.102	0.003	0.39			5700

(b) Subsurface flow model



(a) Single (b) Double

Figure 4. *Pendulum model: ESS for bulk and tail across* 50 *runs (mean and standard deviation) for single and double layers of nesting.*

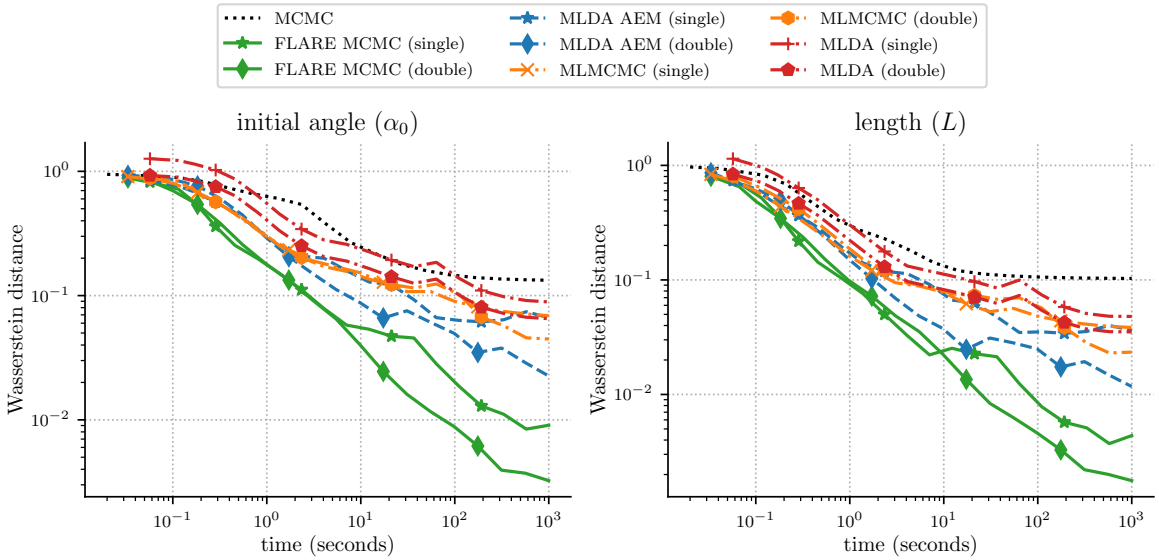


Figure 5. *Wasserstein distance to true distributions for the pendulum model.*

of the pendulum model in the following way. We evaluate the true unnormalized distribution on a grid, normalize it, project it to the marginal of interest, and then treat it as (weighted) samples for a sample-to-sample Wasserstein distance between it and the samples from the MCMC methods. As we refine the grid, the distances become smaller (for almost all methods). We refine the grid until these distances stabilize, resulting in about a 1000-by-1000 grid (1 million log likelihood evaluations). From Figure 5, it is clear that for both parameters, FLARE MCMC ends up with the smallest Wasserstein distance to the true distribution.

4.2. Estimation of soil permeability in subsurface flow. We consider a simple problem in subsurface flow modeling [14]. This model was also used to evaluate the MLDA methods by the original authors [43], and we did not modify the code used by MLDA (except to increase the number of chains and amount of measured time).

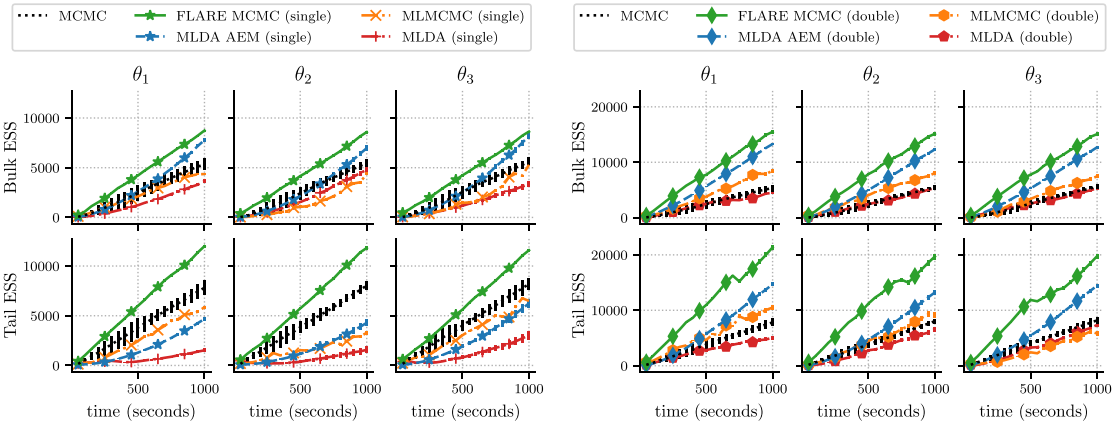
The classical equations governing (steady-state) single-phase subsurface flow consist of Darcy's law coupled with an incompressibility condition,

$$w + k\nabla p = g \text{ and } \nabla \cdot w = 0, \tag{4.2}$$

subject to suitable boundary conditions. All quantities are fields over $\mathcal{D} = [0,1]^2 \subset \mathbb{R}^2$ for these experiments. Here p denotes the hydraulic head of the fluid, k is the permeability tensor, w is filtration velocity (or Darcy flux), and g is the (known) source term.

We are interested in the permeability given observations (with known-variance Gaussian noise) of the hydraulic head at 16 regularly spaced points in $\mathcal{D}$. k is simplified to be the gradient of a random scalar field. The log-Gaussian scalar field is parameterized with a truncated Karhunen–Loéve expansion (to three terms, following MLMCMC [14]). These three parameters (θ) have a standard normal prior, and we sample from their posterior.

Computing the likelihood involves solving a partial differential equation (PDE) with known boundary conditions for a given θ and comparing the results for p at the observation points.



(a) Single (b) Double

Figure 6. *Subsurface flow model: ESS for bulk and tail across* 50 *runs (mean and standard deviation) for single and double layers of nesting.*

The fidelities correspond to different grid resolutions for the PDE solver: 120×120 (highest), 30×30, and 10×10 (coarsest).

Results. Previous work reports that $M = 5$ achieves the best trade-off between effective sample size and computation time for this experimental setup [43]. Therefore, we adopt the same value to ensure a fair comparison with our method. Table 1(b) summarizes the same statistics for this model with the same setup as the pendulum experiments. Figure 6 shows ESS (across all 10 chains) as a function of computation time for each method. In terms of ESS/s, our method improves over the standard MCMC and outperforms the multilevel methods for the same amount of likelihood computational budget, especially in the tail of the distribution. All methods converge to similar means of the parameters with low standard deviation among chains.

We note that MFMC collects fewer samples compared to other methods since each sample requires multiple log-likelihood calculations in the same high fidelity to update K, leading to a significant increase in computational costs. For MFMC, the estimated mean ESS/s for bulk and tail for parameters $[\theta_1, \theta_2, \theta_3]$ are $[0.128, 0.081, 0.710]$ and $[0.227, 0.161, 1.107]$. But, again, the samples from MFMC were never intended to be interpreted as being from the true distribution.

