AGU ADVANCING EARTH AND SPACE SCIENCES

Water Resources Research

RESEARCH ARTICLE
10.1029/2025WR040933

A Novel Sequential Monte Carlo Algorithm for Parameter Estimation in Eco-Hydrological Models

Cong Xu[1], Kun Zhang[2], Gaofeng Zhu[1], Xin Li[3], and Chuang Wei[1]

[1]College of Earth and Environmental Sciences, Lanzhou University, Lanzhou, China, [2]School of Geospatial Engineering and Science, SunYat-SenUniversity, Zhuhai, China, [3]National Tibetan Plateau Data Center, State Key Laboratory of Tibetan Plateau Earth System, Environment and Resources, Institute of Tibetan Plateau Research, Chinese Academy of Sciences, Beijing, China

Abstract Bayesian inference offers a flexible framework for parameter estimation and uncertainty quantification in eco-hydrological models. However, simultaneously achieving robust posterior exploration and high computational efficiency for multimodal, high-dimensional, and computationally intensive targets remains challenging for the widely used Markov chain Monte Carlo (MCMC) and sequential Monte Carlo (SMC) methods. In this study, we developed the Parallel Adaptive Transition Particle Evolution Metropolis Sequential Monte Carlo (PATPEMS) algorithm, which is an adaptive and parallel SMC sampler for posterior distributions of model parameters in offline calibration. PATPEMS employs an adaptive sequence of intermediate distributions to control weight degeneracy and automatically select stages, a flexible scheduling of MCMC proposal kernels used to rejuvenate particles, together with reflection boundary handling to maintain particle diversity, and a particle-level parallelization scheme to exploit multicore architectures and reduce wall-clock time for computationally intensive models. Performance is assessed on four case studies: two synthetic targets probing multimodality and high-dimensional dependence, and a land surface model (LSM) with six parameters constrained by synthetic and real observations. Across all cases, PATPEMS provides close approximations to the target posteriors, judged against the analytic ground truth or reference solutions. For the LSM, parallelization yields substantial wall-clock speedups over the original non-parallel implementation. Compared with the original particle evolution Metropolis sequential Monte Carlo (PEM-SMC) algorithm, these results indicate that PATPEMS provides a more adaptive and parallel framework for robust Bayesian calibration of multimodal, correlated, and computationally demanding land surface and environmental models.

Plain Language Summary Land surface and eco-hydrological models are central to understanding water, energy, and carbon exchanges, but their growing complexity makes parameter calibration and uncertainty estimation increasingly difficult. Standard Bayesian samplers can struggle when posteriors are multimodal, strongly correlated, or expensive to evaluate. We introduce PATPEMS, a sequential Monte Carlo approach that improves practical reliability by adaptively constructing a sequence of intermediate distributions, flexibly combining complementary move kernels, and using reflection-based handling for bounded parameters. Its particle-level parallel design directly leverages multicore computing, reducing wall-clock time for likelihood-intensive models. Across two synthetic benchmarks and land-surface model calibrations with synthetic and real observations, PATPEMS produces posterior estimates consistent with analytic or reference solutions and demonstrates clear runtime benefits in the expensive model setting.

Key Points:

- PATPEMS is an adaptive parallel sequential Monte Carlo sampler for Bayesian parameter calibration
- Adaptive intermediate distributions and flexible move scheduling improve robustness on multimodal, correlated posterior targets
- Particle-level parallelism enables scalable inference for computationally intensive land surface and environmental models

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

G. Zhu,
zhugf@lzu.edu.cn

Citation:

Xu, C., Zhang, K., Zhu, G., Li, X., & Wei, C. (2026). A novel sequential Monte Carlo algorithm for parameter estimation in eco-hydrological models. *Water Resources Research*, *62*, e2025WR040933. https://doi.org/10.1029/2025WR040933

Received 29 APR 2025
Accepted 4 JUN 2026

Author Contributions:

Conceptualization: Kun Zhang, Xin Li
Funding acquisition: Gaofeng Zhu
Investigation: Kun Zhang
Software: Cong Xu, Chuang Wei
Supervision: Gaofeng Zhu, Xin Li
Writing – original draft: Cong Xu
Writing – review & editing: Kun Zhang, Gaofeng Zhu, Xin Li

1. Introduction

Land surface models (LSMs) are vital tools for understanding, projecting, and predicting the relationships between humans and the Earth system (Bonan & Doney, 2018; Lawrence et al., 2019). After three decades of development, LSMs have become more complex and comprehensive, often involving a large number of parameters to represent the properties of the system (Blyth et al., 2021; Fisher & Koven, 2020; Ricciuto et al., 2018). Therefore, how to choose a set of suitable parameters that gives a good fit to data is increasingly critical for improving model performance and identifying model structural limitations (Huang et al., 2016; Lu et al., 2018; Ricciuto et al., 2011). However, properly estimating (calibrating) model parameters is still recognized as a great challenge within the LSM community (Fisher & Koven, 2020).

In recent years, Bayesian inference has gained growing popularity in complex model parameter estimation, due to its flexibility in integrating multi-source data and its ability to rigorously quantify various uncertainties (Jeremiah

et al., 2011; Link & Barker, 2009; Pokhrel et al., 2008; Zyphur & Oswald, 2015). In environmental and eco-hydrological applications, Bayesian methods are used both for online state estimation in dynamical state-space models and for offline calibration problems formulated as static posterior parameter distributions, with the latter being the focus of this study. However, in most practical applications, these posteriors are analytically intractable and must be approximated using computational methods (Marin et al., 2012; Zhu et al., 2025). Among these methods, Kalman filters and smoothers that propagate ensembles of model states (e.g., EnKF/EnKS and their iterative variants) are widely used because they are extremely fast and straightforward to implement for large geophysical systems. However, in strongly nonlinear and non-Gaussian regimes, their linear Gaussian updating typically yields only an approximate representation of the posterior and may understate the uncertainty or multimodal structure (Annan et al., 2005; Evensen, 1994). Alternatively, Bayesian inference can be carried out with Monte Carlo samplers that approximate the posterior by a cloud of model evaluations, and thus accommodate more complex posterior shapes, albeit at a higher computational cost. Among such approaches, Markov chain Monte Carlo (MCMC) algorithms are the most widely used for parameter estimation and posterior approximation in environmental models (LeBauer et al., 2013; Liang et al., 2011; Metropolis et al., 1953). Numerous variants have been proposed to enhance sampling efficiency using single or multiple chains (Calderhead & Girolami, 2009; Hastings, 1970; Luengo et al., 2020). In particular, multi-chain differential evolution (DE) schemes such as differential evolution Markov chain (DE-MC) and differential evolution adaptive Metropolis (DREAM) often explore complex posteriors more effectively than single-chain methods (Ter Braak, 2006; Ter Braak & Vrugt, 2008; Vrugt, 2016; Vrugt et al., 2009). However, MCMC algorithms rely on the long-term behavior of Markov chains to reach the posterior distribution, and diagnosing convergence can be challenging in practice. Sequential Monte Carlo (SMC) samplers provide an alternative strategy for static Bayesian inference: a set of weighted particles is transported from an initial distribution through a sequence of intermediate targets to the desired posterior via importance reweighting, resampling, and rejuvenation moves (Doucet et al., 2000; Jeremiah et al., 2011; Zhu et al., 2018). SMC does not depend on Markov chain convergence and directly delivers a weighted posterior particle ensemble, but suffers from weight degeneracy and particle impoverishment, especially for sharp or high-dimensional targets (Li et al., 2014; Owen & Tribble, 2005).

Given these complementary strengths and weaknesses, hybrid algorithms that combine MCMC and SMC have attracted increasing attention. In the broader SMC literature, many such hybrid algorithms have been developed for state-space and trajectory inference, including the particle Markov chain Monte Carlo method and its interacting variants (Andrieu et al., 2010; Lindsten et al., 2014; Rainforth et al., 2016), particle filters that incorporate MCMC moves for multiple-target tracking (Khan et al., 2004), and particle filtering and smoothing methods (Doucet & Johansen, 2010; Gilks & Berzuini, 2001). In contrast, for static posterior distributions arising in offline calibration, hybridization has mainly taken the form of embedding MCMC rejuvenation kernels within SMC samplers. In most of the existing implementations, these kernels are local random walk Metropolis (RWM) or adaptive RWM updates (Chopin, 2002; Del Moral et al., 2006; Jeremiah et al., 2011, 2012), which can mix poorly on strongly multimodal or highly correlated posteriors and thus limit the efficiency of SMC in demanding calibration problems, highlighting the need for stronger, globally informed proposal mechanisms within SMC to efficiently address challenging multimodal and high-dimensional targets.

Within this line of work on static posteriors, we previously proposed the particle evolution Metropolis-sequential Monte Carlo (PEM-SMC) algorithm (Zhu et al., 2018). Inspired by the efficiency of DE MCMC schemes such as DE-MC and DREAM (Braak, 2006; Ter Braak & Vrugt, 2008; Vrugt et al., 2007), PEM-SMC incorporates a differential evolution proposal and random subspace crossover into the SMC framework to promote cross mode jumps and coordinate wise diversity. PEM-SMC has demonstrated strong performances on challenging posteriors in ecosystem, land surface, and soil hydraulic model calibration (Speich et al., 2021; Wang et al., 2023; Xu et al., 2024), making it a natural backbone for developing a more robust and scalable sampler for expensive environmental models. However, experience with PEM-SMC on computationally intensive models reveals several structural limitations that are typical of practical SMC implementations. First, a fixed sequence of intermediate distributions cannot adapt to case-dependent differences in posterior sharpness and dimensionality: large increments between intermediate distributions trigger weight collapse and frequent resampling, whereas small increments inflate the number of stages and waste computation, thereby degrading performance and efficiency. Second, the ordering of Markov move types is hard-wired: each stage applies a local RWM perturbation, a crossover style recombination step, and a differential evolution Metropolis (DE-MH) update. These three kernels play complementary roles—RWM explores local neighborhoods, crossover mixes information across

coordinates, and DE-MH proposes long-range, cross-mode jumps—so this hard-wired ordering may be far from optimal for targets with differing geometries and dependence structures. Third, the prior practice of discarding out-of-bounds candidates can lead to many rejected proposals when the posterior mass lies close to box constraints, thereby reducing proposal efficiency and acceptance. Finally, wall-clock time remains a practical bottleneck for expensive complex models with long single-run durations (e.g., multi-day LSM calibrations). Although SMC is naturally parallel due to per-particle independence (Geweke & Durham, 2011; Pattuk et al., 2016), the original PEM-SMC algorithm lacked a parallel implementation.

To address these limitations, we propose the Parallel Adaptive Transition Particle Evolution Metropolis Sequential Monte Carlo (PATPEMS) algorithm: an enhanced framework that emphasizes robustness and practicality through three refinements. (a) Parallelism (P) exploits concurrency in both the reweighting and moving step, yielding clear wall-clock speedups for models with expensive likelihood evaluations. (b) Adaptive transition (AT) designs the sequence of intermediate distributions so that increments are aligned with weight stability, thereby constraining stage growth, and improving sample efficiency. And (c) Flexible scheduling of move operators (PEM) enables problem-tailored ordering and repetition of RWM, crossover, and DE-MH operators, coupled with two boundary handlers, to preserve reversibility and reduce wasted proposals.

The remainder of this paper is organized as follows. Section 2 presents the methodological background and our algorithmic design, covering SMC, PEM-SMC, and the proposed PATPEMS algorithm. Section 3 describes the benchmark experiments with known targets, using two synthetic distributions to probe the multimodality and high-dimensional scalability. Section 4 presents two real-world case studies on LSM model parameter estimation with synthetic and real observations. Section 5 discusses the main findings and the scope of applicability. Section 6 concludes the work and outlines directions for future work.

2. Methods

2.1. Problem Formulation and Bayesian Framework

In Earth system models, uncertainties arise in both observations and simulations, and can be mathematically expressed as:

$$y_t = f(\mathbf{x}_t;\boldsymbol{\theta}) + \varepsilon_t, t = 1, 2, \ldots, n \tag{1}$$

where t is the time index, $y_t \in \mathbb{R}$ denotes the observation at time t, $f(\mathbf{x}_t;\boldsymbol{\theta})$ is the model output corresponding to the model forcing inputs $\mathbf{x}_t \in \mathbb{R}$, the d-dimensional parameter vector $\boldsymbol{\theta} \in \mathbb{R}^d$, and the residual error arising from both observation and model uncertainties $\varepsilon_t \in \mathbb{R}$. The goal of model parameter calibration is to estimate the parameter vector $\boldsymbol{\theta}$ such that the model outputs $f(\mathbf{x}_t;\boldsymbol{\theta})$ match the observations y_t as closely as possible.

Bayesian inference, grounded in Bayes' theorem, provides a rigorous and systematic framework for parameter estimation. Within a Bayesian framework, prior knowledge is combined with observed data to update the beliefs about the parameter vector $\boldsymbol{\theta}$. Let $\mathbf{y} = [y_1, y_2, \ldots, y_n]^\top \in \mathbb{R}^n$ denote the observation vector composed of the scalar observations at n time steps. The posterior distribution of the parameters is expressed as:

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\boldsymbol{\theta})\, p(\mathbf{y}|\boldsymbol{\theta})}{p(\mathbf{y})} \propto p(\boldsymbol{\theta})\, p(\mathbf{y}|\boldsymbol{\theta}) \tag{2}$$

where $p(\boldsymbol{\theta}|\mathbf{y})$ is the posterior distribution, representing the updated beliefs about the parameters after observing the data, $p(\boldsymbol{\theta})$ is the prior distribution, reflecting the initial beliefs about the parameters before observing the data, $p(\mathbf{y})$ is a normalizing constant that ensures that the posterior distribution integrates to 1, and $p(\mathbf{y}|\boldsymbol{\theta})$ is the likelihood function, describing the probability density of observing the data $\mathbf{y}$ for given values of the parameter vector $\boldsymbol{\theta}$. The likelihood function depends on the statistical characteristics of the residuals $\boldsymbol{\epsilon} = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n]^\top \in \mathbb{R}^n$, which captures the discrepancies between the model and the observed data.

2.2. Sequential Monte Carlo Sampler (SMC)

The posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$ is often challenging to compute analytically and difficult to sample directly from, especially in high-dimensional or multimodal contexts. SMC methods address these challenges by

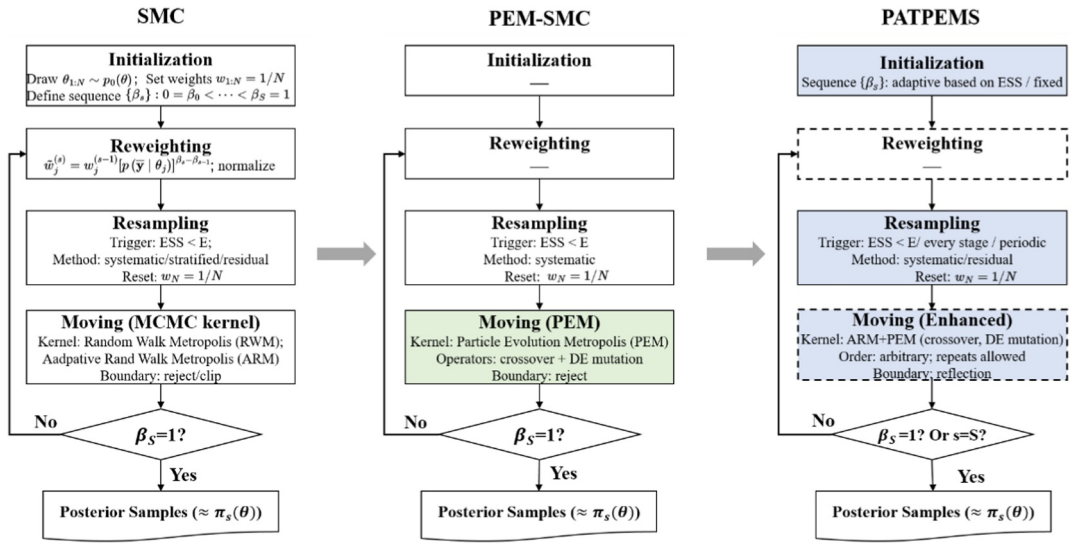

Figure 1. Flowcharts of the three SMC samplers considered in this study. The colored elements in the PEM-SMC and PATPEMS panels highlight the main modifications of the proposed algorithms relative to the baseline SMC framework. The dashed-outline boxes indicate steps executed in parallel; "—" denotes behavior identical to the SMC baseline; $\pi_s(\boldsymbol{\theta})$ denotes the posterior target distribution.

introducing a sequence of intermediate distributions and filtering particles from the prior distribution to the posterior distribution. Here, the SMC framework is employed for offline Bayesian inference of static model parameters. Thus, the term "sequential" refers to the progression through a sequence of intermediate bridging distributions from the prior to the posterior, rather than to physical-time evolution in a state-space filtering problem. Generally, the intermediate distribution is constructed using the geometric bridge method (Jeremiah et al., 2011, 2012):

$$\pi_s(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta})^{1-\beta_s} p(\boldsymbol{\theta}|\mathbf{y})^{\beta_s}, \quad s = 0, 1, \ldots, S \tag{3}$$

where $p_0(\boldsymbol{\theta})$, $\cdot\, \pi_s(\boldsymbol{\theta})$, and $p(\boldsymbol{\theta}|\mathbf{y})$ represent the initial, intermediate, and posterior distributions, respectively. β_s forms a monotonically increasing sequence ranging from 0 to 1, that is, $0 \le \beta_0 \le \beta_1 \le \cdots \le \beta_S = 1$. Hence $\pi_0(\boldsymbol{\theta}) = p_0(\boldsymbol{\theta})$ when $\beta_0 = 0$, and $\pi_S(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{y})$ when $\beta_S = 1$. This geometric bridge construction is similar to the temperature ladders used in parallel tempering MCMC (Earl & Deem, 2005; Miasojedow et al., 2013), which simulate multiple Markov chains at different inverse temperatures with occasional state swaps. Figure 1 gives an overview of the overall SMC framework (initialization, reweighting, resampling, and moving) and contrasts the three samplers introduced in Sections 2.2–2.4: SMC, PEM-SMC, and PATPEMS.

