Mechanical Systems and Signal Processing 232 (2025) 112689

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Dynamic model updating through reliability-based sequential history matching

J. Cheng [a], F.A. DiazDelaO [a,*], P.O. Hristov [b]

[a] Clinical Operational Research Unit, Department of Mathematics, University College London, London, UK
[b] GATE Institute, Sofia University "St. Kliment Ohridski", Sofia, Bulgaria

ARTICLE INFO

Communicated by H. Ouyang

Keywords:
Model updating
Uncertainty quantification
Bayesian history matching
Sequential Monte Carlo

ABSTRACT

Computer models enable the study of complex systems and are extensively used in fields such as physics, engineering, and biology. History Matching (HM) is a statistical calibration method that accounts for various sources of uncertainty to update model parameters and align output with observed data. By iteratively excluding regions of the parameter space unlikely to yield plausible outputs, HM identifies and samples from the so-called non-implausible domain. However, a limitation of HM is that it does not yield full Bayesian posterior distributions for model parameters. Moreover, HM requires re-execution from scratch when new data is observed, lacking the ability to leverage prior results.

To address these limitations, we propose integrating sequential Monte Carlo (SMC) methods with HM to achieve full Bayesian posterior distributions for sequential calibration. The SMC framework offers a flexible and computationally efficient means to update previously constructed distributions as new data becomes available. This approach is demonstrated using an engineering example and a cardio-respiratory case study with sequential data.

Our results show that small perturbations to the posterior distributions can be effectively learned sequentially by updating computed posterior distributions through the SMC framework, thereby enabling dynamic and efficient model updating for evolving data streams.

1. Introduction

Computer models, also called simulators, have become indispensable tools for the mathematical modeling of physical systems. They have consistently and successfully been used across diverse fields, including physics, astrophysics, economics, healthcare, and engineering [1]. Model updating aims to infer plausible values for simulator parameters based on experimental data, a process also known as calibration [2–4]. A key challenge lies in accounting for various sources of uncertainty, such as measurement errors caused by instrument limitations and model discrepancy arising from unavoidable assumptions, simplifications, or incomplete knowledge of the underlying process. The reliability of predictions and decision-making in simulations is closely tied to how well they are calibrated to experimental data while addressing these uncertainties [5,6].

Several methodologies for uncertainty quantification have been developed within the Bayesian framework. These approaches update a prior distribution of unknown parameters to a posterior distribution using data. This is typically done through Markov chain Monte Carlo (MCMC) [7,8]. However, this can be computationally expensive, and this is even more expensive when the underlying simulator is also expensive. To address these challenges, Bayesian History Matching (BHM) [9,10] was developed as an efficient alternative for achieving full Bayesian inference in calibration. History Matching (HM), a likelihood-free calibration

* Corresponding author.
E-mail address: alex.diaz@ucl.ac.uk (F.A. DiazDelaO).

https://doi.org/10.1016/j.ymssp.2025.112689
Received 30 November 2024; Received in revised form 21 February 2025; Accepted 31 March 2025
Available online 15 April 2025

approach [11], identifies acceptable regions in the parameter space under the assumption of multiple sources of uncertainty. HM begins by exploring the entire parameter space and iteratively eliminates regions unlikely to produce plausible matches between observations and model outputs. This is achieved through an *implausibility measure*, which quantifies a standardized distance between observations and simulator outputs given model assumptions and uncertainties. At each iteration, also called *wave*, regions deemed implausible are discarded, progressively shrinking the *non-implausible space*.

Unlike methods relying on exact likelihood functions, HM's flexibility allows users to focus initially on subsets of parameters and outputs, enabling efficient narrowing to target regions [9]. For computationally expensive simulators, emulation techniques [12] can be employed to approximate the original model, reducing computational costs while adapting the emulator to the updated non-implausible space at each wave. When computational costs are manageable, emulation is unnecessary [13]. Although HM does not produce probabilistic statements about unknown parameters, the non-implausible space often encloses high-probability regions of the true posterior, making it a valuable precursor for Bayesian calibration. For this reason, it can also be referred to as a precalibration method.