4.3. Structure formation in the universe with N-body gravitational simulation. An important problem in modern-day cosmology is to generate theoretical models of the universe on very large scales (tens of MegaParsec (Mpc) across) that can be compared to observations. Bayesian inference allows cosmologists to measure quantities of fundamental physics significance, such as the nature of dark energy and dark matter [47]. The theoretical models needed for next-generation telescopes, such as Euclid [6] and the Roman Space Telescope (WFIRST) [53], are based on expensive numerical simulations, some of which require many days of computer time for each evaluation. For such a computationally expensive model, we show the efficacy of FLARE MCMC as compared with the standard Metropolis–Hastings algorithm and MLDA.

One of the most frequently used summary statistics is the galaxy power spectrum, $\boldsymbol{P}_{\text{gg}}$ (a bold $\boldsymbol{P}$ is a power spectrum, not a distribution): the two-point clustering of galaxies in Fourier space as a function of the wavenumber scale, k. We use a slightly simplified model for the galaxy power spectrum for (relative) ease of computation. We perform a forward simulation which starts from a given set of cosmological parameters and predicts the galaxy power spectrum. It works by following the evolution of the universe under the influence of gravity, from its beginnings in an almost uniform density state to the diverse collection of galaxies sitting in dark matter potentials observed today.

We sample from the posterior density of the following four cosmological parameters:

θ_1 This is the dimensionless Hubble constant, h, which characterizes the universal expansion rate and thus the recession velocity of distant galaxies. A redshift zero galaxy at distance d Mpc recedes at a speed $v = H_0 d$, where $H_0 = h \times 100\,\text{km}\,\text{s}^{-1}\text{Mpc}^{-1}$. Measuring h is of importance in understanding dark energy.

θ_2 This is the dimensionless total matter density, $0 < \Omega_0 < 1$. Ω_0 is the energy density of matter as a function of the critical density. Ω_0 is important because it can be used to infer the density of dark matter.

θ_3 This is the dimensionless scalar perturbation amplitude, A_s, of the primordial fluctuations at the wavenumber $k = 0.05\,\text{Mpc}^{-1}$. A_s is of interest because it connects to the uncertain high energy physics of the early universe. Larger values of A_s correspond to a clumpy early universe and so lead to larger $\boldsymbol{P}_{\text{gg}}$.

θ_4 This is the dimensionless linear bias, b, which is used to shift the amplitude of our simulated matter power spectrum to match the amplitude of the galaxy power spectrum. This accounts for the difference between observed galaxies and dark matter (which is used by the forward model),

$$\boldsymbol{P}_{\text{model}}(\theta) = b^2 \cdot \boldsymbol{P}_{\text{dm}}(h, \Omega_0, A_s), \tag{4.3}$$

where b is the scale-independent linear bias, and $\boldsymbol{P}_{\text{dm}}$ is the simulated dark-matter power spectrum directly computed from the output density field of FastPM.

The posterior density is conditioned on the galaxy power spectrum from SDSS-III Baryon Oscillation Spectroscopic Survey (BOSS) Data Release 12 (DR12), our observational data source [13, 4]. We have used a subset of the BOSS data from the North Galactic Cap (NGC) at $z = 0.38$, which includes $\sim 10^6$ galaxies, from [37].

The likelihood function is a multivariate Gaussian between the galaxy power spectrum from BOSS, $\boldsymbol{P}_{\text{gg}}$, and the galaxy power spectrum from the forward model, $\boldsymbol{P}_{\text{model}}(\theta)$:

$$\ln \mathcal{L}(\theta) = -\frac{1}{2}(\boldsymbol{P}_{\text{model}}(\theta) - \boldsymbol{P}_{\text{gg}})^{\intercal} \mathbf{C}^{-1} (\boldsymbol{P}_{\text{model}}(\theta) - \boldsymbol{P}_{\text{gg}}) + k. \tag{4.4}$$

$\mathbf{C}$ is the covariance matrix of the galaxy power spectrum, also estimated observationally.

The most expensive part of the forward model, evolution under gravitational force, is computed using FastPM [19]. FastPM has a couple of tunable fidelity parameters. The size of the region simulated controls the amount of data available and may have a nonlinear effect on the accuracy of the result. We thus fix the size of this region to 1024 Mpc/h and instead change the number of particles. More particles in the simulation mean higher resolution and a more accurate power spectrum at the higher wavenumber k, and thus the likelihood function

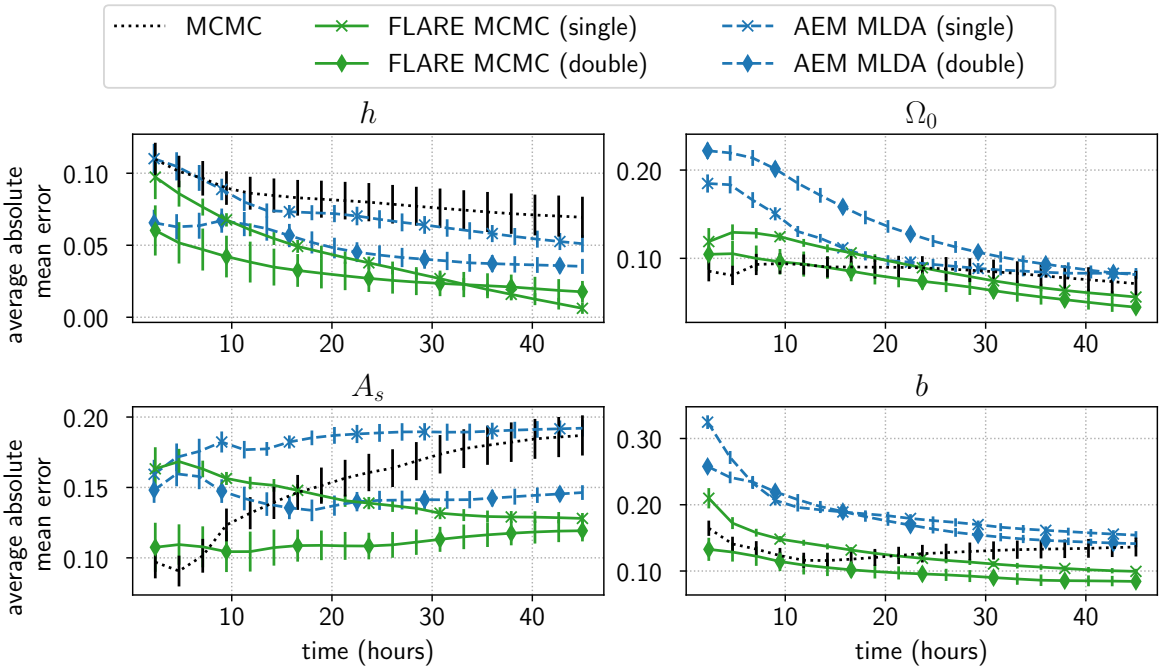


Figure 7. *Average (across* 10 *chains) absolute error in mean estimates for the four parameters, as a function of total simulation time. In the allotted* 48 *hours, each chain sampled* 800 *samples for (plain) MCMC,* 600 *samples for FLARE MCMC (single),* 500 *samples for FLARE MCMC (double),* 290 *samples for MLDA (single) and* 200 *samples for MLDA (double).*

is higher fidelity. For N-body simulations, the compute time usually scales as $N \log N$, where N is the number of particles. Thus a simulation with a 512^3 number of particles is $\simeq 80$ times more expensive than a 128^3 simulation. We therefore set the fidelities by only adjusting the number of particles used in the simulation to be 512 (highest), 384, and 256 (coarsest). Note this calculation is distributed across 20 cores (using MPI) and therefore a savings of 1 hour corresponds to 20 core-hours.