2.2.1. Initialization

The SMC sampler is initialized by drawing a population of N particles $\left\{\boldsymbol{\theta}_j^{(0)}\right\}_{j=1}^N$ from an initial sampling distribution $\pi_0(\boldsymbol{\theta}) = p_0(\boldsymbol{\theta})$. In practice, $p_0(\boldsymbol{\theta})$ is commonly (but not necessarily) chosen as the prior $p(\boldsymbol{\theta})$, and any distribution from which direct simulation is feasible can be used. Each particle is associated with a weight, and the initial weighted sample is represented as $\left\{\boldsymbol{\theta}_j^{(s)}, w_j^{(s)}\right\}_{j=1}^N$. Since at the stage $s = 0$ the target is $\pi_0(\boldsymbol{\theta})$ and particles are drawn i.i.d. from π_0, all the initial importance weights are equal, that is, $w_j^{(0)} = 1/N$. At each stage s, the weighted particle set $\left\{\left(\boldsymbol{\theta}_j^{(s)}, w_j^{(s)}\right)\right\}_{j=1}^N$ is updated through the standard reweight-resample-move cycle.

2.2.2. Reweighting

The reweighting process is essential for the convergence of the SMC framework. In this step, particle positions remain unchanged, that is, $\boldsymbol{\theta}_j^{(s)} = \boldsymbol{\theta}_j^{(s-1)}$, while their weights $w_j^{(s)}(j = 1,2,\ldots,N)$ are updated based on the density ratio between the current and previous intermediate targets:

$$w_j^{(s)} = w_j^{(s-1)} \frac{\pi_s\left(\boldsymbol{\theta}_j^{(s-1)}\right)}{\pi_{s-1}\left(\boldsymbol{\theta}_j^{(s-1)}\right)} = w_j^{(s-1)} \left[p\left(\mathbf{y}|\boldsymbol{\theta}_j^{(s-1)}\right)\right]^{\beta_s - \beta_{s-1}} \tag{4}$$

where the second quality follows from the geometric bridge $\pi_s(\boldsymbol{\theta}) \propto p(\boldsymbol{\theta})[p(\mathbf{y}|\boldsymbol{\theta})]^{\beta_s}$. This formula increases the weights of particles with higher density under $\pi_s(\boldsymbol{\theta})$ compared to $\pi_{s-1}(\boldsymbol{\theta})$ and decreases the weights of those with lower density. The updated weights $w_j^{(s)}(j = 1,2,\ldots,N)$ are then normalized so that $\sum_{j=1}^{N} w_j^{(s)} = 1$. Through reweighting, particles gradually align with the current intermediate distribution $\pi_s(\boldsymbol{\theta})$, giving greater prominence to those closer to the target.

2.2.3. Resampling

Repeated reweighting typically leads to particle degeneracy, where a small number of particles carry most of the weight and the remainder have a negligible influence on the posterior estimates. Resampling combats this by discarding low-weight particles and replicating high-weight ones, thereby restoring an effective population for the next stage. Degeneracy is monitored using the effective sample size,

$$\mathrm{ESS} = \left(\sum_{j=1}^{N} \left(w_j^{(s)}\right)^2\right)^{-1} \tag{5}$$

A common policy is to trigger resampling whenever ESS $< E$, with E chosen in the range of $0.3N - 0.8N$ (Fan et al., 2008). When resampling is invoked, the SMC sampler draws N particles from the empirical discrete distribution

$$\sum_{j=1}^{N} w_j^{(s)} \delta_{\theta_j^{(s)}} \tag{6}$$

where $\delta_{\boldsymbol{\theta}_j^{(s)}}$ denotes the Dirac point mass centered at particle $\boldsymbol{\theta}_j^{(s)}$. Thus, $\sum_{j=1}^{N} w_j^{(s)} \delta_{\boldsymbol{\theta}_j^{(s)}}$ represents the weighted empirical distribution of the current particle system. After resampling, all particle weights are reset to $1/N$. A variety of resampling schemes can be used. Among standard unbiased resampling schemes, multinomial resampling is the simplest but typically introduces the largest random fluctuation in the number of copies assigned to each particle during resampling (Doucet et al., 2000). Stratified and systematic resampling usually reduce this fluctuation at essentially the same computational cost, and systematic resampling is adopted as the default scheme in this study (Arulampalam et al., 2002). Residual resampling first allocates $\lfloor N w_j^{(s)} \rfloor$ copies deterministically and then applies low-variance sampling to the residual weights, which can further reduce resampling variance when the weight distribution is highly uneven.

2.2.4. Moving (MCMC Rejuvenation)

Resampling alleviates weight degeneracy but may also introduce sample impoverishment, that is, many replicated particles with reduced diversity. The move step is therefore used to rejuvenate particles under the current intermediate target π_s through a Markov chain Monte Carlo (MCMC) transition kernel K_s that leaves π_s invariant. This idea has been widely used in SMC-type samplers and related particle methods to restore particle diversity and improve mixing after resampling. For example, Particle-DREAM uses MCMC updates after particle resampling to mitigate particle degeneracy and sample impoverishment and thereby improve particle diversity

(Vrugt et al., 2013). In the present study, we refer to this operation generically as the move step within the SMC sampler.

Let $\left\{\left(\boldsymbol{\theta}_j^{(s)}, w_j^{(s)}\right)\right\}_{j=1}^{N}$ denote the particle population after reweighting and possible resampling. For each particle, a candidate $\boldsymbol{\theta}_p^{(s)}$ is drawn from a stage-specific proposal density $q_s\left(\cdot|\boldsymbol{\theta}_j^{(s)}\right)$ and accept it with the Metropolis-Hastings (M-H) probability:

$$\alpha\left(\boldsymbol{\theta}_j^{(s)}, \boldsymbol{\theta}_p^{(s)}\right) = \min\left\{1, \frac{\pi_s\left(\boldsymbol{\theta}_p^{(s)}\right) j_s\left(\boldsymbol{\theta}_j^{(s)}|\boldsymbol{\theta}_p^{(s)}\right)}{\pi_s\left(\boldsymbol{\theta}_j^{(s)}\right) j_s\left(\boldsymbol{\theta}_p^{(s)}|\boldsymbol{\theta}_j^{(s)}\right)}\right\} \tag{7}$$

where $j_s(\cdot|\cdot)$ denotes the proposal density used at stage s, which may vary across stages, especially for adaptive proposals. In the simplest case without boundary mappings, $j_s\left(\boldsymbol{\theta}_p^{(s)}|\boldsymbol{\theta}_j^{(s)}\right) = q_s\left(\boldsymbol{\theta}_p^{(s)}|\boldsymbol{\theta}_j^{(s)}\right)$ is the proposal density. If accepted, we set $\boldsymbol{\theta}_j^{(s)} \leftarrow \boldsymbol{\theta}_p^{(s)}$ otherwise, $\boldsymbol{\theta}_j^{(s)}$ remains unchanged. The weights are left unchanged because K_s is constructed to have π_s as its stationary distribution.

Common proposal choices q_s include: (i) a random-walk Metropolis (RWM) proposal with a symmetric d-variate Normal perturbation,

$$\boldsymbol{\theta}_p^{(s)} = \boldsymbol{\theta}_j^{(s)} + \boldsymbol{\eta}, \boldsymbol{\eta} \sim \mathcal{N}_d(\mathbf{0}, \gamma^2 \mathbf{I}_d), \tag{8}$$

that is,

$$j_s\left(\boldsymbol{\theta}_p^{(s)}|\boldsymbol{\theta}_j^{(s)}\right) = \mathcal{N}_d\left(\boldsymbol{\theta}_p^{(s)}; \boldsymbol{\theta}_j^{(s)}, \gamma^2 \mathbf{I}_d\right), \tag{9}$$

where $\boldsymbol{\eta} \in \mathbb{R}^d$ is a Gaussian perturbation, $\mathbf{0} \in \mathbb{R}^d$ is the zero vector, and $\mathbf{I}_d \in \mathbb{R}^{d\times d}$ is the d-dimensional identity matrix, and γ is a proposal scale parameter controlling the step size of the random-walk proposal. A practical choice is $\gamma = 2.38/\sqrt{d}$ for moderate dimensions (Ter Braak, 2006). (b) Adaptive random-walk Metropolis (ARM), which uses the empirical covariance of the previous population and thereby adapts to the posterior correlation structure while typically improving acceptance and mixing. Proposals falling outside the prior support are usually rejected. Applied after resampling, the move step counteracts sample impoverishment, enhances exploration of high posterior density regions, and maintains a well-mixed particle system at stage s.

2.3. Particle Evolution Metropolis SMC Sampler (PEM-SMC)

Compared with the baseline SMC, PEM-SMC keeps the initialization, reweighting, resampling, and the geometric-bridge sequence of intermediate distributions unchanged, but replaces the basic move step with particle evolution Metropolis (PEM), in which the crossover operator (XOVER) and the differential evolution proposal (DE-MH) are used to generate candidate particles that are then accepted or rejected through an MH step (Figure 1). This design targets the sample impoverishment that may persist after resampling and the short-step behavior of purely local random-walk proposals under the current intermediate target π_s, thereby improving particle diversity and posterior exploration efficiency.

1. Crossover operator (XOVER): At stage s, we select without replacement a parental pair $(\boldsymbol{\theta}_i^{(s)}, \boldsymbol{\theta}_j^{(s)})$ with $i \neq j$ from the current population, where $\boldsymbol{\theta}_i^{(s)}$ and $\boldsymbol{\theta}_j^{(s)}$ are d-dimensional parameter vectors, and generate an offspring pair $(\boldsymbol{\theta}_{i,p}^{(s)}, \boldsymbol{\theta}_{j,p}^{(s)})$. With a one-point crossover, we draw an index k uniformly from $\{1, \ldots, d\}$ and swap the coordinates to the right of k between the two parents:

$$\boldsymbol{\theta}_{i,\mathrm{p}}^{(s)} = \left(\boldsymbol{\theta}_i^{(s)}[1:k], \boldsymbol{\theta}_j^{(s)}[k+1:d]\right), \quad \boldsymbol{\theta}_{j,\mathrm{p}}^{(s)} = \left(\boldsymbol{\theta}_j^{(s)}[1:k], \boldsymbol{\theta}_i^{(s)}[k+1:d]\right) \tag{10}$$

The operation is applied with probability p_c. The offspring pair is then accepted according to the general Metropolis-Hastings (MH) rule:

$$r_c = \frac{\pi_s(\boldsymbol{\theta}_{i,p}^{(s)})\pi_s(\boldsymbol{\theta}_{j,p}^{(s)})j_s\left(\left(\boldsymbol{\theta}_i^{(s)},\boldsymbol{\theta}_j^{(s)}\right)|(\boldsymbol{\theta}_{i,p}^{(s)},\boldsymbol{\theta}_{j,p}^{(s)})\right)}{\pi_s\left(\boldsymbol{\theta}_i^{(s)}\right)\pi_s\left(\boldsymbol{\theta}_j^{(s)}\right)j_s\left((\boldsymbol{\theta}_{i,p}^{(s)},\boldsymbol{\theta}_{j,p}^{(s)})|\left(\boldsymbol{\theta}_i^{(s)},\boldsymbol{\theta}_j^{(s)}\right)\right)}, \quad \alpha_c = \min\{1, r_c\} \tag{11}$$

where $j_s(\cdot|\cdot)$ denotes the jump density for jumps between the parental and offspring pairs, including the selection probability of the parent pair and the probability of generating the particular offspring, for example, through the draw of k. Under the symmetric selection/generation scheme used here (uniform parent selection and uniform k), one has:

$$j_s\left((\boldsymbol{\theta}_{i,p}^{(s)},\boldsymbol{\theta}_{j,p}^{(s)})|\left(\boldsymbol{\theta}_i^{(s)},\boldsymbol{\theta}_j^{(s)}\right)\right) = j_s\left(\left(\boldsymbol{\theta}_i^{(s)},\boldsymbol{\theta}_j^{(s)}\right)|(\boldsymbol{\theta}_{i,p}^{(s)},\boldsymbol{\theta}_{j,p}^{(s)})\right) \tag{12}$$

Therefore, the proposal terms cancel, and the acceptance probability reduces to the standard Metropolis form, that is, a target density ratio. If the jump density were asymmetric, the general Metropolis-Hastings form in Equation 11 would apply. By recombining coordinates across particles, crossover injects coordinate-level variability into the particle population, which is especially beneficial in high dimensions and improves within-mode mixing and dimension-wise diversity.

2. Differential evolution proposal (DE-MH): To further increase particle diversity, we update a particle $\boldsymbol{\theta}_j^{(s)}$ using two distinct peer particle through a differential-evolution proposal mechanism. This operator is inspired by the differential-evolution Markov chain framework of Ter Braak (2006) and its subsequent development in DREAM-type samplers (Vrugt et al., 2009), and is also closely related to the differential-evolution-style MCMC rejuvenation used in Particle-DREAM (Vrugt et al., 2013). In PEM-SMC, the resulting differential-evolution proposal is written as:

$$\boldsymbol{\theta}_{j,p}^{(s)} = \boldsymbol{\theta}_j^{(s)} + \gamma\left(\boldsymbol{\theta}_{r_1}^{(s)} - \boldsymbol{\theta}_{r_2}^{(s)}\right) + \boldsymbol{\zeta}, \tag{13}$$

where r_1 and r_2 are distinct indices randomly selected from $\{1,\ldots,j-1,j+1,\ldots,N_p\}$. Following Ter Braak (2006), we set the DE jump rate to $\gamma = 2.38/\sqrt{2d}$, which gives near-optimal acceptance rates for DE-MC in d-dimensional Gaussian targets; we add a very small isotropic jitter $\boldsymbol{\zeta} \sim \mathcal{N}_d(\mathbf{0}, b\mathbf{I}_d)$, with $b = 10^{-6}$, to avoid proposal degeneracy; and both γ and the jitter variance are exposed as tunable options in the new algorithm. The proposed candidate is then accepted according to the general Metropolis-Hastings (MH) rule:

$$r_m = \frac{\pi_s(\boldsymbol{\theta}_{j,p}^{(s)})j_s\left(\boldsymbol{\theta}_j^{(s)}|\boldsymbol{\theta}_{j,p}^{(s)}\right)}{\pi_s\left(\boldsymbol{\theta}_j^{(s)}\right)j_s\left(\boldsymbol{\theta}_{j,p}^{(s)}|\boldsymbol{\theta}_j^{(s)}\right)}, \quad \alpha_m = \min\{1, r_m\} \tag{14}$$

where $j_s(\cdot|\cdot)$ denotes the jump density of the differential evolution proposal, including peer selection and the probability of generating the particular Gaussian jitter. Under symmetric peer selection and Gaussian jitter, the forward and reverse transfer probabilities are equal, so the proposal terms cancel, and the acceptance probability reduces to the standard Metropolis form, that is, a target-density ratio. If the jump density were asymmetric, the general Metropolis-Hastings form in Equation 14 would apply. Because the difference vector $(\boldsymbol{\theta}_{r_1}^{(s)} - \boldsymbol{\theta}_{r_2}^{(s)})$ captures population directions of high variation, the differential-evolution proposal naturally produces anisotropic, scale-adaptive jumps that reduce random walk behavior and facilitate mode-to-mode moves, thereby enhancing global exploration.