To approximate posterior distributions after identifying the final non-implausible space, importance sampling (IS) and resampling techniques have been proposed [9,10], forming the basis of BHM. However, BHM has some limitations: achieving good performance requires a large sample size if the importance proposal derived from HM is not sufficiently similar to the true posterior. Additionally, sampling within each wave of HM to explore parameter space remains challenging, particularly for complex simulators where target regions may be orders of magnitude smaller than the original space. Furthermore, BHM is ill-suited for dynamic Bayesian model updating, where parameter values evolve over time and data arrive sequentially—a common scenario in real-world applications.

In this work, we propose a method that combines HM with a sequential Monte Carlo (SMC) sampler to address these limitations. The proposed approach is tested on two case studies: a three-degree-of-freedom mass–spring system with simulated data and a patient-specific respiratory model using sequential real data streams from an intensive care unit (ICU). The paper is organized as follows. In Sections 2 and 3, we discuss a general framework of Bayesian history matching and introduce sequential Monte Carlo sampler to prepare the ground for the introduction of our new algorithm, respectively. We describe the sequence of calibration distributions and our new algorithm itself in Section 4 and show experimental results on a three degree-of-freedom mass spring system and a patient-specific respiratory model in Section 5. Finally, we conclude the paper with a discussion for future perspectives in Section 6.

2. Bayesian history matching

2.1. Bayesian model updating

Bayesian model updating is the process of inferring distributions for model parameters to align a model output with observed data. This method incorporates *prior beliefs* about the unknown parameters, expressed as a prior distribution, and updates these beliefs using Bayes' Theorem based on the likelihood of observed data.

To rigorously perform Bayesian model updating, it is crucial to account for various sources of uncertainty. One such source is *experimental or measurement error*, which arises from the discrepancies between observations and the true values of the underlying process, and is subject to the precision of measurement devices. This relationship can be described as:

$$\boldsymbol{z} = \boldsymbol{y} + \boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^p$ represents the true values of the underlying physical process, $\boldsymbol{z} \in \mathbb{R}^p$ denotes the corresponding observations, and $\boldsymbol{\epsilon}$ is the measurement error vector.

When simulating the underlying process via a computer model, $\boldsymbol{y}$ cannot be perfectly known, even if the model parameters were known exactly. This discrepancy arises due to inevitable simplifying assumptions, incomplete knowledge, or mathematical approximations. This additional uncertainty, referred to as *model discrepancy*, is represented as an additive correction term [12,14]. Based on this, the relationship between the outputs of the simulator and observations can be expressed as:

$$\boldsymbol{z} = \eta(\boldsymbol{x}) + \boldsymbol{\delta} + \boldsymbol{\epsilon}, \tag{2}$$

where $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ represents the unknown parameters of interest, $\eta(\cdot) : \mathcal{X} \to \mathbb{R}^p$ denotes the simulator, and $\boldsymbol{\delta}$ is the model discrepancy. The simulator, model discrepancy, and measurement error are assumed to be independent.

In this work, we assume $\boldsymbol{\delta}$ and $\boldsymbol{\epsilon}$ follow normal distributions:

$$\boldsymbol{\delta} \sim \mathcal{N}(0, \boldsymbol{V}_\delta), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{V}_\epsilon), \tag{3}$$

where $\boldsymbol{V}_\delta \in \mathbb{R}^{p\times p}$ and $\boldsymbol{V}_\epsilon \in \mathbb{R}^{p\times p}$ are the covariance matrices associated with model discrepancy and measurement error, respectively. With this assumption, the likelihood function $l(\boldsymbol{z}|\boldsymbol{x})$ can be regarded as a Gaussian with zero mean and a covariance matrix of $\boldsymbol{V}_\delta + \boldsymbol{V}_\epsilon$ approximately and defined as follows

$$l(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}; \eta(\boldsymbol{x}), \boldsymbol{V}_\delta + \boldsymbol{V}_\epsilon). \tag{4}$$