We compare our estimated distributional means to previous computations on the same data. For the parameters h and Ω_0, we compare to the means reported by Ivanov et al. [37] on the same data using their own MCMC simulation ($h = 0.661$ and $\Omega_m = 0.290$). We have only a single linear bias term, compared with the multiple terms of Ivanov, Simonović, and Zaldarriaga [37]. Therefore, we can compare neither b nor A_s (which is heavily related to b) to their results. Instead, we measure A_s against the best fit value from the Planck satellite [2], $A_s = 2.09$ and $b = 2$, consistent with comparable BOSS measurements [37]. While these are modes (and not means), they are the best independent estimates we can obtain.

Results. Extreme running time dictated smaller values for M for this experiment. We reduced them by a factor of 2 (approximately) and used $M = 2$ for inner substeps. Figure 7 shows that the FLARE MCMC methods converge to the mean values from previous literature better than the standard Metropolis–Hastings method using fewer samples and less time. The A_s parameter has slightly strange behavior. We can still see better convergence of our methods. However, note that the best-fit value of A_s we are taking as "ground truth"

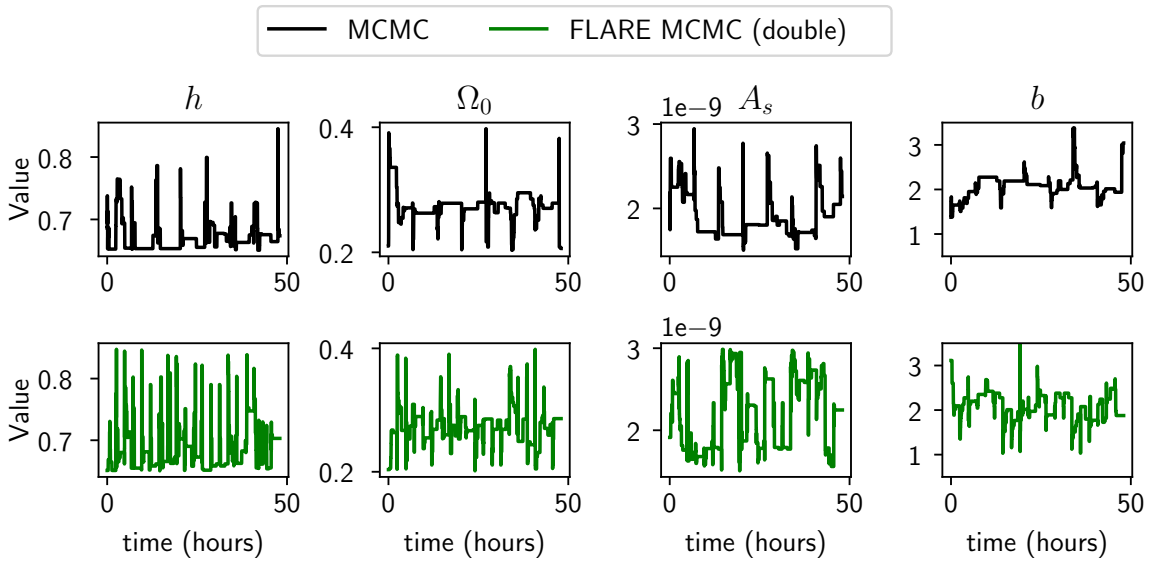


Figure 8. *Cosmology model; trace plot for a random single run of each method.*

is measured (with error) from a different dataset and thus is likely not the true mean of our posterior. Many large-scale structure experiments prefer a lower value of this parameter than that preferred by Planck, a feature known as the S8 tension [1]. The pairwise plots of the posterior can be found in Appendix F. From the posteriors, it is clear that FLARE MCMC is better at approximating the modes of the distribution as compared to MCMC. Figure 8 shows the trace plot for a random run of MCMC and FLARE MCMC. Our method shows better mixing than MCMC and is less likely to reject proposed samples.

5. Summary. Many scientific and engineering problems involve simulations or solving differential equations. In this paper, we present an efficient multifidelity layered MCMC that exploits the ability to reduce the accuracy of models, leading to approximations of the posterior. In a recursive, nested fashion, these approximations act as proposals for MCMC-based inference. We add layer tuning that successfully encourages the approximate proposals to explore the distribution well. We demonstrate with experimental results using models from three different scientific domains with varying costs that out method, FLARE MCMC, is simple yet produces more efficient samples than those of existing adaptive multilevel MCMC methods with the same computational budget.

Appendix A. Convergence rate proofs. In this appendix, we show proofs for the convergence rates in the main text. We first use Lemma A.1 to show that after M steps of a coarse chain, we can obtain a minorized lower bound that can be recursively used in its finer layer.

Lemma A.1. *Let $p_j(\cdot \to \cdot)$ be the transition distribution of the Markov chain at level j with an invariant target distribution $\pi_j(\cdot)$. For any level j, if there exists a $\xi_j > 0$ such that $p_j^1(\theta_j^0 \to \theta_j^1) \geq \xi_j \pi_j(\theta_j^1)$ for all $\theta_j^0, \theta_j^1 \in \Theta$, then*

(A.1) $$p_j^M(\theta_j^0 \to \theta_j^M) \geq \left(1 - (1-\xi_j)^M\right) \pi_j(\theta_j^M) \qquad \forall \theta_j^0, \theta_j^M \in \Theta.$$

Proof. We prove this using induction. We can verify the base case for $k = 1$ such that $p_j^1(\theta_j^0 \to \theta_j^1) \geq (1 - (1-\xi_j)^1)\pi_j(\theta_j^k) = \xi_j \pi_j(\theta_j^1)$. This is held by the assumption made in the lemma.

Assume using the induction hypothesis that $p_j^k(\theta_j^0 \to \theta_j^k) \geq (1 - (1-\xi_j)^k)\pi_j(\theta_j^k)$. We need to show that $p_j^{k+1}(\theta_j^0 \to \theta_j^{k+1}) \geq (1 - (1-\xi_j)^{k+1})\pi_j(\theta_j^{k+1})$.