Together, the crossover (dimension-wise recombination) and differential-evolution proposal (population-difference jumps) form the PEM proposal. When coupled with the M-H accept/reject step, they improve both the within-mode mixing and cross-mode mobility under π_s, thereby increasing particle diversity and mixing efficiency. In this sense, PEM-SMC incorporates proposal ideas related to differential-evolution-based population MCMC methods into the move step of an SMC sampler for static posterior inference.

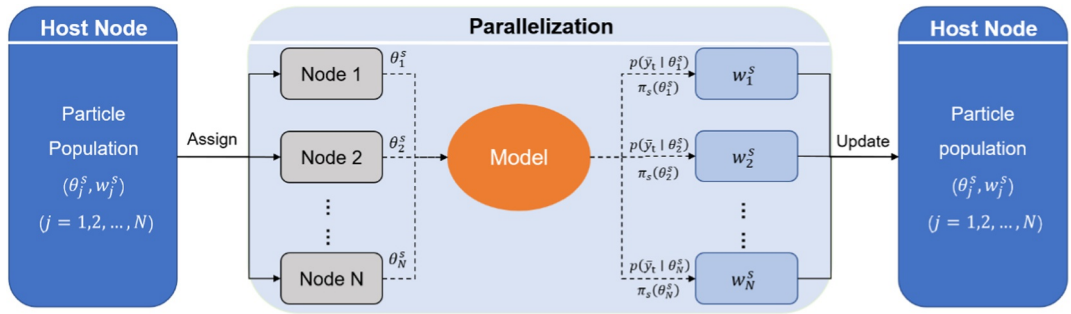

Figure 2. Parallel diagram of the reweighting step in the PATPEMS algorithm.

2.4. PATPEMS: Parallel Evaluation, Adaptive Transition, and Enriched Moves

Compared with the original PEM-SMC algorithm, PATPEMS introduces three main enhancements (Figure 1). First, the likelihood evaluations during both weighting and move steps are parallelized to better exploit multi-core hardware. Second, an adaptive conditional effective sample size (CESS)-based scheme constructs the sequence of intermediate distributions while controlling stage growth. Third, a more flexible configuration of local and global move kernels, together with reflective boundary handling near box constraints, improves robustness under constrained posteriors.

These modifications are introduced within the same general SMC framework described above, in which particles are gradually transported from the prior distribution to the posterior distribution through a geometric-bridge sequence of intermediate targets. Related hybrid particle MCMC methods, including Particle-DREAM, have also used MCMC rejuvenation to improve particle diversity after resampling (Vrugt et al., 2013). In PATPEMS, however, the main extension lies in integrating parallel likelihood evaluation, adaptive transition of intermediate distributions, enriched move operators, and reflective boundary treatment into a unified SMC sampler for computationally intensive static parameter estimation.

2.4.1. Parallel Implementation for Computationally Intensive Models

The two main computational hot spots in SMC-type samplers are the reweighting step (likelihood evaluation for the current particles) and the move step (likelihood evaluation for newly proposed candidates). In PATPEMS, we exploit the fact that these evaluations are independent across particles and use a simple host-worker scheme to carry them out in parallel.

1. Parallelization of the reweighting step: because the reweighting step is evaluated for each particle independently, we execute it in parallel using multiple compute nodes (e.g., CPU cores). At each stage s, the host partitions the current population $\left\{\left(\boldsymbol{\theta}_j^{(s)}, w_j^{(s)}\right)\right\}_{j=1,\ldots,N_p}$ across the available compute nodes (e.g., CPU cores). On each node, the assigned particles are passed to the model to generate simulated outputs $f\left(\mathbf{x}_t;\boldsymbol{\theta}_j^{(s)}\right)$ and the corresponding likelihood values $p\left(\mathbf{y}|\boldsymbol{\theta}_j^{(s)}\right)$. Each node then sends its local results back to the host process. The host updates and normalizes the weights to form the global set $\left\{w_j^{(s)}\right\}$ (Figure 2). To avoid race conditions caused by concurrent model execution and file I/O, each worker runs in a dedicated working directory that is removed after completion.
2. Parallelization of the move step: Candidate particles generated by the selected sequence of move kernels (ARM and PEM operators) are also evaluated on the worker nodes (Figure 3). Each worker receives a batch of candidates $\{\boldsymbol{\theta}_{j,p}^{(s)}\}$, calls the model to obtain simulated outputs $f\left(\mathbf{x}_t;\boldsymbol{\theta}_{j,\mathrm{p}}^{(s)}\right)$ and likelihood values $p\left(\mathbf{y}|\boldsymbol{\theta}_{j,\mathrm{p}}^{(s)}\right)$, and applies the M-H acceptance test locally under the current target π_s. Only the accepted updates are sent back to the host to refresh the population. Conceptually, each particle is updated independently by a local M-H step and then written back to the population.

For realistic hydrologic and Earth system models, likelihood evaluation dominates runtime. The proposed parallelization reduces wall-clock time almost linearly with the number of available cores until model I/O or memory

becomes the bottleneck. The scheme is agnostic to the adaptive schedule, requires minimal interprocess communication, and preserves the statistical correctness of the sampler. In practice, parallel reweighting and moving make PATPEMS viable for computationally intensive parameter estimation where serial SMC/PEM-SMC would be prohibitively slow.

2.4.2. Adaptive Transition of Intermediate Distributions

A fixed exponential sequence $\{\beta_s\}_{s=0}^{S}$ is simple but does not adapt to the difficulty of the target distribution. When the likelihood is sharp, large fixed increments may place successive intermediate distributions too far apart and create weight degeneracy; when it is flat, the same increments make them unnecessarily close and inflate the number of stages. To mitigate these issues, PATPEMS adopts an adaptive schedule that chooses $\beta_{s+1} > \beta_s$ at every stage so that the importance weights remain stable and the total number of stages remains moderate. Let $\Delta\beta = \beta_{s+1} - \beta_s$. With the current normalized weights,

$$\left\{w_i^{(s)}\right\}_{i=1}^{N_p}\left(\sum_{i=1}^{N_p} w_i^{(s)} = 1\right) \tag{15}$$

and incremental factors $u_i(\Delta\beta) = \exp\{\Delta\beta\ell_i\}$ where $\ell_i = \log p\left(\mathbf{y}|\boldsymbol{\theta}_i^{(s)}\right)$, the conditional effective sample size (CESS) is:

$$\mathrm{CESS}(\Delta\beta) = \frac{\left(\sum_{i=1}^{N_p} w_i^{(s)} u_i(\Delta\beta)\right)^2}{\sum_{i=1}^{N_p} w_i^{(s)} u_i^2(\Delta\beta)} \tag{16}$$

The relative measure $R_c = \mathrm{CESS}(\Delta\beta)/N_p$ reduces to the current ESS/N_p when $\Delta\beta = 0$. At each stage, we select $\Delta\beta$ so that $\tau_{\mathrm{low}} \le R_c \le \tau_{\mathrm{high}}$. Because R_c decreases monotonically in $\Delta\beta$ whenever $\mathrm{Var}_w(\ell) > 0$, we first use the variance approximation $R_c \approx \exp\{-\mathrm{Var}_w(\ell)\Delta\beta^2\}$ to obtain a predicted step size $\Delta\beta_{\mathrm{pred}}$ at the geometric mid-point of the target band:

$$\Delta\beta_{\mathrm{pred}} = \sqrt{\frac{-\log\sqrt{\tau_{\mathrm{low}}\tau_{\mathrm{high}}}}{\max(\mathrm{Var}_w(\ell), \varepsilon)}} \tag{17}$$

To avoid excessively small or large changes in the predicted increment, it is then restricted by three simple bounds:

$$\Delta\beta_{\min} \le \Delta\beta \le \Delta\beta_{\max}, \quad \Delta\beta \le G \times \Delta\beta_{\mathrm{pred}} \tag{18}$$

where G represents the growth factor, which prevents sudden step inflation across stages; $\Delta\beta$ is kept between the user-specified lower and upper limits $\Delta\beta_{\min}$ and $\Delta\beta_{\max}$ (defaults: 10^{-5} and 0.1), and it is not allowed to grow faster than. If the resulting step still does not place R_c within the band $[\tau_{\mathrm{low}}, \tau_{\mathrm{high}}]$, we run a short monotone bisection search on the interval $[\Delta\beta_{\min}, \Delta\beta_{\max}]$ to find the largest $\Delta\beta$ with $R_c \ge \pi_{\mathrm{low}}$ and then nudge it toward the interior of the band. This R_c band schedule yields steps that are large enough to make progress and small enough to avoid collapse, automatically shortening stages in sharp regions and lengthening them in diffuse regions, while the guard set $(\Delta\beta_{\min}, \Delta\beta_{\max}, G)$ provides robust, problem-agnostic safety rails.

2.4.3. Flexible Particle Movement Strategy and Boundary Handling

In PEM-SMC, the loss of particle diversity after resampling is alleviated by a fixed move step that rejuvenates the population using three MCMC operators: a local random-walk Metropolis (RWM) update together with the two PEM operators, namely crossover and differential-evolution proposal. PATPEMS retains this general rejuvenation idea, which is also consistent with related hybrid particle-MCMC methods that use post-resampling MCMC updates to improve particle diversity, but further enriches and reorganizes the move stage for static

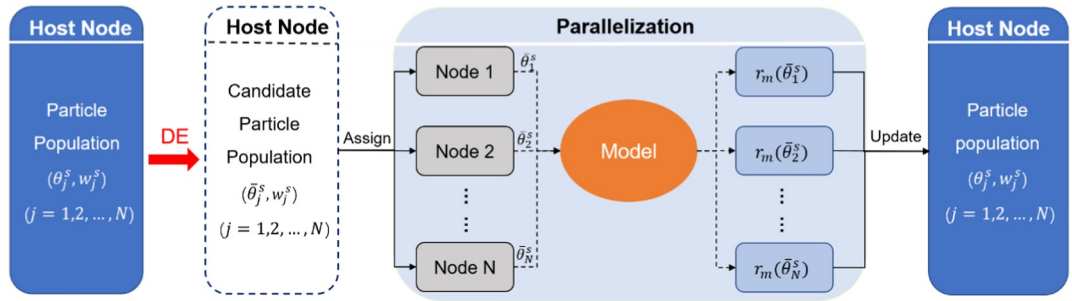

Figure 3. Parallel diagram of the differential-evolution proposal step in the PATPEMS algorithm.

parameter inference. Specifically, RWM is replaced by an adaptive random-walk Metropolis (ARM) kernel that learns covariance information from the current particle population, leading to better local scaling and typically improved acceptance and mixing near high-posterior-density regions. Because the three operators act through different mechanisms—differential-evolution proposal promotes long-range jumps through population differences, crossover enhances dimension-wise diversity through recombination, and ARM provides local refinement—PATPEMS allows the execution order to be user-defined and permits repetitions of any operator within a stage. This flexibility improves the ability of the sampler to explore multimodal and anisotropic posterior distributions while maintaining particle diversity.

When the population is near the prior boundary, the ARM and DE-MH proposals may step outside the prior box $\boldsymbol{B} = \prod_{q=1}^{d} [\theta_{\min,q}, \theta_{\max,q}]$, where subscript q denotes the coordinate index. The original PEM-SMC algorithm handled such cases by rejecting and redrawing within bounds (Figure 4a), which can lead to repeated proposal attempts and reduce proposal efficiency near the boundaries. In PATPEMS, we instead introduce a deterministic reflection map that sends any out-of-bounds proposals back into B, thereby reducing repeated proposal attempts while retaining a valid Metropolis-Hastings transition under bounded priors (Figure 4b), following the reflection-based boundary treatment discussed in the DREAM literature (Vrugt, 2016; Vrugt & Ter Braak, 2011). This treatment is introduced mainly to provide a theoretically cleaner and more consistent bounded-domain handling for the M-H proposals. We define the coordinate wise reflection operator T_{refl} , which maps any unconstrained candidate vector back into the bounded prior box $\boldsymbol{B}$, as $T_{\text{refl}} : \mathbb{R}^d \to \boldsymbol{B}$. For the qth coordinate, let the interval width be $w_q = \theta_{\max,q} - \theta_{\min,q}$. Given an unconstrained candidate $\tilde{\boldsymbol{\theta}}_{j,\text{p}}^{(s)}$, where j denotes the particle index, p indicates the proposed candidate, and q denotes the coordinate index, the reflected qth coordinate is obtained as:

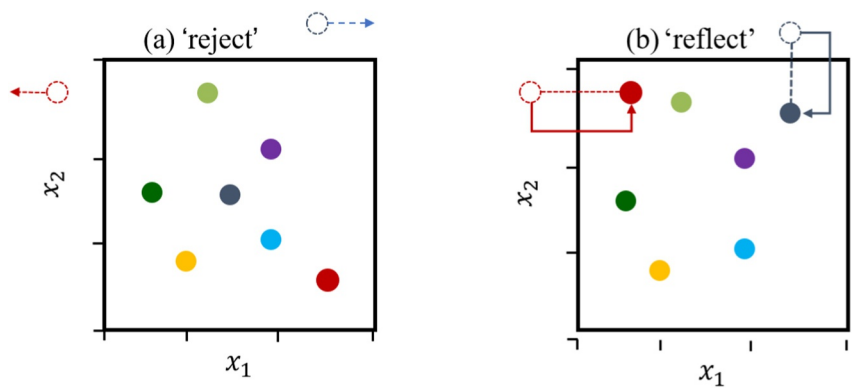

Figure 4. Boundary handling for out-of-bounds proposals in PATPEMS: (a) "reject" (redraw): proposals outside the prior box are rejected and redrawn within bounds; (b) "reflect": the exceeded coordinate is mirrored at the boundary via the triangular wave reflection map Equation 19, avoiding wasted proposals and improving mixing near the edges of the box. The reflection schematic in panel (b) is adapted from Figure 6 in Vrugt (2016).

$$\theta^{(s)}_{j,p,q} = \theta_{\min,q} + \rho\left(\tilde{\theta}^{(s)}_{j,p,q} - \theta_{\min,q}; w_q\right) \tag{19}$$

Applying this mapping independently for $q = 1, \ldots, d$ yields the reflected vector $T_{\text{refl}}\left(\tilde{\boldsymbol{\theta}}^{(s)}_{j,p}\right) \in \boldsymbol{B}$. Here, we use a coordinate-wise triangular-wave reflection operator, following the reflection construction described by Vrugt (2016), defined as:

$$\rho(u; w) = w - \{|\text{mod}\,(u, 2w) - w| \in [0, w]\} \tag{20}$$

where mod $(\cdot, \cdot)$ denotes the modulo operator. This triangular-wave expression is mathematically equivalent to repeatedly applying the single-reflection rule until the coordinate re-enters the interval $[\theta_{\min,q}, \theta_{\max,q}]$. Here, it provides a reversible and practically convenient coordinate-wise mapping from $\mathbb{R}^d$ back to the bounded prior box $\boldsymbol{B}$. As further shown by a dedicated boundary-active diagnostic test Text S1 in Supporting Information S1, the reflection treatment yields higher acceptance rates for both the ARM and DE-MH moves than reject-and-redraw, while leading to a very similar posterior approximation. These results indicate that its main practical benefit lies in more effective and theoretically cleaner handling of boundary-crossing proposals, rather than in materially altering the final posterior approximation.

Having specified the mappings, we next clarify that it is compatible with a reversible Metropolis-Hastings move. Starting from the current particle $\boldsymbol{\theta}^{(s)}_j$, we first generate an uncontrained proposal

$$\tilde{\boldsymbol{\theta}}^{(s)}_{j,p} = \boldsymbol{\theta}^{(s)}_j + \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim g \tag{21}$$

where g is a base increment density symmetric about zero, that is, $g(\boldsymbol{\epsilon}) = g(-\boldsymbol{\epsilon})$, for example, the ARM proposal with stage-wise frozen covariance or the DE-MH proposal with symmetric jitter. If $\tilde{\boldsymbol{\theta}}^{(s)}_{j,p} \notin B$, we map it back into B coordinate-wise using the reflection map T_{refl}, yielding the in-box proposal

$$\boldsymbol{\theta}^{(s)}_{j,p} = T_{\text{ref}}\left(\tilde{\boldsymbol{\theta}}^{(s)}_{j,p}\right) \tag{22}$$

The resulting stage-s proposal density is therefore the induced density:

$$j_s\left(\boldsymbol{\theta}^{(s)}_{j,p} | \boldsymbol{\theta}^{(s)}_j\right) = \int \delta\left(\boldsymbol{\theta}^{(s)}_{j,p} - T_{\text{ref}}\left(\boldsymbol{\theta}^{(s)}_j + \boldsymbol{\epsilon}\right)\right) g(\boldsymbol{\epsilon})\, d\boldsymbol{\epsilon} \tag{23}$$

This reflected-proposal construction follows the reversibility arguments developed for DREAM boundary handling in prior work (Vrugt, 2016; Vrugt & Ter Braak, 2011), and is adopted here to handle bounded parameter domains within the PATPEMS move step. For the reflection operator, T_{refl} is involutive and measure-preserving, with unit Jacobian almost everywhere. The formal derivations and the corresponding one-dimensional forward reverse pairing argument are provided in Text S2 of Supporting Information S1. Since g is symmetric and T_{refl} induces a reversible reflected proposal, we obtain:

$$j_s\left(\boldsymbol{\theta}^{(s)}_{j,p} | \boldsymbol{\theta}^{(s)}_j\right) = j_s\left(\boldsymbol{\theta}^{(s)}_j | \boldsymbol{\theta}^{(s)}_{j,p}\right) \tag{24}$$

Accordingly, the proposal terms in Equation 5 cancel, and the Metropolis–Hastings acceptance probability simplifies to:

$$\alpha\left(\boldsymbol{\theta}^{(s)}_j, \boldsymbol{\theta}^{(s)}_{j,p}\right) = \min\left\{1, \frac{\pi_s\left(\boldsymbol{\theta}^{(s)}_{j,p}\right)}{\pi_s\left(\boldsymbol{\theta}^{(s)}_j\right)}\right\} \tag{25}$$

This result holds whether or not a boundary reflection occurs. Hence, the move step leaves π_s invariant while avoiding wasteful redraws at the boundary.