Let $\pi(\boldsymbol{x})$ be the prior distribution of the model parameters. The posterior distribution, denoted by $p(\boldsymbol{x}|\boldsymbol{z})$, is thus:

$$p(\boldsymbol{x}|\boldsymbol{z}) \propto \pi(\boldsymbol{x}) l(\boldsymbol{z}|\boldsymbol{x}). \tag{5}$$

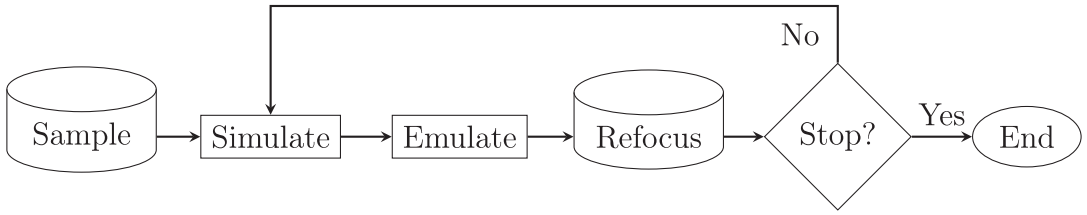

Fig. 1. History matching workflow.

2.2. History matching

History matching (HM) is a precalibration method designed for computationally expensive simulators [9,10,15]. Its primary goal is to iteratively discard *implausible* regions of the parameter space that are unlikely to yield model outputs consistent with experimental observations.

Fig. 1 illustrates the typical HM workflow. The process begins by generating an initial set of samples $\{\boldsymbol{x}_i\}_{i=1}^N$ from the prior distribution $\pi(\cdot)$. A uniform prior is typically chosen, and the initial samples are generated using a space-filling design such as the maximin Latin hypercube design [16]. The simulator is then evaluated to obtain outputs $\{\eta(\boldsymbol{x}_i)\}_{i=1}^N$. If the simulator is computationally expensive, the training runs $D = \{\boldsymbol{x}_i, \eta(\boldsymbol{x}_i)\}_{i=1}^N$ are used to train an emulator, which provides a computationally efficient approximation of the simulator. Being an approximation, emulation introduces an additional source of uncertainty known as *code uncertainty*.

History matching requires defining an *implausibility measure* and a corresponding cut-off value to identify parameter configurations that are not likely to provide a sensible match between model output and data. Samples with implausibility measures below the cut-off value are retained as *non-implausible*. A simple and effective sampling method, such as the p-variate random walk proposed in [9], can be used to generate new samples in each wave, a process known as refocussing. After excluding implausible regions, the remaining space is referred to as the *non-implausible space*. In subsequent waves, the simulator is re-evaluated to refine the non-implausible space. Refocussed training runs are used to update the emulator. The process repeats until a stopping criterion is met, such as when all remaining samples are deemed non-implausible according to the implausibility measure, or when the emulator's variance falls below some threshold. Gaussian process (GP) emulators [17] are a widely used choice in HM, as they naturally quantify code uncertainty. When using an emulator, the relationship between observations and emulator predictions can be expressed as:

$$\boldsymbol{z} = \mathbb{E}(\hat{\eta}(\boldsymbol{x})) + \gamma(\boldsymbol{x}) + \boldsymbol{\delta} + \boldsymbol{\epsilon}, \tag{6}$$

where $\hat{\eta}(\boldsymbol{x})$ is the emulator, $\mathbb{E}(\cdot)$ is the emulator posterior mean, $\gamma(\boldsymbol{x})$ represents code uncertainty, $\boldsymbol{\delta}$ accounts for model discrepancy, and $\boldsymbol{\epsilon}$ captures measurement error. The likelihood function $l'(\boldsymbol{z}|\boldsymbol{x})$ is updated as

$$l'(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}; \hat{\eta}(\boldsymbol{x}), \boldsymbol{V}_\epsilon + \boldsymbol{V}_\delta + \boldsymbol{V}_\gamma(\boldsymbol{x})), \tag{7}$$