Note $p_j^k(\theta_j^0 \to \theta_j^k)$ can be written as $p_j^k(\theta_j^0 \to \theta_j^k) = \phi^k \pi_j(\theta_j^k) + (1-\phi^k) r_j(\theta_j^k \mid \theta_j^0)$, where ϕ^k is the probability that the chain couples to the stationary distribution in k steps, and $r_j(\theta_j^k \mid \theta_j^0)$ is the remaining distribution that depends on θ_j^0.

$$\begin{aligned}
p_j^{k+1}(\theta_j^0 \to \theta_j^{k+1}) &= \int \left[p_j^k(\theta_j^0 \to \theta_j^k) \cdot p_j^1(\theta_j^k \to \theta_j^{k+1}) \right] d\theta_j^k \\
&= \int p_j^1(\theta_j^k \to \theta_j^{k+1}) \left[\phi^k \pi_j(\theta_j^k) + (1-\phi^k) r_j(\theta_j^k \mid \theta_j^0) \right] d\theta_j^k \\
&= \int \phi^k \pi_j(\theta_j^k) p_j^1(\theta_j^k \to \theta_j^{k+1}) d\theta_j^k + \int (1-\phi^k) r_j(\theta_j^k \mid \theta_j^0) p_j^1(\theta_j^k \to \theta_j^{k+1}) d\theta_j^k.
\end{aligned}$$

We know $\phi^k \geq 1 - (1-\xi_j)^k$. Thus,

$$\geq (1 - (1-\xi_j)^k)\pi_j(\theta_j^{k+1}) + (1-\xi_j)^k \int r_j(\theta_j^k \mid \theta_j^0) p_j^1(\theta_j^k \to \theta_j^{k+1}) d\theta_j^k.$$

Replacing $p_j^1(\theta_j^k \to \theta_j^{k+1})$ between two consecutive samples with the base assumption,

$$\begin{aligned}
&\geq (1 - (1-\xi_j)^k)\pi_j(\theta_j^{k+1}) + \xi_j (1-\xi_j)^k \int r_j(\theta_j^k \mid \theta_j^0) \pi_j(\theta_j^{k+1}) d\theta_j^k \\
&= (1 - (1-\xi_j)^k)\pi_j(\theta_j^{k+1}) + \xi_j (1-\xi_j)^k \pi_j(\theta_j^{k+1}) \\
&= (1 - (1-\xi_j)^{k+1})\pi_j(\theta_j^{k+1}).
\end{aligned}$$

Therefore, using proof by induction, we have that $p_j^M(\theta_j^0 \to \theta_j^M) \geq (1 - (1-\xi_j)^M)\pi_j(\theta_j^M)$. ∎

Proof of Lemma 3.3. The transition kernel is given by

$$\begin{aligned}
p_j^1(\theta_j^i \to \theta_j^{i+1}) &= \mathcal{A}_j(\theta_j^i \to \theta_j^{i+1}) \cdot q_j(\theta_j^{i+1} | \theta_j^i) + \delta(\theta_j^{i+1} - \theta_j^i) \int \left(1 - \mathcal{A}_j(\theta_j^i \to \theta_j')\right) q_j(\theta_j' |) \theta_j^i \, d\theta_j' \\
&\geq \mathcal{A}_j(\theta_j^i \to \theta_j^{i+1}) \cdot q_j(\theta_j^{i+1} | \theta_j^i) \\
&= \mathcal{A}_j(\theta_j^i \to \theta_j^{i+1}) \cdot p_{j+1}^M(\theta_{j+1}^0 \to \theta_{j+1}^M).
\end{aligned}$$

The sample θ_j^{i+1} is proposed using the Mth sample from the $j+1$ chain and therefore is the same as θ_{j+1}^M. Thus, from Lemma A.1,

$$\begin{aligned}
&\geq \mathcal{A}_j(\theta_j^i \to \theta_j^{i+1}) \cdot (1 - (1-\xi_{j+1})^M) \cdot \pi_{j+1}(\theta_j^{i+1}) \\
&= \min\left(1, \frac{\pi_j(\theta_j^{i+1})}{\pi_j(\theta_j^i)} \cdot \frac{\pi_{j+1}(\theta_j^i)}{\pi_{j+1}(\theta_j^{i+1})}\right) \cdot (1 - (1-\xi_{j+1})^M) \cdot \pi_{j+1}(\theta_j^{i+1}).
\end{aligned}$$

Let $r(\theta) = \frac{\pi_{j+1}(\theta)}{\pi_j(\theta)}$. Then,

$$\begin{aligned}
&= \min\left(1, \frac{r(\theta_j^i)}{r(\theta_j^{i+1})}\right) \cdot (1-(1-\xi_{j+1})^M) \cdot r(\theta_j^{i+1}) \cdot \pi_j(\theta_j^{i+1}) \\
&= (1-(1-\xi_{j+1})^M) \cdot \min\left(r(\theta_j^{i+1}), r(\theta_j^i)\right) \cdot \pi_j(\theta_j^{i+1}) \\
&\geq (1-(1-\xi_{j+1})^M) \cdot \min_\theta (r(\theta)) \cdot \pi_j(\theta_j^{i+1}) \\
&= \xi_j \cdot \pi_j(\theta_j^{i+1}),
\end{aligned}$$

where $\xi_j = (1-(1-\xi_{j+1})^M) \cdot \min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$. ∎

Appendix B. Optimal M proof.

Proof of Lemma 3.5. Since the ratio $\min_\theta(\frac{\pi_{j+1}(\theta)}{\pi_j(\theta)})$ is constant with respect to M_j, we simplify the objective function to be maximized as

$$f(M_j) = \frac{1-(1-\xi_{j+1})^{M_j}}{b_j + M_j B_{j+1}}, \qquad M_j \geq 0, \quad \xi_{j+1} \in (0,1), \quad b_j, B_{j+1} > 0.$$

We define

$$c := 1-\xi_{j+1} \in (0,1), \qquad \beta := \frac{b_j}{B_{j+1}}, \qquad \Upsilon := -\log c > 0, \qquad \text{and} \qquad \mu := \beta + \frac{1}{\Upsilon}.$$

Then,

$$f(M_j) = \frac{1-c^{M_j}}{\beta B_{j+1} + M_j B_{j+1}} = \frac{1-c^{M_j}}{B_{j+1}(\beta + M_j)}.$$

Differentiating the objective function, we get

$$f'(M_j) = \frac{-c^{M_j} \log c\,(\beta + M_j) - (1-c^{M_j})}{(\beta + M_j)^2 * B_{j+1}}.$$

Setting the derivative to zero and using $\Upsilon = -\log c$ gives

$$-c^{M_j} \log c\,(\beta + M_j) = 1 - c^{M_j} \quad \Longrightarrow \quad c^{M_j}\,(1 + \Upsilon(\beta + M_j)) = 1.$$

Since $c^{M_j} = e^{\log(c^{M_j})} = e^{M_j \log c} = e^{M_j(-\Upsilon)}$, we replace $c^{M_j} = e^{-\Upsilon M_j}$ to obtain

$$(1 + \Upsilon(\beta + M_j))\, e^{-\Upsilon M_j} = 1.$$

By the definition of μ, we have $1 + \Upsilon(\beta + M_j) = \Upsilon\mu + \Upsilon M_j$. Therefore,

$$(\Upsilon\mu + \Upsilon M_j) e^{-\Upsilon M_j} = 1.$$

Let $y = \Upsilon\mu + \Upsilon M_j$. Then $\Upsilon M_j = y - \Upsilon\mu$, and substituting this gives

$$\begin{aligned}
y e^{-(y-\Upsilon\mu)} &= 1, \\
y e^{-y} &= e^{-\Upsilon\mu}, \\
-y e^{-y} &= -e^{-\Upsilon\mu}.
\end{aligned}$$

Using the Lambert W function for branch $k=-1$, we get

$$y = -W_{-1}\left(-e^{-\Upsilon\mu}\right). \tag{B.1}$$

Since $\Upsilon M_j = y - \Upsilon\mu$, we get

$$M_j = \frac{y}{\Upsilon} - \mu = -\frac{1}{\Upsilon}W_{-1}\left(-e^{-\Upsilon\mu}\right) - \mu.$$

Therefore, the maximizer for our objective function is

$$M_j^* = -\frac{1}{\Upsilon}W_{-1}\left(-e^{-\Upsilon\mu}\right) - \mu, \tag{B.2}$$

where $\Upsilon = -\log(1-\xi_{j+1})$, $\mu = \frac{b_j}{B_{j+1}} + \frac{1}{\Upsilon}$, and W_{-1} is the -1 branch of the Lambert W function. ∎

Appendix C. Layer tuning ergodicity proof.