2.4.4. Implementation and Practical Guidance

Before turning to implementation details, we briefly summarize the convergence properties of PATPEMS. Under mild regularity conditions on the prior, likelihood, and proposal kernels, and for a reasonable intermediate sequence $\{\beta_s\}_{s=0}^{S}$, the algorithm produces consistent particle approximations to the bridge sequence $\{\pi_s\}_{s=0}^{S}$. Specifically, for each fixed stage s, the empirical measure of the particles converges to π_s as the number of particles increases. In particular, at the terminal stage $s = S$, the particle system converges to the posterior distribution $p(\boldsymbol{\theta}|\mathbf{y})$. A formal statement of this result, together with a proof sketch based on standard SMC arguments, is given in Text S3 of Supporting Information S1.

In the remainder of this subsection, we translate these theoretical requirements into practical choices of computing environment, default hyperparameters, and tuning strategies for the intermediate-distribution sequence, resampling, and move steps. A complete configuration table with typical ranges is provided in Text S4 of Supporting Information S1.

PATPEMS is implemented in MATLAB and uses the Parallel Computing Toolbox. All simulations in this study were run at the Supercomputing Center of Lanzhou University, where jobs were distributed over multi-core CPU nodes by the system scheduler. The implementation runs on recent MATLAB versions (MATLAB R2018 or later recommended) under Linux and Windows and can be used on either a single workstation or a high-performance computing (HPC) cluster. The MATLAB implementations of the PATPEMS and MT-DREAM$_{(ZS)}$ algorithms used in case studies 1–4 are openly available on Zenodo (https://doi.org/10.5281/zenodo.17546472) for reproducibility and reuse.

For most calibration problems, PATPEMS can be used with a simple default configuration. Users specify parameter bounds that define a uniform prior for each parameter and the likelihood function, and implement a routine that evaluates the log-likelihood (or log-posterior) within these bounds. A particle number N_p in the range of 500–2000, together with a generous stage budget (e.g., $S = 1{,}000$), is usually sufficient for an initial run, since S acts only as an upper limit for the adaptively chosen β sequence. The default adaptive β schedule, ESS-triggered resampling, and the standard mixture of ARM, DE-MH, and crossover moves are recommended as the initial configuration without modification. Parallel execution can be enabled whenever multiple CPU cores are available. A complete list of hyperparameters, default values, suggested ranges, and tuning guidance based on diagnostic symptoms is given in Text S4 of Supporting Information S1.

The pseudocode of the PATPEMS algorithm is given below.

Input: prior $p_o(\boldsymbol{\theta})$; likelihood $p(\mathbf{y}|\boldsymbol{\theta})$; hyperparameters N_p , S , CESS_{target}; resampling policy and method; move-operator sequence (order and repeats); boundary handler; parallel settings.

Initialization: draw $\boldsymbol{\theta}_i^{(0)} \sim p_0(\boldsymbol{\theta})$ for $j = 1, \ldots, N_p$; set $w_j^{(0)} = 1/N_p$; $\beta_0 = 0$ and $s = 0$

while $(\beta_s < 1)$ and $(s < S)$ do

Adaptive intermediate distribution sequence (CESS-guided)

Choose $\Delta\beta_s$ by monotone search such that $\mathrm{CESS}(\Delta\beta_s) \approx \mathrm{CESS}_{target}$

Set $\beta_{s+1} = \min(\beta_s + \Delta\beta_s, 1)$.

Reweighting (parallelizable across particles)

for $j = 1, \ldots, N_p$ (in parallel) do

$$w_j^{(s+1)} \leftarrow w_j^{(s)} \left[p\left(\mathbf{y}|\boldsymbol{\theta}_j^{(s)}\right) \right]^{\Delta\beta_s}$$

end for

Normalize the weights so that $\sum_{j=1}^{N_p} w_j^{(s+1)} = 1$

Resampling (triggered by ESS or periodic policy)

If resampling is triggered, then

Resample $\left\{ \left(\boldsymbol{\theta}_j^{(s)}, w_j^{(s+1)} \right) \right\}_{j=1}^{N_p}$ using the chosen method

Set $w_j^{(s+1)} = 1/N_p$ for all j.
end if
# Move (user-defined operator order and optional repeats)
for op in the user-specificed move-operator sequence {DE-MH, XOVER, ARM}, with optional repeats, do
for $i = 1, \ldots, N_p$ (in parallel) do
Propose a candidate $\boldsymbol{\theta}_{j,p}^{(s)}$ from $\boldsymbol{\theta}_j^{(s)}$ using operator op.
Apply the boundary handler to obtain an in-box candidate if needed.
Evaluate likelihood $p(\mathbf{y}|\boldsymbol{\theta}_{j,p}^{(s)})$
Define the current intermediate target: $\pi_s(\boldsymbol{\theta}) \propto p(\mathbf{y}|\boldsymbol{\theta})^{\beta_s} p_0(\boldsymbol{\theta})$
Compute M-H ratio: $r = \min\left\{1, \frac{\pi_s(\boldsymbol{\theta}_{j,p}^{(s)})}{\pi_s(\boldsymbol{\theta}_j^{(s)})}\right\}$
Set the acceptance probability $\alpha = r$.
if $U(0,1) \leq \alpha$ then $\boldsymbol{\theta}_j^{(s)} \leftarrow \boldsymbol{\theta}_{j,p}^{(s)}$; else keep $\theta_j^{(s)}$ unchanged.
end for
end for
Update the stage index: $s \leftarrow s + 1$.
end while
if $\beta_s < 1$ then
Terminate because the maximum number of stages S has been reached;
increase S and rerun the algorithm.
end if

3. Benchmark Experiments With Known Targets

In this section, we describe how we evaluated the ability of PATPEMS to recover complex posterior distributions under controlled conditions with known analytic targets. The benchmark set included strong multimodality and high dimensionality. Specifically, we considered: (a) a two-dimensional 20-mode Gaussian mixture, and (b) a 100-dimensional asymmetric bimodal Gaussian mixture. For each target, PATPEMS was run multiple times with different random seeds to assess the robustness of the diagnostics and posterior estimates. Table S3 in Supporting Information S1 summarizes the PATPEMS settings used across these benchmarks, and all the performance metrics are reported as averages over repeated runs together with measures of variability.

We quantify the fidelity of PATPEMS to known posteriors using two complementary distances and a small set of algorithmic diagnostics. The marginal 2-Wasserstein distance (W_2) between the reference (F) and estimated (G) marginal cumulative distribution functions (CDFs) is computed by quantile matching on a common probability grid; for a scalar parameter,

$$W_2(F,G) = \left(\int_0^1 \left[F^{-1}(p) - G^{-1}(p) \right]^2 dp \right)^{1/2} \tag{26}$$

where $p \in [0,1]$ denotes the quantile level. In practice, we approximate W_2 on a discrete grid $\{p_k\}_{k=1}^K \subset (0,1)$ with $p_k = k/(K+1)$ and typically $K = 512$, by matching the empirical quantiles $F^{-1}(p_k)$ and $G^{-1}(p_k)$ and numerically integrating over the grid. For vector-valued parameters $\boldsymbol{\theta}$, we optionally standardize each coordinate before computing the marginal W_2 values for coordinate q, so that distances are comparable across coordinates. We then average the marginal W_2 values over $q = 1, \ldots, d$ and report the mean values as a single summary, together with 95% confidence intervals across runs. Smaller W_2 indicates closer alignment, and $W_2 = 0$ implies identical marginal distributions.

To further assess agreement in location and spread across parameters, we compute a marginal moment alignment index D_s. For a d-dimensional parameter vector, let $\mu_{q,r}$ and $\sigma_{q,r}$ denote the reference marginal mean and standard deviation of coordinate q, and let $\mu_{q,s}$ and $\sigma_{q,s}$ denote the corresponding sample-based estimates from the posterior samples, where the subscript r stands for "reference" and the subscript s stands for "sample." We define:

$$D_s = \left[\frac{1}{2d}\sum_{q=1}^{d}\left\{\left(\frac{\mu_{q,r}-\mu_{q,s}}{\sigma_{q,r}}\right)^2+\left(\frac{\sigma_{q,r}-\sigma_{q,s}}{\sigma_{q,r}}\right)^2\right\}\right]^{1/2} \tag{27}$$

This index summarizes how well the marginal means and standard deviations are recovered, with smaller values indicating better agreement; $D_s = 0$ denotes perfect agreement. For single-coordinate summaries, we use the same expression with $d = 1$ and the sum restricted to the coordinate of interest.

Algorithmic diagnostics focus on the adaptive sequence of intermediate distributions $\{\beta_s\}_{s=0}^{S}$ and the PATPEMS particle system with its rejuvenation moves. Along the evolution path, we examine the trajectory of β_s together with the effective sample size ESS, computed from the normalized importance weights, and the conditional ESS (CESS), computed from the incremental unnormalized weights, to assess weight degeneracy and the aggressiveness of each intermediate step. We also track acceptance rates for the two rejuvenation kernels—random walk Metropolis-Hastings (ARM) and differential evolution Metropolis-Hastings (DE-MH)—as well as the number of resampling events triggered by ESS/CESS thresholds, as compact summaries of mixing and degeneracy. Finally, we compute the cumulative log-evidence, $\log Z$, as the running sum of the stage-wise log normalizing constant estimates. A smooth, nearly monotonic trajectory indicates a stable sequence of intermediate distributions, whereas sharp jumps (often accompanied by ESS/CESS collapses) signal overly aggressive updates of the bridging exponent or numerical instability.

3.1. Case Study 1: A Two-Dimensional Probability Distribution With 20 Modes

The first case study considered a two-dimensional target density $\pi(\mathbf{x})$, defined as the 20-component Gaussian mixture used by Liang and Wong (2001):

$$\pi(\mathbf{x}) = \sum_{i=1}^{20}\frac{\omega_i}{2\pi\sigma_i^2}\exp\left\{-\frac{1}{2\sigma_i^2}(\mathbf{x}-\boldsymbol{\mu}_i)^{\mathrm{T}}(\mathbf{x}-\boldsymbol{\mu}_i)\right\} \tag{28}$$

where $\mathbf{x} = (x_1, x_2)^{\mathrm{T}} \in [-1, 10]^2$, $\sigma_1 = \sigma_2 = \cdots = \sigma_{20} = 0.1$, $\omega_1 = \omega_2 = \cdots = \omega_{20} = 0.05$, and $(\cdot)^{\mathrm{T}}$ denotes transpose. The 20 mean vectors $\boldsymbol{\mu}_i \in \mathbb{R}^2$ are:

$$(\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_{20}) = \begin{pmatrix} 2.18 & 8.67 & 4.24 & 8.14 & 3.93 & 3.25 & 1.70 & 4.59 & 6.91 & 6.87 \\ 5.76 & 9.59 & 8.48 & 1.68 & 8.82 & 3.47 & 0.50 & 5.60 & 5.81 & 5.40 \\ 5.41 & 2.70 & 4.98 & 1.14 & 8.33 & 4.93 & 1.83 & 2.26 & 5.54 & 1.69 \\ 2.65 & 7.88 & 3.70 & 2.39 & 9.50 & 1.50 & 0.99 & 0.31 & 6.86 & 8.11 \end{pmatrix}^{T} \tag{29}$$

Most local modes in this 2-D 20 component Gaussian mixture are separated by more than 15 standard deviations, making it a stringent test for multimodal inference. In our numerical experiment, we restricted the state space to the square domain $[-1,10]^2$ and adopted a uniform prior over this bounded domain. For PATPEMS, the default hyperparameter settings were used in this case study, as listed in Text S4.3 of Supporting Information S1.

Figure 5 presents algorithmic diagnostics for a representative PATPEMS run on the 2-D 20-mode Gaussian mixture (RNG seed = 1): the sequence of bridging distributions β_s increases smoothly to 1; pre-resampling ESS and relative CESS remain near the target under the chosen resampling policy; acceptance rates for ARM, XOVER, and DE-MH decline as annealing proceeds; and the cumulative log-evidence $\log Z$ moves in an almost monotonic fashion without abrupt jumps, indicating a numerically stable sequence of intermediate distributions with limited weight degeneracy.

Figure 6 displays the corresponding posterior sample. The central panel shows the final particle cloud, with tight clusters around all 20 ground-truth component means, thereby recovering the full multimodal structure. The side panels compare the associated one-dimensional marginals with the analytic mixture: across both coordinates, the area-normalized kernel density estimations (KDEs) of the posterior samples closely follow the true densities,

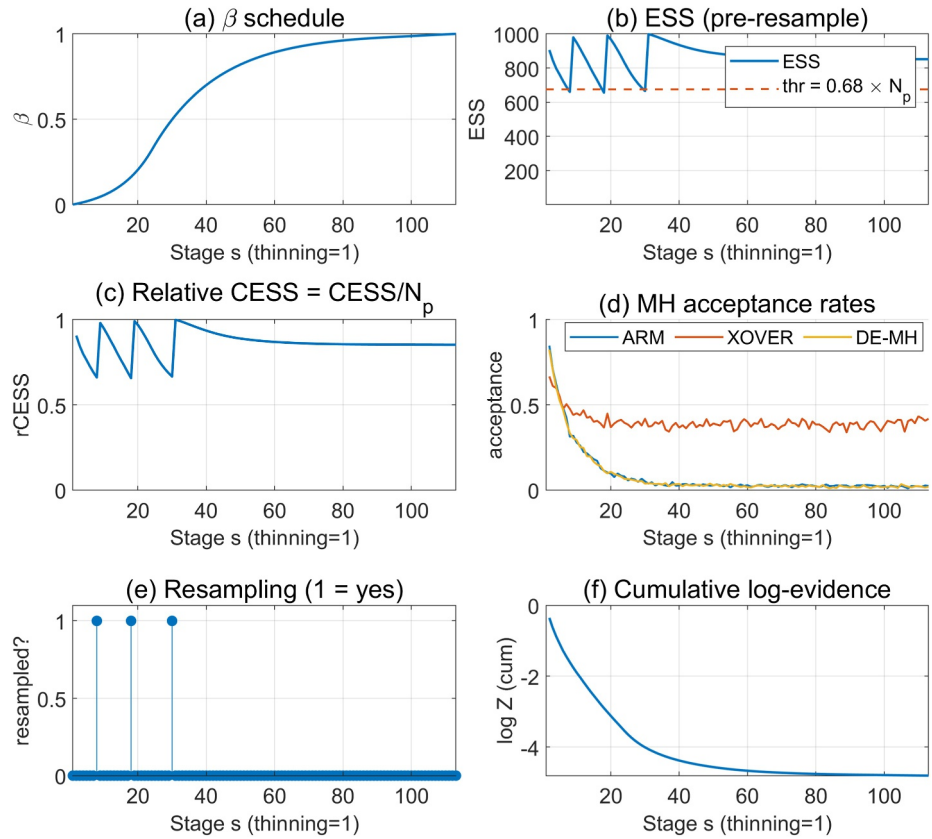

Figure 5. Diagnostics of a PATPEMS run for the 2-D 20-mode Gaussian mixture (representative run, RNG seed = 1). Panels show (a) the sequence of intermediate distributions β_s; (b) pre-resampling ESS with the resampling threshold; (c) relative CESS (=CESS/N_p); (d) M-H acceptance rates for the ARM, XOVER, and DE-MH kernels; (e) resampling indicators (1 = resampled); and (f) the cumulative log-evidence estimate.

although small visual differences remain near a few high-density peaks. The reported marginal moment-alignment index D_s and 1-D Wasserstein-2 distance W_2 are likewise small, corroborating accurate alignment in location, scale, and overall shape.

To assess stochastic variability, we repeated the experiment for random seeds (Text S5.5 in Supporting Information S1). Although different seeds induce modest run-to-run fluctuations, the shaded uncertainty bands (±1 SD) around the marginal KDE means are uniformly narrow, and the cross-seed averages of D_s and W_2 (mean ± SD) remain low, indicating that randomness has limited influence and does not affect the conclusions. Overall, PATPEMS achieves an accurate approximation of the 20-mode target and exhibits robustness to random seeds.

3.2. Case Study 2: A 100-Dimensional Bimodal Gaussian Mixture Distribution

The second case study consideres a challenging high-dimensional synthetic target: a 100-dimensional unequal-weight bimodal Gaussian mixture, examined over the hypercube $[-10,10]^{100}$ (Vrugt et al., 2009). Let $\boldsymbol{\theta} \in [-10,10]^{100}$, and the target density is defined as:

$$\pi(\theta) = \frac{1}{3}\mathcal{N}_{100}(\theta; 51_{100}, \mathbf{I}_{100}) + \frac{2}{3}\mathcal{N}_{100}(\theta; -51_{100}, \mathbf{I}_{100}) \tag{30}$$

where $\mathbf{1}_{100}$ denotes the 100-dimensional vector of ones. This corresponds to two isotropic modes with common covariance $\mathbf{I}_{100}$, unequal mixing weights (1/3, 2/3), and well-separated means at $\pm 51_{100}$. This setting stresses

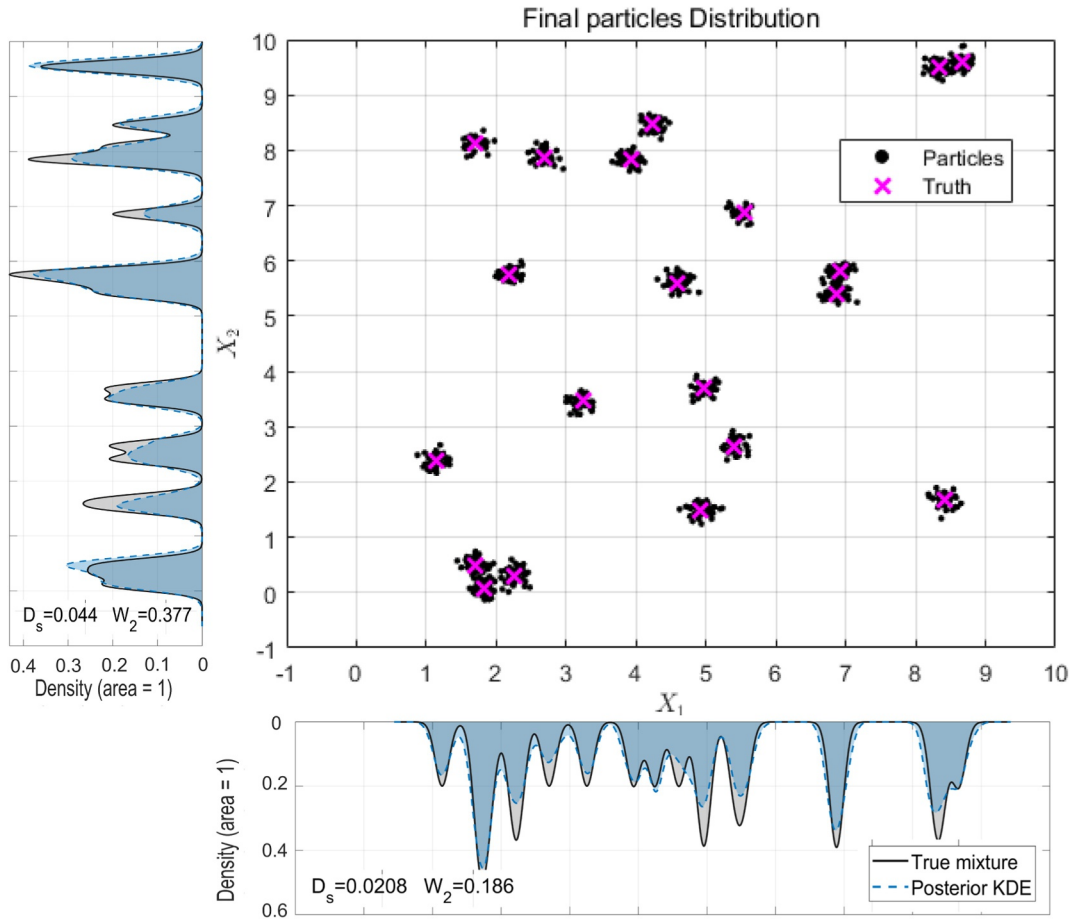

Figure 6. PATPEMS posterior for the 2-D 20-mode Gaussian mixture. The central panel shows the final particle cloud under the default hyperparameter settings, where the black dots denote particles and the magenta crosses mark the true component means. The left and bottom panels display the corresponding one-dimensional marginals for x_2 and x_1, respectively. The solid black curves give the analytic mixture, and the shaded blue regions with dashed blue lines show KDEs of the posterior samples (densities area-normalized). The in-panel numbers report the marginal moment-alignment index D_s and the one-dimensional Wasserstein-2 distance W_2 between the estimated and reference marginals.

multimodal exploration under high dimensionality and unequal occupancy, thereby providing an informative stress test for PATPEMS.

We adopted a balanced baseline configuration, denoted B_1, selected from short pilot runs rather than adopted as a default setting to balance accuracy and computational cost. Specifically, we used $N_p = 1000$ particles and an adaptive sequence of intermediate distributions that keeps the relative CESS within a target band [0.8, 0.9] by adjusting $\Delta\beta_s$, together with a small step floor and a mild growth limiter to avoid stage inflation. Resampling was triggered when the effective sample size satisfied ESS $< 0.72N_p$. The move-operator sequence {ARM, XOVER, DE-MH × 2} combines long range cross mode jumps, high dimensional recombination, and local refinement.

Under this configuration, PATPEMS accurately recovers both modes and their unequal occupancies. The diagnostics in Figure 7 show that the sequence of bridging distributions β_s anneals smoothly to 1, the pre-resampling ESS and relative CESS stay close to their targets with relatively sparse resampling, the M-H acceptance rates remain in the intended range, and the cumulative log-evidence stabilizes without abrupt jumps. These results indicate numerically stable performance in this 100-dimensional setting.

The posterior samples further confirm the accuracy of the approximation. Particle paths quickly move toward ±5 and then fluctuate within each basin with occasional cross-mode transitions (Figure 8a). Figure 8b confirms the marginal fidelity: for $\theta_1-\theta_{20}$, the empirical one-dimensional posteriors are nearly indistinguishable from the

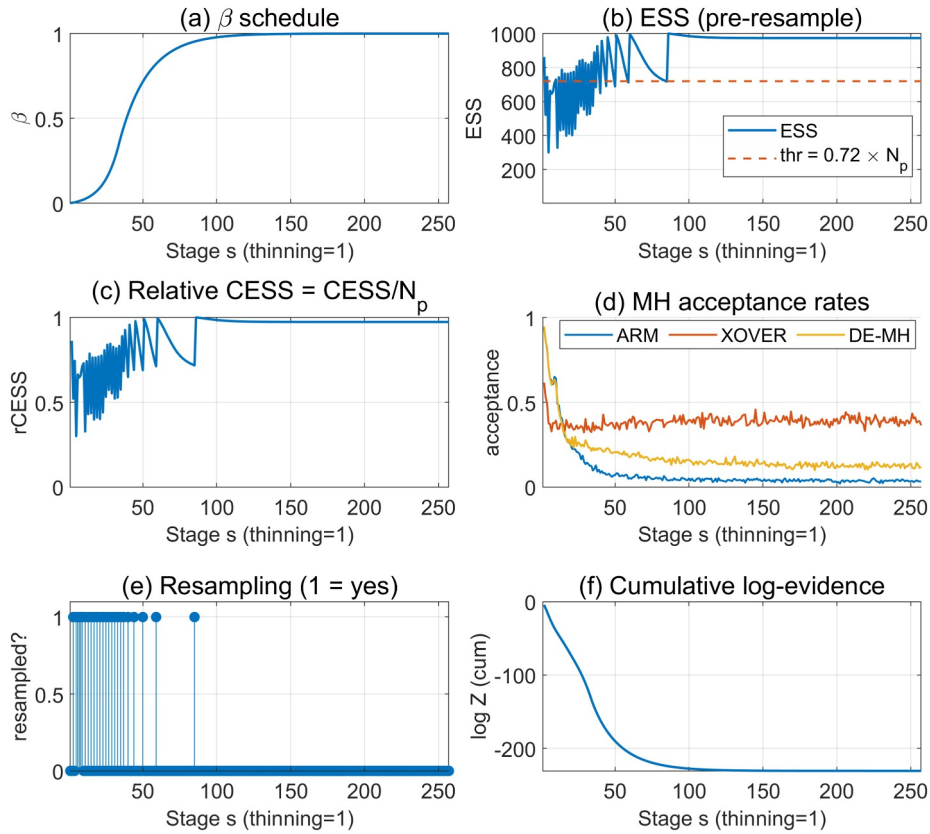

Figure 7. Diagnostics of a PATPEMS run for the 100-dimensional asymmetric bimodal Gaussian target (configuration B_1; RNG seed = 1). The panels show the same diagnostics as in Figure 5: (a) the sequence of intermediate distributions β_s; (b) pre-resampling ESS with the resampling threshold; (c) relative CESS(=CESS/N_p); (d) M-H acceptance rates for the ARM, XOVER, and DE-MH kernels; (e) resampling indicators (1 = resampled); and (f) the cumulative log-evidence log Z.

analytic mixture, and the remaining 80 coordinates exhibit the same level of agreement (Text S6.2 in Supporting Information S1).

To further characterize the tuned PATPEMS configuration used above, we also performed a systematic ablation of the PATPEMS hyperparameters using the same 100 dimensional unequal weight bimodal target as a stress test. The full settings and diagnostics are reported in Text S7 of Supporting Information S1. Briefly, these experiments show that posterior accuracy is most sensitive to the adaptive sequence of intermediate distributions and the DE-MH/XOVER move backbone, whereas wall-clock time is controlled mainly by N_p. With moderately conservative ESS-triggered resampling, once the sequence of bridging distributions and moves are tuned, different random seeds yield very similar results, indicating that this configuration is robust even in high-dimensional multimodal settings.

4. Real-World Case Study: Common Land Model Parameter Estimation

This section introduces the two Common Land Model (CoLM) parameter calibration experiments conducted in realistic model-data settings. Because, in practice, the exact posterior is unknown, we used one synthetic experiment and one real-data experiment to evaluate the performance of PATPEMS. In the synthetic observation case, data were generated from the CoLM with known parameter values and no imposed model-data mismatch. This setup validates the inference pipeline and tests whether PATPEMS can recover the generating parameters and their posterior distribution. The real-data experiment used observations from the A'rou Station site on the east edge of the Qinghai–Tibet Plateau to infer the joint posterior of the key parameters and to assess predictive performance under possible model-data mismatch. In both CoLM experiments, MT-DREAM$_{(ZS)}$ was run with carefully tuned settings and long chains so that its posterior provided a benchmark reference for PATPEMS.

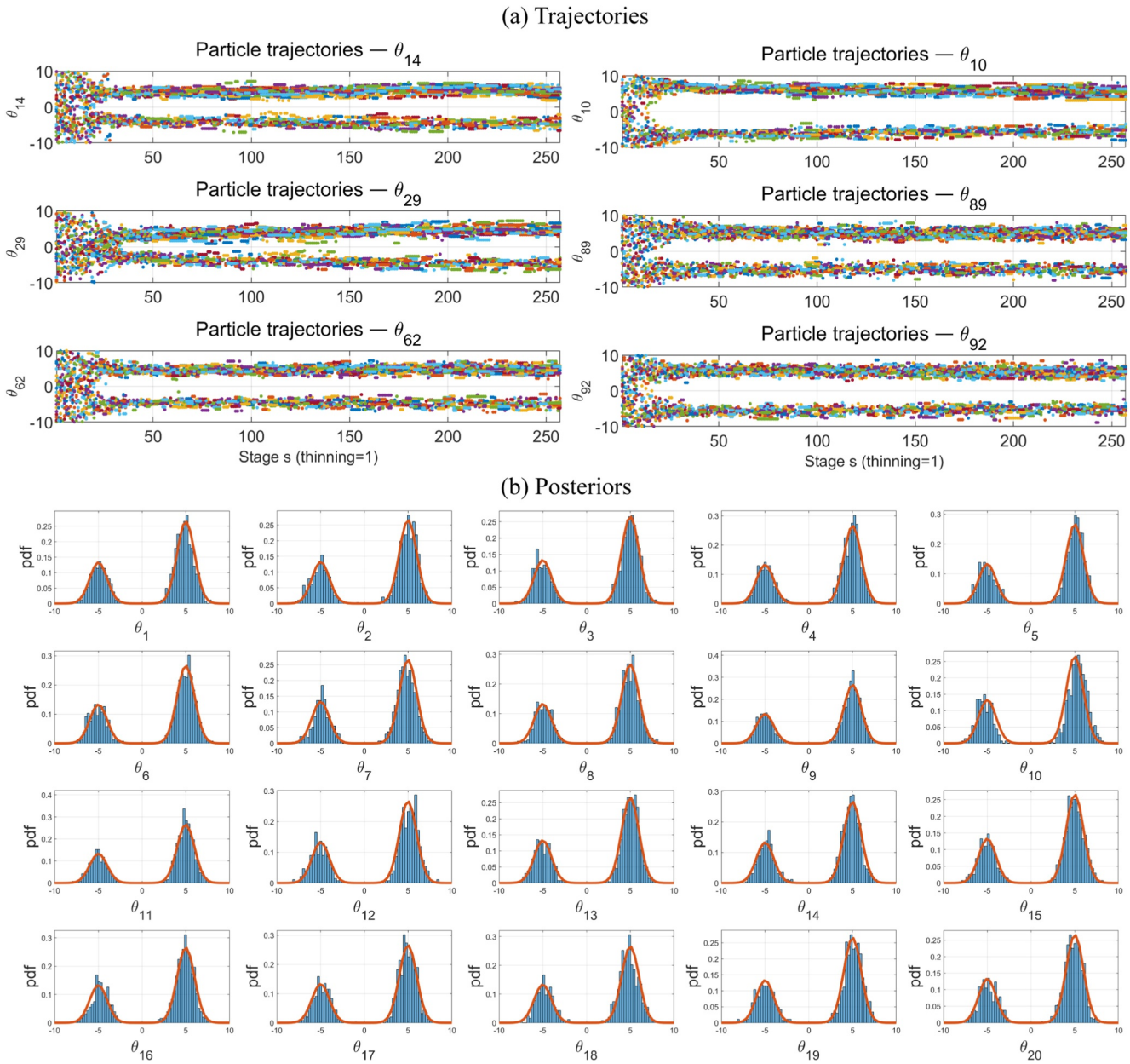

Figure 8. PATPEMS results for the 100-dimensional asymmetric bimodal Gaussian target (configuration B_1; RNG seed = 1). (a) Trajectories of 20 randomly selected particles for six randomly selected coordinates across evolution stages. (b) One-dimensional marginals for coordinates $\theta_1 - \theta_{20}$: the blue histograms show the PATPEMS posterior samples, and the orange curves show the analytic mixture density. The remaining 80 coordinates are provided in Text S6.2 of Supporting Information S1.

Convergence of the MT-DREAM$_{(ZS)}$ algorithm was monitored using the Gelman-Rubin diagnostic $\hat{R}$ (Gelman & Rubin, 1992), and in accordance with the DREAM documentation (Vrugt, 2016), only samples collected after all parameters satisfied $\hat{R} \leq 1.2$ were retained as posterior draws.

4.1. Case Study 3: Synthetic CoLM Benchmark for High-Dimensional Bayesian Calibration

This synthetic benchmark aimed to assess the accuracy of the PATPEMS algorithm for the CoLM under the condition of no model-data mismatch. The CoLM model was developed by Dai et al. (2003) to simulate energy, momentum, water, and carbon exchanges between land and atmosphere (Meng et al., 2009). Based on their importance in water-carbon interactions, three key output variables from the CoLM—latent heat flux (LE), net

ecosystem exchange (NEE), and root zone soil moisture at 10 cm (RSM)—were selected as target variables for the Bayesian parameter estimation. Based on a comprehensive sensitivity analysis (Text S8.1 in Supporting Information S1), six parameters—P34, P2, P4, P33, P5, and P9—were identified and selected for Bayesian estimation of these three targets; their priors and bounds are provided in Table S9 of Supporting Information S1. Half-hourly meteorological forcing data from A'rou Station (38°02′ N, 100°27′ E) were used to drive the CoLM model (Liu et al., 2018; Text S9 in Supporting Information S1). Standard spin-up was performed during June 1 to 31 August 2020, and the 2021 period was used for calibration.