Proof of Lemma 3.6.

(a) Consider layer $J-1$. Using Theorem 3.4, we get

$$\left\|p^M_{J-1,\gamma_{J-1}}(\theta\to\cdot) - \psi_{J-1}(\cdot)\right\| \le (1-\xi_{J-1})^M, \quad \xi_{J-1} = (1-(1-\xi_J)^M)\min_\theta \frac{\psi_J(\theta)}{\psi_{J-1}(\theta)}.$$

From the definition of layer tuning,

$$\begin{aligned}
&= (1-(1-\xi_J)^M)\min_\theta \frac{(\tilde\pi_J(\theta)+\omega_J)\cdot\zeta_{J-1}(\omega_{J-1})}{(\tilde\pi_{J-1}(\theta)+\omega_{J-1})\cdot\zeta_J(\omega_J)}\\
&= (1-(1-\xi_J)^M)\min_\theta \frac{\tilde\pi_J(\theta)+\omega_J}{\tilde\pi_{J-1}(\theta)+\omega_{J-1}}\\
&\quad\times\frac{Z+\omega_{J-1}\cdot V}{Z+\omega_J\cdot V},
\end{aligned}$$

where ζ is the normalizing constant of the new distribution that depends on ω, Z is the normalizing constant of the original distribution, and V is the volume of Θ. With the bounds for ω from the assumption,

$$\begin{aligned}
&\ge (1-(1-\xi_J)^M)\min_\theta\left(\frac{\tilde\pi_J(\theta)+\underline\omega}{\tilde\pi_{J-1}(\theta)+\overline\omega}\right)\left(\frac{Z+\underline\omega\cdot V}{Z+\overline\omega\cdot V}\right)\\
&\triangleq \overline\xi_{J-1}.
\end{aligned}$$

By induction with base case at layer J,

$$\forall j, \left\|p^n_{j,\gamma_j}(\theta\to\cdot) - \psi_j(\cdot)\right\| \le (1-\xi_j)^n \text{ where } \xi_j \ge \overline\xi_j\ . \tag{C.1}$$

We need to show that for all $\tau > 0$, there exists $n = n(\tau) \in \mathbb{N}$ such that

$$\left\| p^n_{j,\gamma_j}(\theta \to \cdot) - \psi_j(\cdot) \right\| \leq \tau$$

for all $\theta \in \mathcal{X}_j$ and $\gamma_j \in \Gamma_j$.
From equation (C.1), we want

$$(1 - \xi_j)^n \leq \tau$$
$$n \geq \frac{\ln \tau}{\ln(1 - \xi_j)}$$

Since $\ln(1 - \xi_j) \leq \ln(1 - \overline{\xi_j})$,

$$n \geq \frac{\ln \tau}{\ln(1 - \overline{\xi_j})} .$$

Thus, for all $\tau > 0$, there exists $n = \max_\tau \frac{\ln \tau}{\ln(1-\overline{\xi_j})}$ such that $\|p^n_{j,\gamma_J}(\theta \to \cdot) - \psi_j(\cdot)\| \leq \tau$.

(b) At every step t, the change in γ_j maps to change in ω_{j+1}. Diminishing adaptation is guaranteed by a gradient descent algorithm with diminishing stepsize that updates ω_{j+1} at each layer to minimize the Kullback–Leibler divergence between layers ψ_j and ψ_{j+1}. At each step of the gradient descent algorithm, ω_{j+1} is updated as $\omega^{t+1}_{j+1} = \omega^t_{j+1} - \eta_i \frac{\partial}{\partial \omega_{j+1}} H_{j+1}$. To get diminishing adaptation, the update needs to converge as

$$\lim_{t\to\infty} \left\| \eta_t \frac{\partial}{\partial \omega_{j+1}} H_{j+1} \right\| \approx 0.$$

Since we bound $\gamma_j \leftrightarrow \omega_{j+1}$ away from zero,

$$\begin{aligned}
\frac{\partial}{\partial \omega_{j+1}} H_{j+1} &= \frac{1}{\tilde{\pi}_{j+1}(\theta^0_{j+1}) + \omega_{j+1}} - \frac{1}{\tilde{\pi}_{j+1}(\theta^M_{j+1}) + \omega_{j+1}} \\
&\leq \frac{1}{\tilde{\pi}_{j+1}(\theta^0_{j+1}) + \omega_{j+1}} \\
&\leq \frac{1}{\tilde{\pi}_{j+1}(\theta^0_{j+1}) + \underline{\omega}} \\
&\leq \frac{1}{\underline{\omega}}.
\end{aligned}$$

Therefore, the update is

$$\begin{aligned}
\lim_{t\to\infty} \left\| \eta_t \frac{\partial}{\partial \omega_{j+1}} H_{j+1} \right\| &\leq \lim_{t\to\infty} \left\| \eta_t \frac{1}{\underline{\omega}} \right\| \\
&\leq \frac{1}{\underline{\omega}} \lim_{t\to\infty} \eta_t.
\end{aligned}$$

If the stepsize η_t asymptotes to 0, adaptation decreases to 0 as $t \to \infty$. Therefore $\lim_{t\to\infty} \|\omega^t_{j+1} - \omega^{t+1}_{j+1}\| = 0$, and thus $\lim_{t\to\infty} \sup_\theta \|p_{j,\gamma^t_j}(\theta \to \cdot) - p_{j,\gamma^{t+1}_j}(\theta \to \cdot)\| = 0$. ■

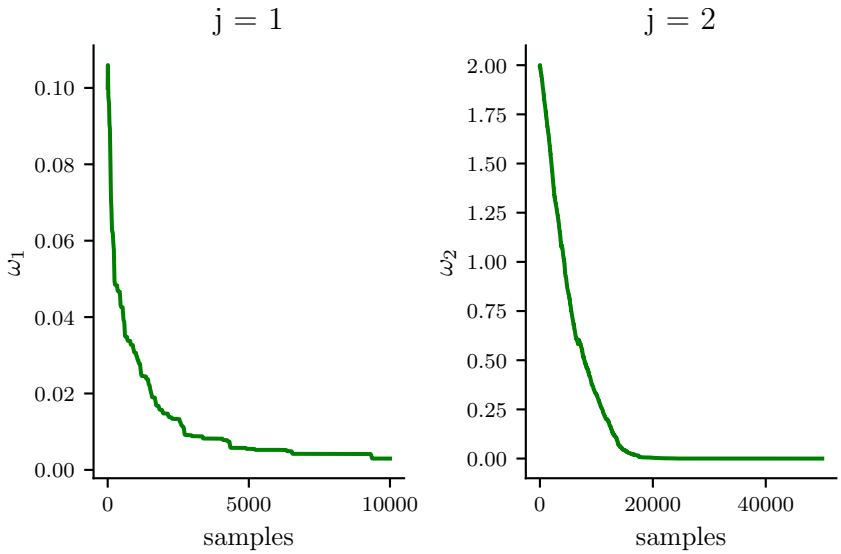


Figure 9. *Evolution of ω_j for doubly nested layers $j = 1$ and $j = 2$.*

Appendix D. Uniform smoothing parameter. Plotted in Figure 9 is the evolution of our tuning parameter ω_j as a function of samples collected for one example chain of the doubly nested method for the pendulum model. Each sample at $j = 1$ starts a chain of length $M = 5$ at inner layer $j = 2$. The inner layer $j = 2$ uses the small angle approximation of the pendulum as the fidelity. Since it is a poor approximation of the posterior as shown in Figure 2, we start with a relatively high value of ω. This helps the coarsest layer better explore the high-probability regions. As shown, the tuning parameter ω converges close to zero after a few samples in both layers. We use a learning rate of 10^{-3} for both layers.

Appendix E. Computational infrastructure. Our experiments were performed on a machine with 4 Intel Xeon Silver 4214 CPUs running at 2.20GHz for our experiments (a total of 48 cores). The machine has 250GB of memory, but memory was never a restriction during our experiments.

All methods use multiprocessing, that is, each chain is run in parallel using a different core. The time listed is across one run of a single chain; however, the effective sample size is calculated across 10 different chains.

For the pendulum and hydrology models, likelihood calculations were carried out on a single core. For the cosmology model, the likelihood calculations were carried out in parallel across 20 cores. Therefore, for the cosmology experiments, saving a day's worth of computation time on the graphs corresponds to saving 20 days worth of core-hours.

Appendix F. Pairwise plot for cosmology model. Plotted in Figure 10 is the pairwise plot for all four parameters of the cosmology model. MCMC generates more samples in the

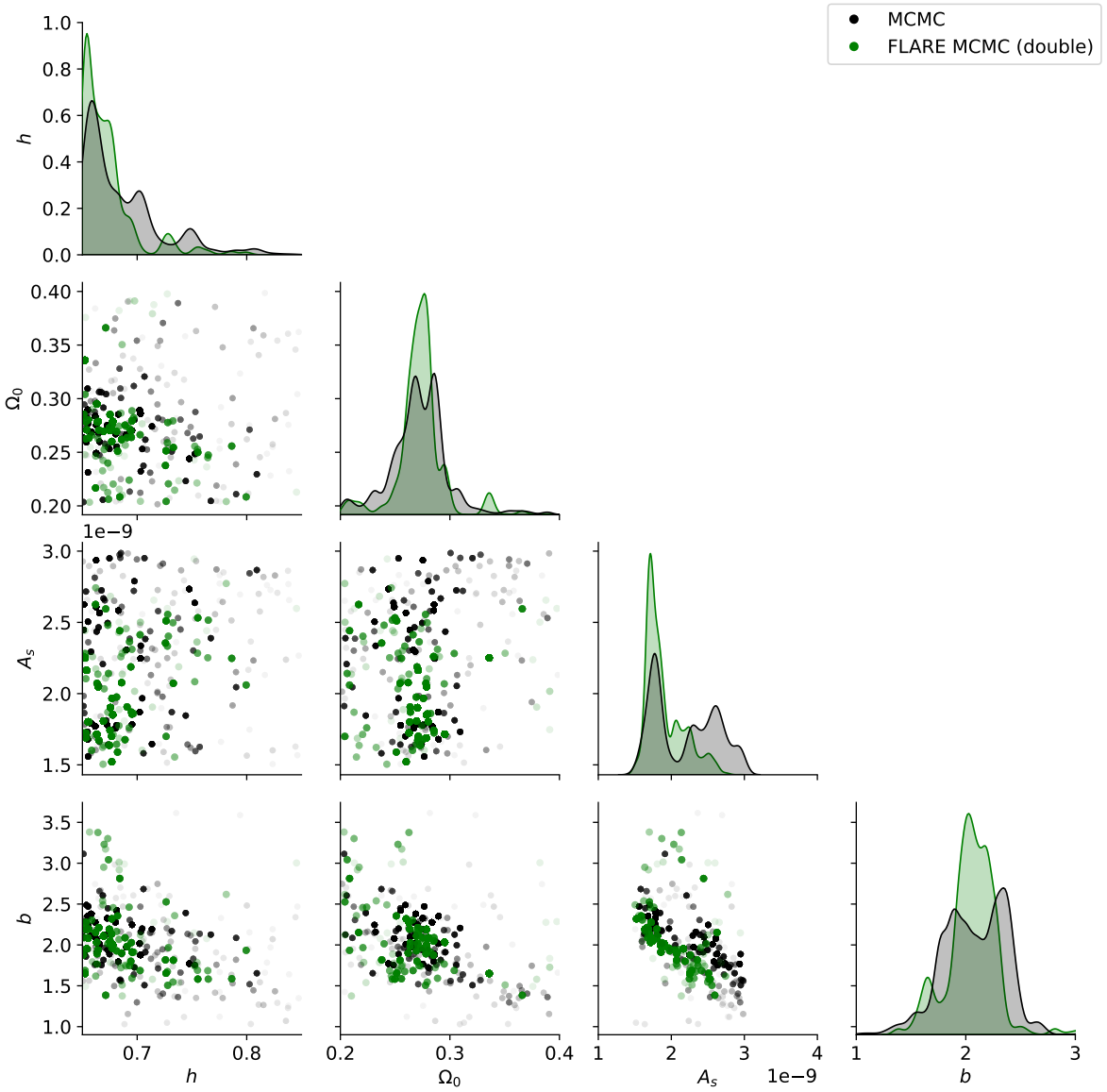


Figure 10. *Cosmology model: Pairwise plots for all four parameters plotted for a random run of samples collected for* 48 *hours of MCMC (black) and FLARE MCMC (green).*

same period of time, yet these samples have not yet converged to the distribution and are still scattered across the space, compared with the relatively compact FLARE MCMC samples.

REFERENCES

[1] E. Abdalla, G. F. Abellán, A. Aboubrahim, A. Agnello, Ö. Akarsu, Y. Akrami, G. Alestas, et al., *Cosmology intertwined: A review of the particle physics, astrophysics, and cosmology associated with the cosmological tensions and anomalies*, J. High Energy Astrophys., 34 (2022), pp. 49–211, https://doi.org/10.1016/j.jheap.2022.04.002.

[2] N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A. Banday, et al., *Planck* 2018 *results*, Astron. Astrophys., 641 (2020), A6, https://doi.org/10.1051%2F0004-6361%2F201833910.

[3] F. Al-Awadhi, M. Hurn, and C. Jennison, *Improving the acceptance rate of reversible jump MCMC proposals*, Statist. Probab. Lett., 69 (2004), pp. 189–198, https://doi.org/10.1016/j.spl.2004.06.025.