To enable direct quantification of accuracy, we fixed the CoLM default parameter vector as the ground truth $\boldsymbol{\theta}^*$. Synthetic observations were generated by first computing noise-free simulations $f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta}^*)$ for each target $v \in \{\mathrm{LE}, \mathrm{NEE}, \mathrm{RSM}\}$, and then adding independent, zero-mean, homoscedastic Gaussian errors $\varepsilon_t^{(v)} \sim \mathcal{N}(0, \sigma_v^2)$, assumed mutually independent over time and across variables, with $\sigma_v = 0.05 \times \mathrm{SD}\{f^{(v)}\}$. Thus, the synthetic observation for target v at time t is $y_t^{(v)} = f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta}^*) + \varepsilon_t^{(v)}$, and the inference stage adopts the same error model through the three-target joint Gaussian log-likelihood:

$$\log p(\mathbf{y}_t|\boldsymbol{\theta}) = -\frac{1}{2}\sum_{v\in\{\mathrm{LE,NEE,RSM}\}}\left[\log\left(2\pi\sigma_v^2\right) + \frac{\left(y_t^{(v)} - f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta})\right)^2}{\sigma_v^2}\right] \tag{31}$$

Here, $\mathbf{y}_t = \left(y_t^{(\mathrm{LE})}, y_t^{(\mathrm{NEE})}, y_t^{(\mathrm{RSM})}\right)^{\mathrm{T}}$ denotes the three-target observation vector at time t.

To obtain a high-quality reference posterior under identical priors and likelihood, we ran MT-DREAM$_{(\mathrm{ZS})}$ with multiple parallel chains, generations, and multi-try proposals, and adaptive scaling during burn-in (Vrugt, 2016). Hyperparameters and run settings for both MT-DREAM$_{(\mathrm{ZS})}$ and PATPEMS are summarized in Table S3 of Supporting Information S1. For MT-DREAM$_{(\mathrm{ZS})}$, we allocated a total of 10,000 iterations to ensure convergence. Convergence was monitored following the DREAM documentation using the Gelman-Rubin potential scale reduction factor $\hat{R}_q$, where $q = 1, \ldots, d$ indexes the parameter coordinates. The chains are extended until $\max_{q=1,\ldots,d}\hat{R}_q \leq 1.2$. We then applied PATPEMS under the same priors and likelihood specification.

The evaluation focuses on the six sensitivity-selected parameters using two complementary metrics: (a) Distributional discrepancy, quantified by the one-dimensional Wasserstein-2 distance W_2 between per-parameter marginals (definition and computation given earlier; here W_2 is used purely as an inter-distribution distance). (b) Deviation from the default parameter, measured as the absolute difference between each sampler's marginal maximum a posteriori (MAP) estimate and the default value, reported as a percentage of that parameter's prior width, that is, $\left|\hat{\theta}_q^{\mathrm{MAP}} - \theta_q^*\right| / (\theta_{\max,q} - \theta_{\min,q}) \times 100\%$, where $\hat{\theta}_q^{\mathrm{MAP}}$ denotes the marginal MAP estimate of parameter q, θ_q^* is its default value, and $\theta_{\max,q} - \theta_{\min,q}$ is its prior width. Smaller values indicate closer agreement with the default parameter.

Convergence diagnostics are strong for both samplers. For MT-DREAM$_{(\mathrm{ZS})}$, both the univariate and multivariate $\hat{R}$ drop below the threshold early and remain stable, the acceptance rate stays nearly 40%, and the adaptive crossover selection concentrates on $n_{\mathrm{CR}} = 1.00$ (Figures S16–S19 in Supporting Information S1). The trace plots of all MT-DREAM$_{(\mathrm{ZS})}$ chains for the six calibrated parameters further confirm that the chains stabilize quickly around the MAP region. For PATPEMS, the CESS-guided sequence of intermediate distributions is smooth, resampling is triggered when needed, DE-MH and crossover acceptance rates stabilize after an initial ramp-up, the cumulative log-evidence increases monotonically, and the corresponding particle-path plots across stages show rapid concentration and no residual drift for all parameters (Figures S22 and S23 in Supporting Information S1).

With identical priors and likelihood, the two samplers yield closely matching posteriors across all six parameters. The marginal MAP error, normalized by prior width, is <0.2% for both methods, and the 1-D Wasserstein-2 distances (W_2) are small (≤ 0.17 in parameter units; Figure 9). Importantly, the marginal shapes (skewness, tail mass, and peak sharpness) now align closely across all six parameters, with near-perfect agreement between methods. Any remaining discrepancies are negligible relative to the prior ranges and do not affect the point

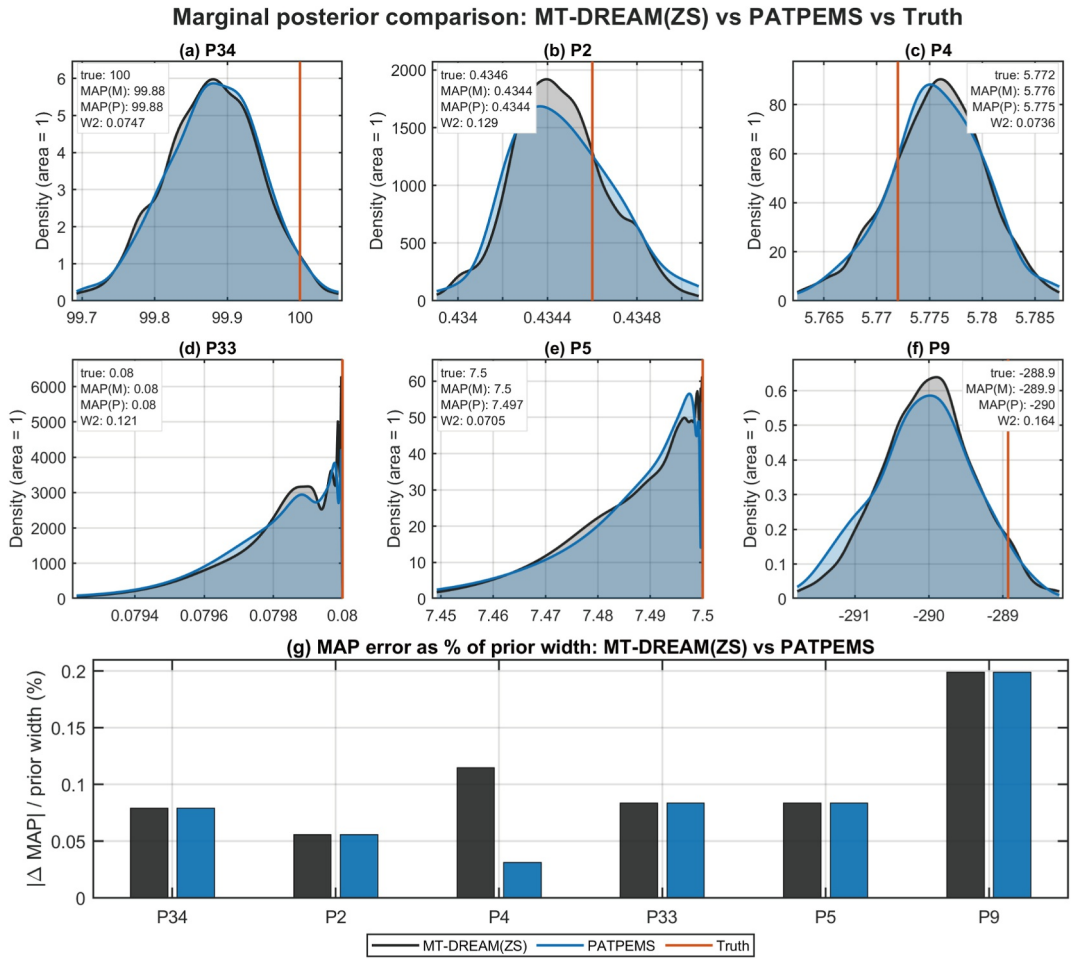

Figure 9. One-dimensional marginal posteriors for six parameters, comparing MT-DREAM$_{(ZS)}$ (black) and PATPEMS (blue). The shaded curves are area-normalized KDEs on the bounded prior domain; the orange vertical line marks the truth. The MT-DREAM$_{(ZS)}$ reference posterior is assembled from post-burn-in stationary draws (last 5,000 iterations; convergence achieved before iteration 2000). The PATPEMS KDE uses samples from the final evolution stage. The insets report, for each panel, the truth, the marginal MAP from MT-DREAM$_{(ZS)}$ [MAP(M)], the marginal MAP from PATPEMS [MAP(P)], and the standardized 1-D Wasserstein distance (W_2). (g) Estimation accuracy summary: absolute MAP deviation from the truth expressed as a percentage of the prior width for each parameter (lower is better). Sampler diagnostics and algorithm-specific marginal histograms are provided in Figures S18–S24 of Supporting Information S1.

estimates or uncertainty quantification. Overall, in this no model-data mismatch setting, PATPEMS accurately recovers the true posterior and closely tracks the long-run MT-DREAM$_{(ZS)}$ baseline.

4.2. Case Study 4: Estimating CoLM Parameters Using Observed Data Sets

In Case 4, we applied PATPEMS and MT-DREAM$_{(ZS)}$ to the CoLM parameter calibration using real observations with two goals: (a) to verify that PATPEMS can reproduce the reference posterior obtained by MT-DREAM$_{(ZS)}$ in a realistic setting, and (b) to assess how parameter calibration improves both the deterministic performance and probabilistic predictive skill of the CoLM. The six sensitive parameters of the CoLM, identified in Case 3, were calibrated against half-hourly observations of LE, NEE, and RSM from the A'rou site for 1 June–31 August 2021.

The main calibration used a concentrated Gaussian likelihood to link the CoLM outputs to the LE, NEE, and RSM observations. Let $y_t^{(v)}$ and $f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta})$ denote the observation and CoLM simulation, respectively, for the variable $v \in \{\mathrm{LE}, \mathrm{NEE}, \mathrm{RSM}\}$ at half-hourly time $t = 1, \ldots, T$. We assume:

$$y_t^{(v)} = f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta}) + \varepsilon_t^{(v)}, \varepsilon_t^{(v)} \sim \mathcal{N}\left(0,\sigma_v^2\right) \tag{32}$$

with residuals that are mutually independent across variables and time. For a given parameter vector $\boldsymbol{\theta}$, we compute the residual sum of squares:

$$\mathrm{SSE}_v(\boldsymbol{\theta}) = \sum_{t=1}^{T}\left(y_t^{(v)} - f^{(v)}(\mathbf{x}_t;\boldsymbol{\theta})\right)^2 \tag{33}$$

and plug in the maximum-likelihood estimate $\hat{\sigma}_v^2(\boldsymbol{\theta}) = \mathrm{SSE}_v(\boldsymbol{\theta})/T$. The resulting joint log-likelihood is:

$$\log p(\mathbf{y}|\boldsymbol{\theta}) = -\frac{T}{2}\sum_{v\in\{\mathrm{LE,NEE,RSM}\}}\left[\log(2\pi) + \log\left(\frac{\mathrm{SSE}_v(\boldsymbol{\theta})}{T}\right) + 1\right] \tag{34}$$

Compared with Case 3, where the residual standard deviations were fixed to 5% of the observed range of each variable, this specification retains the independent Gaussian assumption but lets the error variance be inferred from the residuals at each $\boldsymbol{\theta}$.

Under this Gaussian likelihood, MT-DREAM$_{(ZS)}$ was first run to obtain a reference posterior, and the marginal posteriors from PATPEMS were compared against this reference to check the sampling accuracy. Based on the PATPEMS posterior samples, we then evaluated the impact of calibration by comparing, for each target variable, the root-mean-square error (RMSE) and Nash-Sutcliffe efficiency (NSE) between simulations (default vs. posterior-median parameters) and observations.

To assess the quality of the full posterior predictive distributions, we additionally constructed 95% posterior predictive intervals and summarized them with the negatively oriented interval score (Gneiting & Raftery, 2007). For each time step and variable, the interval score can be calculated as:

$$S_\alpha^{\mathrm{int}}(l,u;x) = (u-l) + \frac{2}{\alpha}(l-x)\,1\{x < l\} + \frac{2}{\alpha}(x-u)\,1\{x > u\} \tag{35}$$

where l and u are the lower and upper bounds of the central $(1-\alpha)\times 100\%$ prediction interval, x is the corresponding observation, $\alpha = 0.05$ in this study (95% intervals), and $1\{\cdot\}$ is the indicator function. When the observation lies inside the interval, the score reduces to the interval width $u - l$; if it falls outside, an additional penalty proportional to the distance from the interval is added. Thus, smaller values of $S_{0.05}^{\mathrm{sin}\,t}$ indicate sharper yet better-calibrated predictive intervals.

Algorithm settings for both samplers are summarized in Supporting Information Table S3. The convergence diagnostics confirm that the MT-DREAM$_{(ZS)}$ reference run is well mixed (Figure S27 in Supporting Information S1): all univariate $\hat{R}$ values for the six calibrated parameters rapidly drop below 1.1, and the multivariate $\hat{R}^d$ likewise approaches 1 after about 1,500 generations, indicating that the resulting reference posterior is stable and reliable. For PATPEMS, the adaptive sequence of intermediate distributions (Figure S28 in Supporting Information S1) similarly show well-behaved intermediate targets, stable particle diversity, and a numerically robust performance throughout the run.

Under the shared Gaussian likelihood assumption, the marginal posteriors from PATPEMS closely match those from MT-DREAM$_{(ZS)}$ for all six CoLM parameters (Figure 10). The MT-DREAM$_{(ZS)}$ reference densities were estimated from the last 5,000 post-burn-in samples, and the PATPEMS KDEs almost overlap with them, with very similar modes and small Wasserstein distances. This agreement indicates that PATPEMS reproduces the MT-DREAM$_{(ZS)}$ posterior accurately in this realistic calibration setting.

Using the posterior samples from PATPEMS under the Gaussian likelihood assumption, we propagated the CoLM over the calibration period and compared the resulting simulations with the default parameter run. It can be seen that calibration systematically improves the deterministic performance (Table 1): RMSE is reduced for all three targets, with particularly strong gains for LE and RSM, and NSE increases accordingly, most notably for RSM, where NSE improves from a negative value for the default run to a clearly positive value after calibration.

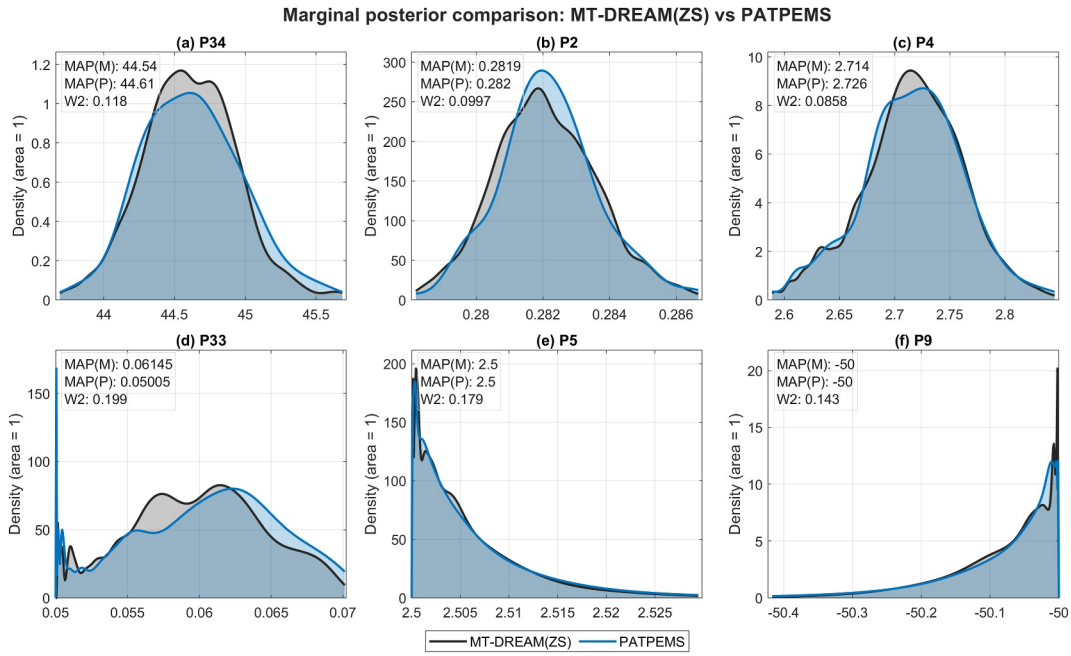

Figure 10. Marginal posterior comparison for the six calibrated CoLM parameters in Case 4: MT-DREAM$_{(ZS)}$ (black) versus PATPEMS (blue) under the Gaussian likelihood assumption. Shaded areas show KDEs normalized to unit area. MT-DREAM$_{(ZS)}$ densities are based on the last 5,000 post-burn-in samples.