[4] S. Alam, M. Ata, S. Bailey, F. Beutler, D. Bizyaev, J. Blazek, A. Bolton, et al., *The clustering of galaxies in the completed SDSS*-III *Baryon Oscillation Spectroscopic Survey: Cosmological analysis of the DR*12 *galaxy sample*, Mon. Not. Roy. Astron. Soc., 470 (2017), pp. 2617–2652, https://doi.org/10.1093/mnras/stx721.

[5] G. Altekar, S. Dwarkadas, J. P. Huelsenbeck, and F. Ronquist, *Parallel Metropolis coupled Markov chain Monte Carlo for Bayesian phylogenetic inference*, Bioinformatics, 20 (2004), pp. 407–415, https://doi.org/10.1093/bioinformatics/btg427.

[6] L. Amendola, S. Appleby, A. Avgoustidis, D. Bacon, T. Baker, M. Baldi, et al., *Cosmology and fundamental physics with the Euclid satellite*, Living Rev. Relativ., 21 (2018), 2, https://doi.org/10.1007/s41114-017-0010-3.

[7] A. Beskos, A. Jasra, K. Law, R. Tempone, and Y. Zhou, *Multilevel sequential Monte Carlo samplers*, Stochastic Process. Appl., 127 (2017), pp. 1417–1440, https://doi.org/10.1016/j.spa.2016.08.004.

[8] D. Cai and R. P. Adams, *Multi-fidelity Monte Carlo: A pseudo-marginal approach*, in Advances in Neural Information Processing Systems, Vol. 35, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, eds., Curran Associates, 2022, pp. 21654–21667, https://proceedings.neurips.cc/paper_files/paper/2022/file/8803b9ae0b13011f28e6dd57da2ebbd8-Paper-Conference.pdf.

[9] J. A. Christen and C. Fox, *Markov chain Monte Carlo using an approximation*, J. Comput. Graph. Stat., 14 (2005), pp. 795–810, https://doi.org/10.1198/106186005X76983.

[10] P. R. Conrad, Y. M. Marzouk, N. S. Pillai, and A. Smith, *Accelerating asymptotically exact MCMC for computationally intensive models via local approximations*, J. Amer. Stat. Assoc., 111 (2016), pp. 1591–1607, https://doi.org/10.1080/01621459.2015.1096787.

[11] T. Cui, C. Fox, and M. O'Sullivan, *A posteriori stochastic correction of reduced models in delayed-acceptance MCMC, with application to multiphase subsurface inverse problems*, Internat. J. Numer. Methods Engrg., 118 (2019), pp. 578–605, https://doi.org/10.1002/nme.6028.

[12] T. Cui, C. Fox, and M. O'Sullivan, *Adaptive Error Modelling in MCMC Sampling for Large Scale Inverse Problems*, Tech. report 687, Faculty of Engineering, University of Auckland, 2012, https://www.researchgate.net/publication/50981205_Adaptive_Error_Modelling_in_MCMC_Sampling_for_Large_Scale_Inverse_Problems.

[13] K. S. Dawson, D. J. Schlegel, C. P. Ahn, S. F. Anderson, É. Aubourg, S. Bailey, R. H. Barkhouser, et al., *The Baryon oscillation spectroscopic survey of SDSS*-III, Astron. J., 145 (2013), 10, https://doi.org/10.1088/0004-6256/145/1/10.

[14] T. J. Dodwell, C. Ketelsen, R. Scheichl, and A. L. Teckentrup, *A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow*, SIAM/ASA J. Uncertain. Quantif., 3 (2015), pp. 1075–1108, https://doi.org/10.1137/130915005.

[15] J. Dormand and P. Prince, *A family of embedded Runge-Kutta formulae*, J. Comput. Appl. Math., 6 (1980), pp. 19–26, https://doi.org/10.1016/0771-050X(80)90013-3.

[16] A. Doucet, N. de Freitas, and N. Gordon, *An introduction to sequential Monte Carlo methods*, in Sequential Monte Carlo Methods in Practice, Statist. Engrg. Inform. Sci., A. Doucet, N. de Freitas, and N. Gordon, eds., Springer, New York, 2001, pp. 3–14, https://doi.org/10.1007/978-1-4757-3437-9_1.

[17] S. Duane, A. Kennedy, B. J. Pendleton, and D. Roweth, *Hybrid Monte Carlo*, Phys. Lett. B, 195 (1987), pp. 216–222, https://doi.org/10.1016/0370-2693(87)91197-X.

[18] Y. Efendiev, T. Hou, and W. Luo, *Preconditioning Markov chain Monte Carlo simulations using coarse-scale models*, SIAM J. Sci. Comput., 28 (2006), pp. 776–803, https://doi.org/10.1137/050628568.

[19] Y. Feng, M.-Y. Chu, U. Seljak, and P. McDonald, *FASTPM: A new scheme for fast simulations of dark matter and haloes*, Mon. Not. Roy. Astron. Soc., 463 (2016), pp. 2273–2286, https://doi.org/10.1093/mnras/stw2123.