Table 1
Deterministic and Probabilistic Performance of the CoLM in Case 4: Default Parameters Versus PATPEMS Calibration Under the Gaussian Likelihood Assumption

Target	Metric	Default	PATPEMS (Gaussian)
LE	RSME	78.71	61.58
	NSE	0.618	0.76
	$S_{0.05}^{\sin t}$	518.56	372.72
NEE	RMSE	8.90	8.49
	NSE	0.25	0.31
	$S_{0.05}^{\sin t}$	53.45	52.42
RSM	RMSE	5.03	2.54
	NSE	−1.51	0.35
	$S_{0.05}^{\sin t}$	22.81	11.93

Note. Metrics are computed on the original half-hourly LE, NEE, and RSM series. RMSE and NSE are based on the CoLM simulations generated with either the default parameter vector or the posterior-median vector from PATPEMS, compared against observations, whereas the interval score is evaluated from the corresponding 95% posterior predictive distributions obtained by combining the CoLM outputs with the Gaussian residual model. Lower RMSE and $S_{0.05}^{\sin t}$ and higher NSE indicate better performance.

Probabilistic performance is likewise enhanced: the interval score $S_{0.05}^{\sin t}$ decreases for LE, NEE, and RSM, indicating sharper yet better-calibrated 95% posterior predictive intervals.

These improvements are also evident in the aggregated daily time series used for visualization (Figure 11). For all targets, the posterior-median simulations track the observations more closely than the default run, reducing both bias and phase errors, while the 95% posterior predictive bands capture most of the observed variability without becoming excessively wide.

As a final sensitivity check, we also repeated the calibration with a more flexible robust likelihood that allows per-series Box-Cox/Yeo-Johnson transformations, AR(1) residual correlation, mild heteroscedasticity, and inflated or Student-t innovations (Text S10.1 in Supporting Information S1). This analysis is kept separate from the main Gaussian-likelihood results above. Under the robust likelihood, MT-DREAM$_{(ZS)}$ and PATPEMS again deliver nearly identical marginal posteriors (Figure S29 in Supporting Information S1), demonstrating that the two algorithms provide consistent inferences across markedly different likelihood specifications. However, comparing the Gaussian and robust likelihoods reveals noticeable changes for some parameters, most clearly P34, P4, and P33, with modest shifts in location and spread. The robust-likelihood comparison also shows smaller gains in RMSE and NSE and substantially larger interval scores, suggesting that, for this data set, the simple independent Gaussian error model provides more informative and better-calibrated predictions than the robust specification. These results indicate that the calibrated CoLM parameters retain some sensitivity to how the residual structure and scaling are modeled.

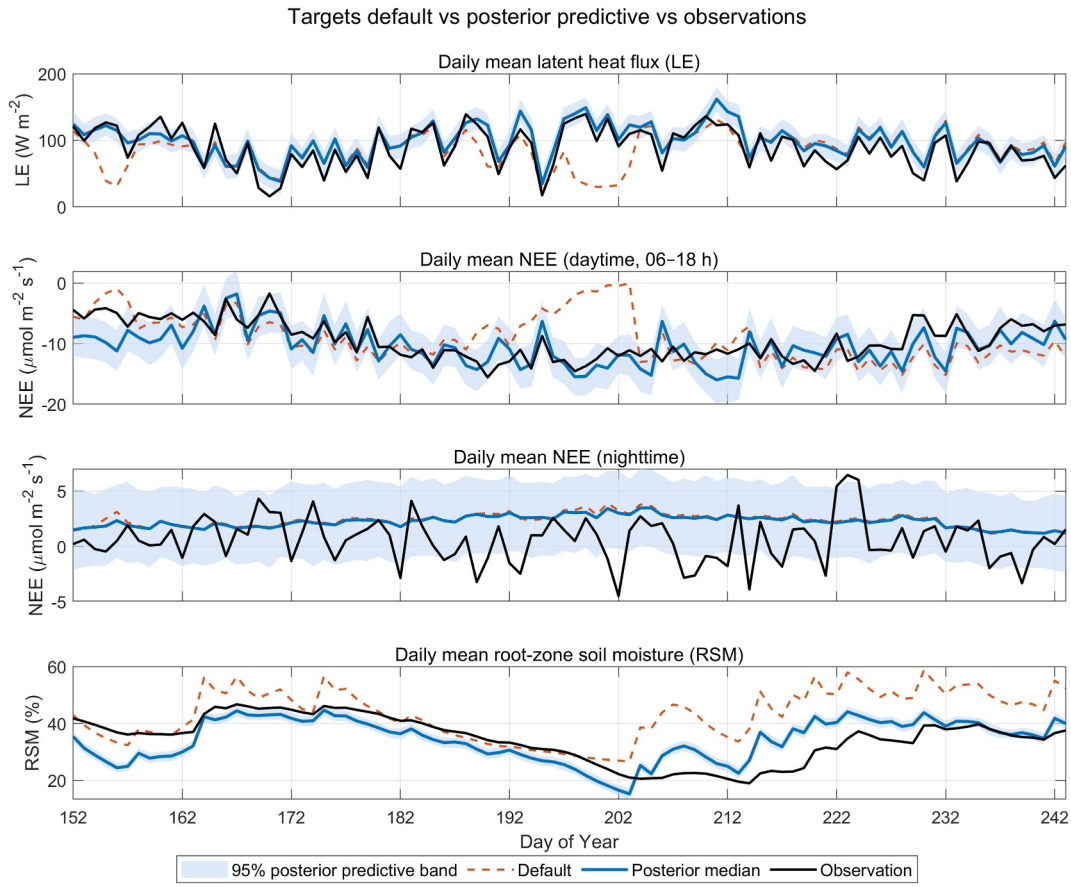

Figure 11. Daily mean CoLM targets: default run versus posterior predictive distribution under the Gaussian likelihood assumption. Half-hourly LE, daytime net ecosystem exchange (NEE), nighttime NEE, and RSM series are averaged to daily means for visualization. The black lines show the observations, the orange dashed lines the default CoLM simulation, the blue solid lines the simulation at the PATPEMS posterior-median parameter vector, and the shaded bands the corresponding 95% posterior predictive intervals obtained by combining the CoLM outputs with the Gaussian residual model.

5. Discussion

5.1. Methodological Positioning of PATPEMS

SMC and MCMC can be combined for different inferential purposes. One line of methods is developed mainly for state-space problems, where particles represent time-evolving latent states or state-parameter trajectories, and MCMC moves are used to improve state-parameter inference. Particle-DREAM is a representative example, as it embeds DREAM-type MCMC kernels within a particle-filtering framework for joint state and parameter estimation in dynamical systems (Vrugt et al., 2013). Another line uses SMC as a sampler for offline approximation of static posterior distributions, where MCMC rejuvenation moves help maintain particle diversity across intermediate targets (Chopin, 2002; Del Moral et al., 2006; Doucet et al., 2000; Jeremiah et al., 2011, 2012). PATPEMS belongs to this latter setting: its particles are candidate parameter vectors, and its sequential structure denotes the bridging process from the prior to the posterior, from $\pi_s(\boldsymbol{\theta})$ to $\pi_{s+1}(\boldsymbol{\theta})$, rather than physical time propagation of model states.

For offline static parameter estimation, the innovation of PATPEMS lies in how the particle evolution process is organized and strengthened. The algorithm integrates an adaptive bridging schedule, flexible ordering and repetition of MCMC move operators, reversible boundary handling, and particle level parallelization. These components are designed to work together in the same sampling framework: the adaptive transition regulates the distance between consecutive intermediate targets, the move operators rejuvenate particles at complementary

exploration scales, the boundary treatment preserves valid proposal behavior under bounded priors, and the parallel structure reduces wall clock burden for expensive model evaluations. In this way, PATPEMS targets the central difficulty of static posterior sampling: moving a finite particle population through increasingly informative intermediate distributions while maintaining diversity and a valid posterior approximation.

5.2. Algorithmic Performance

Building on the original PEM-SMC framework, PATPEMS introduces two additional design components: an adaptive scheme for the sequence of intermediate distributions and a combined MCMC move mechanism with multiple complementary kernels. First, the adaptive sequence of intermediate distributions, driven by a target band for the relative conditional effective sample size, ties each update of the bridging exponent to a stability diagnostic of the weighted particle system, with bounds on the minimum step size and its growth rate to avoid both collapse inducing jumps and an unnecessary proliferation of stages (Del Moral et al., 2006; Jasra et al., 2011; Zhou et al., 2016). In the four case studies with different complexities, the adaptive procedure generally required fewer than 300 intermediate stages to transport the initially specified particles to the target posterior distributions. This suggests that the CESS-based adaptive mechanism can guarantee the generated intermediate distribution sequence transitions quickly and stably from the prior to the posterior. At the same time, by reducing unnecessary intermediate stages and thereby avoiding repeated reweighting and move operations, it further improves the computational efficiency of the PATPEMS algorithm.

Second, the move step in PATPEMS is designed as a configurable manner rather than a fixed procedure. During this step, the combination, number, and order of operators used to update particles can be flexibly specified according to the target distribution. For complex high-dimensional distributions, such as Case Study 2, this flexibility is particularly important, because adding operators or repeating selected operators within a move step can substantially strengthen the exploration ability of the particle population. In the original PEM-SMC algorithm, the move step used a fixed operator sequence {XOVER, DE-MH}, which could handle at most about 35 dimensions for this type of multimodal Gaussian mixture target. In contrast, PATPEMS allows the move configuration to be adjusted more flexibly, and the configuration {ARM, XOVER, DE-MH $\times$ 2} scaled the same target structure to 100 dimensions in Case Study 2. The additional analyses in Text S11 of Supporting Information S1 further examine the role and limitations of XOVER, showing that its contribution depends on the target distribution and that it is more effective in high-dimensional settings, where coordinate-wise recombination can help maintain particle diversity during the move step. These results indicate that the improved high-dimensional performance of PATPEMS arises not from a single transition kernel, but from the configurable combination of multiple complementary operators within the SMC move step. This design also suggests a feasible direction for further improving MCMC-based sampling algorithms. For example, multiple proposal densities or repeated proposal mechanisms can be used at each update stage to generate more diverse candidate samples and thereby strengthen spatial exploration, especially for high-dimensional or multimodal posterior distributions.

5.3. Computational Efficiency

For SMC type methods, computational efficiency depends primarily on four factors: the number of particles, the length of the intermediate distribution sequence, the number of particle update operators, and the cost of each forward model run. Increasing the number of particles or adding move operators generally increases the total number of model evaluations, while a longer sequence of intermediate distributions also leads to repeated reweighting and rejuvenation steps. Therefore, PATPEMS does not remove the intrinsic computational burden of sampling based Bayesian calibration. Its practical efficiency instead comes from the particle based structure of the algorithm, in which most model evaluations during the reweighting and move steps are independent and can be distributed across parallel workers. For example, in the same CoLM experiment, the wall clock time decreased from 16.71 h with 20 workers to 13.51 h with 40 workers and 12.15 h with 64 workers. Thus, the practical advantage of PATPEMS lies not in minimizing the number of serial model evaluations, but in exploiting parallel throughput to convert independent particle evaluations into improved posterior exploration. This makes the method particularly suitable for expensive calibration problems where many forward simulations can be executed in parallel. However, this advantage should not be interpreted as removing the practical barriers associated with very expensive forward models. If a single forward run takes minutes or hours, parallel sampling alone may still be insufficient. In such cases, practical Bayesian calibration will likely require combining sampling methods with

surrogate models, reduced order approximations, or multifidelity strategies (Benner et al., 2015; Gong et al., 2016; Peherstorfer et al., 2018).

A direct comparison with DREAM type algorithms also indicates that PATPEMS may require more total model evaluations in some cases. In the tested cases, PATPEMS generally requires approximately 2–3 times more model evaluations to obtain a comparable posterior approximation, depending on the target distribution, tuning configuration, and accuracy criterion (Texts S5.4, S6.1, and S8.3 in Supporting Information S1). This higher cost mainly reflects the need to maintain and evolve a full particle population through repeated reweighting and multiple move operators. However, these model evaluations are highly parallelizable in PATPEMS, which can partly compensate for the larger evaluation count in terms of wall clock time. Therefore, PATPEMS is more suitable when sufficient parallel computing resources are available and each model run is not prohibitively expensive. In this situation, many particle evaluations can be performed at the same time, making the larger number of model evaluations more manageable. For extremely expensive forward models, additional acceleration strategies may still be needed.

5.4. Structural Deficiencies of the CoLM Model

Parameter calibration with advanced Bayesian algorithms helps not only to tune models but also to expose structural limitations. In our CoLM applications (Cases 3 and 4), both PATPEMS and MT-DREAM$_{(ZS)}$ yield very similar posteriors under matched priors and likelihoods, yet neither removes the systematic underestimation of nighttime NEE (Figure 9). This indicates that the bias is not primarily a parameter issue but reflects the structural uncertainty, in particular in the parameterization of soil respiration. In the CoLM, soil respiration is represented by a purely temperature-driven Lloyd-Taylor-type exponential function (Lloyd & Taylor, 1994), originally fitted to annual or longer-term data, which is known to be inadequate for resolving sub-daily carbon dynamics (Chen et al., 2013; Raich et al., 2002). Our results support strengthening the soil carbon module by incorporating more mechanistic biogeochemical couplings, such as Century style carbon pools (Parton et al., 1993), microbial regulation as in RothC-26.3 (Coleman & Jenkinson, 1996), and enzyme-mediated kinetics similar to recent CoLM implementations (Lawrence et al., 2019).

Case 4 shows that likelihood choice cannot by itself compensate for structural error. Under both a simple Gaussian residual model and a more elaborate robust likelihood, the two samplers produce nearly identical posteriors for a given likelihood, and neither likelihood formulation removes the nighttime NEE bias. This suggests that, in the present setting, the differences between Gaussian and robust error models mainly redistribute how mismatch is partitioned between "noise" and structure, while the dominant limitation lies in the process representation itself rather than in the sampler or the specific residual model.

6. Conclusion

In this study, we developed the PATPEMS algorithm, an adaptive and parallel sequential Monte Carlo sampler for Bayesian parameter calibration in complex environmental models. Building on our previous PEM-SMC framework, PATPEMS introduces three key refinements aimed at the practical limitations of the existing SMC-based approaches: (a) adaptive intermediate-distribution scheme to control weight degeneracy and automatically select intermediate stages; (b) flexible scheduling of Metropolis-type move kernels, coupled with appropriate boundary handling, to improve particle diversity and mixing under box constraints; and (c) explicit particle-level parallelization to exploit modern multicore architectures and reduce wall-clock time for computationally intensive models. Together, these design choices strengthen the robustness and computational efficiency of SMC on challenging posterior targets.

The performance of PATPEMS was assessed on four case studies of increasing complexity, including multimodal and high-dimensional synthetic targets and a computationally demanding LSM (CoLM) constrained by multistream observations of latent heat, NEE, and root-zone soil moisture. Across all the experiments, PATPEMS provided close approximations to the target posteriors, including synthetic ground truths and reference solutions from the MT-DREAM$_{(ZS)}$ MCMC algorithm. In the CoLM applications, the calibrated posteriors delivered clear improvements over the default parameter set in reproducing the observed dynamics of all three target variables, while the parallel implementation yielded substantial reductions in wall-clock time compared with a non-parallel SMC baseline. Relative to the original PEM-SMC implementation, PATPEMS reduced the number of evolution

stages and model evaluations needed to reach a given posterior resolution, and showed clear, although sub-linear, strong scaling as the number of workers increased.

Overall, PATPEMS provides a practical and scalable Bayesian calibration tool for multimodal, high-dimensional, and computationally expensive land surface and environmental models. Future work will focus on tighter integration with advanced parallel hardware, automated tuning of move-kernel mixtures and likelihood specifications, and extending the framework to higher-dimensional parameter spaces and additional Earth system applications.

Conflict of Interest

The authors declare no conflicts of interest relevant to this study.

Availability Statement

The meteorological forcing data and eddy covariance observations from the A'rou Station site were obtained from the National Tibetan Plateau Data Center (https://doi.org/10.11888/Atmos.tpdc.272465). The PSUADE version 1.7.8a parameter sensitivity analysis software package is available at https://computing.llnl.gov/projects/psuade/software. The CoLM model codes (2014 version) can be accessed via the Land-Atmosphere Interaction Research Group at Sun Yat-sen University, detailed at http://globalchange.bnu.edu.cn/research/models. The DREAM$_{(ZS)}$ algorithm as code was downloaded from https://faculty.sites.uci.edu/jasper/software. The MATLAB implementations of the PATPEMS and MT-DREAM$_{(ZS)}$ algorithms used in case studies 1–4 are openly available on Zenodo at https://doi.org/10.5281/zenodo.17546472.