[20] C. Fox and G. Nicholls, *Sampling conductivity images via MCMC*, in The Art and Science of Bayesian Image Analysis, 1997, pp. 91–100, https://scholar.google.com/citations?view_op=view_citation&hl=en&user=mnq3780AAAAJ&citation_for_view=mnq3780AAAAJ:UeHWp8X0CEIC.
[21] A. Gelman, G. O. Roberts, and W. R. Gilks, *Efficient Metropolis jumping rules*, in Bayesian Statistics, J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, eds., Oxford University Press, Oxford, 1996, pp. 599–608, https://doi.org/10.1093/oso/9780198523567.003.0038.
[22] M. B. Giles, *Multilevel Monte Carlo path simulation*, Oper. Res., 56 (2008), pp. 607–617, https://doi.org/10.1287/opre.1070.0496.
[23] P. J. Green, *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination*, Biometrika, 82 (1995), pp. 711–732, https://doi.org/10.1093/biomet/82.4.711.
[24] P. J. Green and A. Mira, *Delayed rejection in reversible jump Metropolis–Hastings*, Biometrika, 88 (2001), pp. 1035–1053, https://doi.org/10.1093/biomet/88.4.1035.
[25] A. Gregory and C. J. Cotter, *A seamless multilevel ensemble transform particle filter*, SIAM J. Sci. Comput., 39 (2017), pp. A2684–A2701, https://doi.org/10.1137/16M1102021.
[26] A. Gregory, C. J. Cotter, and S. Reich, *Multilevel ensemble transform particle filtering*, SIAM J. Sci. Comput., 38 (2016), pp. A1317–A1338, https://doi.org/10.1137/15M1038232.
[27] H. Haario, E. Saksman, and J. Tamminen, *An adaptive Metropolis algorithm*, Bernoulli, 7 (2001), pp. 223–242, https://doi.org/10.2307/3318737.
[28] E. Hairer, S. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations* I *Nonstiff Problems*, 2nd ed., Springer Ser. Comput. Math. 8, Springer, Berlin, 1993, https://doi.org/10.1007/978-3-540-78862-1.
[29] A.-L. Haji-Ali, F. Nobile, L. Tamellini, and R. Tempone, *Multi-index stochastic collocation convergence rates for random PDEs with parametric regularity*, Found. Comput. Math., 16 (2016), pp. 1555–1605, https://doi.org/10.1007/s10208-016-9327-7.
[30] W. K. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, 57 (1970), pp. 97–109, https://doi.org/10.1093/biomet/57.1.97.
[31] S. Heinrich, *Multilevel Monte Carlo methods*, in Proceedings of the Third International Conference on Large-Scale Scientific Computing. Revised Papers, LSSC '01, Lecture Notes in Comput. Sci. 2179, Springer, Berlin, 2001, pp. 58–67, https://doi.org/10.1007/3-540-45346-6_5.
[32] D. Higdon, H. Lee, and Z. Bi, *A Bayesian approach to characterizing uncertainty in inverse problems using coarse and fine-scale information*, IEEE Trans. Signal Process., 50 (2002), pp. 389–399, https://doi.org/10.1109/78.978393.
[33] G. E. Hinton, *Training products of experts by minimizing contrastive divergence*, Neural Comput., 14 (2002), pp. 1771–1800, https://doi.org/10.1162/089976602760128018.
[34] V. H. Hoang, C. Schwab, and A. M. Stuart, *Complexity analysis of accelerated MCMC methods for Bayesian inversion*, Inverse Problems, 29 (2013), 085010, https://doi.org/10.1088/0266-5611/29/8/085010.
[35] H. Hoel, K. J. H. Law, and R. Tempone, *Multilevel ensemble Kalman filtering*, SIAM J. Numer. Anal., 54 (2016), pp. 1813–1839, https://doi.org/10.1137/15M100955X.
[36] M. D. Hoffman and A. Gelman, *The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo*, J. Mach. Learn. Res., 15 (2014), pp. 1593–1623, https://jmlr.org/papers/volume15/hoffman14a/hoffman14a.pdf.
[37] M. M. Ivanov, M. Simonović, and M. Zaldarriaga, *Cosmological parameters from the BOSS galaxy power spectrum*, J. Cosmol. Astropart. Phys., 2020 (2020), 042, https://doi.org/10.1088/1475-7516/2020/05/042.
[38] A. Jasra, K. Kamatani, K. Law, and Y. Zhou, *Bayesian static parameter estimation for partially observed diffusions via multilevel Monte Carlo*, SIAM J. Sci. Comput., 40 (2018), pp. A887–A902, https://doi.org/10.1137/17M1112595.
[39] A. Jasra, K. Kamatani, K. J. H. Law, and Y. Zhou, *Multilevel particle filters*, SIAM J. Numer. Anal., 55 (2017), pp. 3068–3096, https://doi.org/10.1137/17M1111553.
[40] A. Jasra, K. Kamatani, K. J. H. Law, and Y. Zhou, *A multi-index Markov chain Monte Carlo method*, Int. J. Uncertain. Quantif., 8 (2018), pp. 61–73, https://doi.org/10.1615/Int.J.UncertaintyQuantification.2018021551.

[41] J. Kaipio and E. Somersalo, *Statistical inverse problems: Discretization, model reduction and inverse crimes*, J. Comput. Appl. Math., 198 (2007), pp. 493–504, https://doi.org/10.1016/j.cam.2005.09.027.

[42] J. Liu and R. Chen, *Sequential Monte Carlo methods for dynamic systems*, J. Amer. Statist. Assoc., 93 (1998), pp. 1032–1044, https://doi.org/10.1080/01621459.1998.10473765.

[43] M. B. Lykkegaard, T. J. Dodwell, C. Fox, G. Mingas, and R. Scheichl, *Multilevel delayed acceptance MCMC*, SIAM/ASA J. Uncertain. Quantif., 11 (2023), pp. 1–30, https://doi.org/10.1137/22M1476770.

[44] M. B. Lykkegaard, G. Mingas, R. Scheichl, C. Fox, and T. J. Dodwell, *Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in PyMC*3, preprint, https://arxiv.org/abs/2012.05668, 2020.

[45] E. Marinari and G. Parisi, *Simulated tempering: A new Monte Carlo scheme*, Europhys. Lett. (EPL), 19 (1992), pp. 451–458, https://doi.org/10.1209/0295-5075/19/6/002.

[46] R. M. Neal, *Bayesian Learning for Neural Networks*, Lect. Notes Stat. 118, Springer, New York, 1996, https://doi.org/10.1007/978-1-4612-0745-0.

[47] P. J. E. Peebles, *The Large-Scale Structure of the Universe*, Princeton Ser. Phys., Princeton University Press, Princeton, 1980, https://doi.org/10.2307/j.ctvxrpz4n.

[48] B. Peherstorfer, K. Willcox, and M. Gunzburger, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, SIAM Rev., 60 (2018), pp. 550–591, https://doi.org/10.1137/16M1082469.

[49] B. D. Ripley, *Stochastic Simulation*, John Wiley & Sons, New York, 1987, https://doi.org/10.1002/9780470316726.

[50] G. O. Roberts and J. S. Rosenthal, *General state space Markov chains and MCMC algorithms*, Probability Surveys, 1 (2004), pp. 20–71, https://doi.org/10.1214/154957804100000024.

[51] G. O. Roberts and J. S. Rosenthal, *Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms*, J. Appl. Probab., 44 (2007), pp. 458–475, https://doi.org/10.1239/jap/1183667414.

[52] J. Salvatier, T. V. Wiecki, and C. Fonnesbeck, *Probabilistic programming in Python using PyMC*3, PeerJ Comput. Sci., 2 (2016), e55, https://doi.org/10.7717/peerj-cs.55.

[53] D. Spergel, N. Gehrels, J. Breckinridge, M. Donahue, A. Dressler, B. S. Gaudi, T. Greene, O. Guyon, et al., *Wide-Field InfraRed Survey Telescope-Astrophysics Focused Telescope Assets WFIRST-AFTA Final Report*, preprint, https://arxiv.org/abs/1305.5422, 2013.

[54] H. Strasser, *Mathematical Theory of Statistics: Statistical Experiments and Asymptotic Decision Theory*, De Gruyter, Berlin, New York, 1985, https://doi.org/10.1515/9783110850826.

[55] I. Sutskever and T. Tieleman, *On the convergence properties of contrastive divergence*, in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Chia Laguna Resort, Sardinia, Italy, Proc. Mach. Learn. Res. 9, Y. W. Teh and M. Titterington, eds., PMLR, 2010, pp. 789–795, https://proceedings.mlr.press/v9/sutskever10a.html.

[56] R. H. Swendsen and J.-S. Wang, *Replica Monte Carlo simulation of spin-glasses*, Phys. Rev. Lett., 57 (1986), pp. 2607–2609, https://doi.org/10.1103/PhysRevLett.57.2607.

[57] L. Tierney and A. Mira, *Some adaptive Monte Carlo methods for Bayesian inference*, Statistics in Medicine, 18 (1999), pp. 2507–2515, https://doi.org/10.1002/(SICI)1097-0258(19990915/30)18:17/18%3C2507::AID-SIM272%3E3.0.CO;2-J.