Acknowledgments

This study was supported by the Science and Technology Research Project of Inner Mongolia Autonomous Region Inner Mongolia "Unveiling the List and Leading the Way" Project (2024JBGS0003-1), the National Natural Science Foundation of China (42471105), the Natural Science Foundation of Gansu Province (23JRRA1025), the Gansu Provincial Science and Technology Plan Project (24YFFA012). We would like to express our sincere gratitude to the principal investigators and their teams for providing the site observation data sets used in this study. In addition, we acknowledge the support from the Supercomputing Center of Lanzhou University.

References

Andrieu, C., Doucet, A., & Holenstein, R. (2010). Particle markov chain Monte Carlo methods. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *72*(3), 269–342. https://doi.org/10.1111/j.1467-9868.2009.00736.x

Annan, J. D., Hargreaves, J. C., Edwards, N. R., & Marsh, R. (2005). Parameter estimation in an intermediate complexity earth system model using an ensemble Kalman filter. *Ocean Modelling*, *8*(1–2), 135–154. https://doi.org/10.1016/j.ocemod.2003.12.004

Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, *50*(2), 174–188. https://doi.org/10.1109/78.978374

Benner, P., Gugercin, S., & Willcox, K. (2015). A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, *57*(4), 483–531. https://doi.org/10.1137/130932715

Blyth, E. M., Arora, V. K., Clark, D. B., Dadson, S. J., De Kauwe, M. G., Lawrence, D. M., et al. (2021). Advances in land surface modelling. *Current Climate Change Reports*, *7*(2), 45–71. https://doi.org/10.1007/s40641-021-00171-5

Bonan, G. B., & Doney, S. C. (2018). Climate, ecosystems, and planetary futures: The challenge to predict life in Earth system models. *Science*, *359*(6375), eaam8328. https://doi.org/10.1126/science.aam8328

Braak, C. J. F. T. (2006). A Markov Chain Monte Carlo version of the genetic algorithm differential evolution: Easy Bayesian computing for real parameter spaces. *Statistics and Computing*, *16*(3), 239–249. https://doi.org/10.1007/s11222-006-8769-1

Calderhead, B., & Girolami, M. (2009). Estimating Bayes factors via thermodynamic integration and population MCMC. *Computational Statistics & Data Analysis*, *53*(12), 4028–4045. https://doi.org/10.1016/j.csda.2009.07.025

Chen, S., Huang, Y., Xie, W., Zou, J., Lu, Y., & Hu, Z. (2013). A new estimate of global soil respiration from 1970 to 2008. *Chinese Science Bulletin*, *58*(33), 4153–4160. https://doi.org/10.1007/s11434-013-5912-1

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, *89*(3), 539–552. https://doi.org/10.1093/biomet/89.3.539

Coleman, K., & Jenkinson, D. S. (1996). RothC-26.3 - A model for the turnover of carbon in soil. In D. S. Powlson, P. Smith, & J. U. Smith (Eds.), *Evaluation of soil organic matter models* (pp. 237–246). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-61094-3_17

Dai, Y., Zeng, X., Dickinson, R. E., Baker, I., Bonan, G. B., Bosilovich, M. G., et al. (2003). The common land model. *Bulletin of the American Meteorological Society*, *84*(8), 1013–1024. https://doi.org/10.1175/BAMS-84-8-1013

Del Moral, P., Doucet, A., & Jasra, A. (2006). Sequential monte Carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *68*(3), 411–436. https://doi.org/10.1111/j.1467-9868.2006.00553.x

Doucet, A., Godsill, S., & Andrieu, C. (2000). On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, *10*(3), 197–208. https://doi.org/10.1023/A:1008935410038

Doucet, A., & Johansen, A. M. (2010). A tutorial on particle Filtering and Smoothing: Fifteen years later.

Earl, D. J., & Deem, M. W. (2005). Parallel tempering: Theory, applications, and new perspectives. *Physical Chemistry Chemical Physics*, *7*(23), 3910. https://doi.org/10.1039/b509983h

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, *99*(C5), 10143–10162. https://doi.org/10.1029/94JC00572

Fan, Y., Leslie, D. S., & Wand, M. P. (2008). Generalised linear mixed model analysis via sequential Monte Carlo sampling. *Electronic Journal of Statistics*, *2*(none). https://doi.org/10.1214/07-EJS158

Fisher, R. A., & Koven, C. D. (2020). Perspectives on the future of land surface models and the challenges of representing complex terrestrial systems. *Journal of Advances in Modeling Earth Systems*, *12*(4), e2018MS001453. https://doi.org/10.1029/2018MS001453

Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, *7*(4). https://doi.org/10.1214/ss/1177011136

Geweke, J., & Durham, G. (2011). Massively parallel sequential monte carlo for Bayesian inference. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.1964731

Gilks, W. R., & Berzuini, C. (2001). Following a moving target—Monte carlo inference for dynamic bayesian models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *63*(1), 127–146. https://doi.org/10.1111/1467-9868.00280

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*(477), 359–378. https://doi.org/10.1198/016214506000001437

Gong, W., Duan, Q., Li, J., Wang, C., Di, Z., Ye, A., et al. (2016). Multiobjective adaptive surrogate modeling-based optimization for parameter estimation of large, complex geophysical models. *Water Resources Research*, *52*(3), 1984–2008. https://doi.org/10.1002/2015WR018230

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

Huang, M., Ray, J., Hou, Z., Ren, H., Liu, Y., & Swiler, L. (2016). On the applicability of surrogate-based Markov chain Monte Carlo-Bayesian inversion to the community land model: Case studies at flux tower sites. *Journal of Geophysical Research: Atmospheres*, *121*(13), 7548–7563. https://doi.org/10.1002/2015JD024339

Jasra, A., Stephens, D. A., Doucet, A., & Tsagaris, T. (2011). Inference for lévy-driven stochastic volatility models via adaptive sequential Monte Carlo. *Scandinavian Journal of Statistics*, *38*(1), 1–22. https://doi.org/10.1111/j.1467-9469.2010.00723.x

Jeremiah, E., Sisson, S., Marshall, L., Mehrotra, R., & Sharma, A. (2011). Bayesian calibration and uncertainty analysis of hydrological models: A comparison of adaptive Metropolis and sequential Monte Carlo samplers. *Water Resources Research*, *47*(7), 2010WR010217. https://doi.org/10.1029/2010WR010217

Jeremiah, E., Sisson, S. A., Sharma, A., & Marshall, L. (2012). Efficient hydrological model parameter optimization with sequential Monte Carlo sampling. *Environmental Modelling & Software*, *38*, 283–295. https://doi.org/10.1016/j.envsoft.2012.07.001

Khan, Z., Balch, T., & Dellaert, F. (2004). An MCMC-based particle filter for tracking multiple interacting targets. In T. Pajdla & J. Matas (Eds.), *Computer vision - ECCV 2004* (pp. 279–290). Springer. https://doi.org/10.1007/978-3-540-24673-2_23

Lawrence, D. M., Fisher, R. A., Koven, C. D., Oleson, K. W., Swenson, S. C., Bonan, G., et al. (2019). The community land model version 5: Description of new features, benchmarking, and impact of forcing uncertainty. *Journal of Advances in Modeling Earth Systems*, *11*(12), 4245–4287. https://doi.org/10.1029/2018MS001583

LeBauer, D. S., Wang, D., Richter, K. T., Davidson, C. C., & Dietze, M. C. (2013). Facilitating feedbacks between field measurements and ecosystem models. *Ecological Monographs*, *83*(2), 133–154. https://doi.org/10.1890/12-0137.1

Li, T., Sun, S., Sattar, T. P., & Corchado, J. M. (2014). Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches. *Expert Systems with Applications*, *41*(8), 3944–3954. https://doi.org/10.1016/j.eswa.2013.12.031

Liang, F., Liu, C., & Carroll, R. (2011). *Advanced markov chain Monte Carlo methods: Learning from past samples*. John Wiley & Sons.

Liang, F., & Wong, W. H. (2001). Real-parameter evolutionary Monte Carlo with applications to bayesian mixture models. *Journal of the American Statistical Association*, *96*(454), 653–666. https://doi.org/10.1198/016214501753168325

Lindsten, F., Jordan, M. I., & Schön, T. B. (2014). Particle Gibbs with ancestor sampling. *Journal of Machine Learning Research*, *15*(1), 2145–2184.

Link, W. A., & Barker, R. J. (2009). *Bayesian inference: With ecological applications*. Academic Press.

Liu, S., Li, X., Xu, Z., Che, T., Xiao, Q., Ma, M., et al. (2018). The heihe integrated observatory network: A basin-scale land surface processes observatory in China. *Vadose Zone Journal*, *17*(1), 1–21. https://doi.org/10.2136/vzj2018.04.0072

Lloyd, J., & Taylor, J. A. (1994). On the temperature dependence of soil respiration. *Functional Ecology*, *8*(3), 315. https://doi.org/10.2307/2389824

Lu, D., Ricciuto, D., Stoyanov, M., & Gu, L. (2018). Calibration of the E3SM land model using surrogate-based global optimization. *Journal of Advances in Modeling Earth Systems*, *10*(6), 1337–1356. https://doi.org/10.1002/2017MS001134

Luengo, D., Martino, L., Bugallo, M., Elvira, V., & Särkkä, S. (2020). A survey of Monte Carlo methods for parameter estimation. *EURASIP Journal on Applied Signal Processing*, *2020*(1), 25. https://doi.org/10.1186/s13634-020-00675-6

Marin, J.-M., Pudlo, P., Robert, C. P., & Ryder, R. J. (2012). Approximate bayesian computational methods. *Statistics and Computing*, *22*(6), 1167–1180. https://doi.org/10.1007/s11222-011-9288-2

Meng, C. L., Li, Z.-L., Zhan, X., Shi, J. C., & Liu, C. Y. (2009). Land surface temperature data assimilation and its impact on evapotranspiration estimates from the common land model. *Water Resources Research*, *45*(2), 2008WR006971. https://doi.org/10.1029/2008WR006971

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*(6), 1087–1092. https://doi.org/10.1063/1.1699114

Miasojedow, B., Moulines, E., & Vihola, M. (2013). An adaptive parallel tempering algorithm. *Journal of Computational & Graphical Statistics*, *22*(3), 649–664. https://doi.org/10.1080/10618600.2013.778779

Owen, A. B., & Tribble, S. D. (2005). A quasi-Monte Carlo Metropolis algorithm. *Proceedings of the National Academy of Sciences*, *102*(25), 8844–8849. https://doi.org/10.1073/pnas.0409596102

Parton, W. J., Scurlock, J. M. O., Ojima, D. S., Gilmanov, T. G., Scholes, R. J., Schimel, D. S., et al. (1993). Observations and modeling of biomass and soil organic matter dynamics for the grassland biome worldwide. *Global Biogeochemical Cycles*, *7*(4), 785–809. https://doi.org/10.1029/93GB02042

Pattuk, E., Kantarcioglu, M., Ulusoy, H., & Malin, B. (2016). CheapSMC: A framework to minimize SMC cost in cloud (version 1). *arXiv*. https://doi.org/10.48550/ARXIV.1605.00300

Peherstorfer, B., Willcox, K., & Gunzburger, M. (2018). Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Review*, *60*(3), 550–591. https://doi.org/10.1137/16M1082469

Pokhrel, P., Gupta, H. V., & Wagener, T. (2008). A spatial regularization approach to parameter estimation for a distributed watershed model. *Water Resources Research*, *44*(12), 2007WR006615. https://doi.org/10.1029/2007WR006615

Raich, J. W., Potter, C. S., & Bhagawati, D. (2002). Interannual variability in global soil respiration, 1980–94. *Global Change Biology*, *8*(8), 800–812. https://doi.org/10.1046/j.1365-2486.2002.00511.x

Rainforth, T., Naesseth, C., Lindsten, F., Paige, B., Vandemeent, J.-W., Doucet, A., & Wood, F. (2016). Interacting particle markov chain Monte Carlo. In M. F. Balcan & K. Q. Weinberger (Eds.), *Proceedings of the 33rd international conference on machine learning* (Vol. 48, pp. 2616–2625). PMLR. Retrieved from https://proceedings.mlr.press/v48/rainforth16.html

Ricciuto, D. M., King, A. W., Dragoni, D., & Post, W. M. (2011). Parameter and prediction uncertainty in an optimized terrestrial carbon cycle model: Effects of constraining variables and data record length. *Journal of Geophysical Research*, *116*(G1), G01033. https://doi.org/10.1029/2010JG001400

Ricciuto, D., Sargsyan, K., & Thornton, P. (2018). The impact of parametric uncertainties on biogeochemistry in the E3SM land model. *Journal of Advances in Modeling Earth Systems*, *10*(2), 297–319. https://doi.org/10.1002/2017MS000962

Speich, M., Dormann, C. F., & Hartig, F. (2021). Sequential Monte-Carlo algorithms for Bayesian model calibration – A review and method comparison. *Ecological Modelling*, *455*, 109608. https://doi.org/10.1016/j.ecolmodel.2021.109608

Ter Braak, C. J. F., & Vrugt, J. A. (2008). Differential evolution Markov Chain with snooker updater and fewer chains. *Statistics and Computing*, *18*(4), 435–446. https://doi.org/10.1007/s11222-008-9104-9

Ter Braak, C. J. F. T. (2006). A Markov Chain Monte Carlo version of the genetic algorithm differential evolution: Easy Bayesian computing for real parameter spaces. *Statistics and Computing*, *16*(3), 239–249. https://doi.org/10.1007/s11222-006-8769-1

Vrugt, J. A. (2016). Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation. *Environmental Modelling & Software*, *75*, 273–316. https://doi.org/10.1016/j.envsoft.2015.08.013

Vrugt, J. A., Hyman, J. M., Robinson, B. A., Higdon, D., & Diks, C. G. H. (2007). Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Stimulation*. Retrieved from https://www.osti.gov/biblio/960766

Vrugt, J. A., & Ter Braak, C. J. (2011). Dream (D): An adaptive Markov Chain Monte Carlo simulation algorithm to solve discrete, noncontinuous, and combinatorial posterior parameter estimation problems. *Hydrology and Earth System Sciences*, *15*(12), 3701–3713. https://doi.org/10.5194/hess-15-3701-2011

Vrugt, J. A., ter Braak, C. J., Diks, C. G., & Schoups, G. (2013). Hydrologic data assimilation using particle Markov chain Monte Carlo simulation: Theory, concepts and applications. *Advances in Water Resources*, *51*, 457–478. https://doi.org/10.1016/j.advwatres.2012.04.002

Vrugt, J. A., Ter Braak, C. J. F., Diks, C. G. H., Robinson, B. A., Hyman, J. M., & Higdon, D. (2009). Accelerating markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International Journal of Nonlinear Sciences and Numerical Stimulation*, *10*(3). https://doi.org/10.1515/IJNSNS.2009.10.3.273

Wang, Y., Ma, R., & Zhu, G. (2023). Representation of the influence of soil structure on hydraulic conductivity prediction. *Journal of Hydrology*, *619*, 129330. https://doi.org/10.1016/j.jhydrol.2023.129330

Xu, C., Zhu, G., Zhang, Y., & Che, T. (2024). Partitioning of evapotranspiration and its response to eco-environmental factors over an alpine grassland on the Qinghai–Tibetan Plateau based on the SiB2 model. *Hydrological Processes*, *38*(10), e15295. https://doi.org/10.1002/hyp.15295

Zhou, Y., Johansen, A. M., & Aston, J. A. D. (2016). Toward automatic model comparison: An adaptive sequential Monte Carlo approach. *Journal of Computational & Graphical Statistics*, *25*(3), 701–726. https://doi.org/10.1080/10618600.2015.1060885

Zhu, G., Chen, Q., Yu, X., Xu, C., Zhang, K., Wang, Y., et al. (2025). PEM-SMC: An algorithm for optimizing model parameters. *Software Impacts*, *23*, 100728. https://doi.org/10.1016/j.simpa.2024.100728

Zhu, G., Li, X., Ma, J., Wang, Y., Liu, S., Huang, C., et al. (2018). A new moving strategy for the sequential Monte Carlo approach in optimizing the hydrological model parameters. *Advances in Water Resources*, *114*, 164–179. https://doi.org/10.1016/j.advwatres.2018.02.007

Zyphur, M. J., & Oswald, F. L. (2015). Bayesian estimation and inference: A user's guide. *Journal of Management*, *41*(2), 390–420. https://doi.org/10.1177/0149206313501200