where $\boldsymbol{V}_\epsilon, \boldsymbol{V}_\delta$, and $\boldsymbol{V}_\gamma(\boldsymbol{x})$ are covariance matrices associated with measurement error, model discrepancy, and code uncertainty, respectively. The resulting posterior for calibration becomes proportional to

$$\pi(\boldsymbol{x}) l'(\boldsymbol{z}|\boldsymbol{x}). \tag{8}$$

The implausibility measure for one-dimensional output can hence be defined as:

$$I(\boldsymbol{x}; z) = \frac{|z - \mathbb{E}(\hat{\eta}(\boldsymbol{x}))|}{\sqrt{V_\epsilon + V_\delta + V_\gamma(\boldsymbol{x})}},$$

A common choice for the cut-off value is 3, based on the 3-sigma rule [18]. For multidimensional outputs, a multivariate implausibility measure can be used [1]:

$$I(\boldsymbol{x}; \boldsymbol{z}) = (\boldsymbol{z} - \mathbb{E}(\hat{\eta}(\boldsymbol{x})))^\top \left(\boldsymbol{V}_\epsilon + \boldsymbol{V}_\delta + \boldsymbol{V}_\gamma(\boldsymbol{x})\right)^{-1} (\boldsymbol{z} - \mathbb{E}(\hat{\eta}(\boldsymbol{x}))), \tag{9}$$

where $\boldsymbol{V}_\epsilon + \boldsymbol{V}_\delta + \boldsymbol{V}_\gamma(\boldsymbol{x})$ is assumed invertible. Cut-off values are typically derived from the chi-squared distribution [1,9,10].

HM allows for simpler measures like the maximum implausibility at $\boldsymbol{x}$ in early waves:

$$I_M(\boldsymbol{x}; \boldsymbol{z}) = \max_j I(\boldsymbol{x}; z_j), \tag{10}$$

where $I(\boldsymbol{x}; z_j)$ is the one-dimensional implausibility for the jth output. This flexibility allows gradual refinement of the parameter space and reintroduction of challenging input–output combinations when they become feasible.

2.3. Importance sampling

Importance sampling is a method for approximating a target distribution by drawing samples from a simpler proposal distribution and assigning weights to each sample based on the ratio of the target distribution to the proposal distribution. This approach is especially effective when the target distribution is complex or computationally intensive to sample directly.

Bayesian History Matching (BHM) [10] leverages HM as a precursor to efficiently locate high-probability regions in the parameter space. Once these regions are identified, importance sampling can be performed to assign weights to the samples, providing a computationally efficient alternative to directly implementing a full Bayesian inference algorithm from scratch.

For the final non-implausible space detected by HM, an importance proposal distribution $q(\cdot)$ is used to generate samples. These samples are then re-weighted according to the following rule for $i = 1, \dots, N$:

$$w(\boldsymbol{x}'^{(i)}) = \frac{w^{un}\left(\boldsymbol{x}'^{(i)}\right)}{\sum_{k=1}^{N} w^{un}\left(\boldsymbol{x}'^{(k)}\right)}, \tag{11}$$

where $w^{un}\left(\boldsymbol{x}'^{(i)}\right)$ is the unnormalized weight of $\boldsymbol{x}'^{(i)}$, and $\boldsymbol{x}'^{(i)}$ represents the samples drawn from the proposal distribution $q(\cdot)$. The unnormalized weights are calculated as:

$$w^{un}\left(\boldsymbol{x}'^{(i)}\right) = \frac{\pi(\boldsymbol{x}'^{(i)}) \exp\left(-\frac{1}{2} I(\boldsymbol{x}'^{(i)}; \boldsymbol{z})\right)}{q(\boldsymbol{x}'^{(i)}) \sqrt{\det(\boldsymbol{V}_{\delta} + \boldsymbol{V}_{\epsilon} + \boldsymbol{V}_{\gamma}(\boldsymbol{x}'^{(i)}))}}, \tag{12}$$

where $\pi(\boldsymbol{x})$ is the prior distribution, and $I(\boldsymbol{x}; \boldsymbol{z})$ is the implausibility measure.

An effective choice of the importance proposal distribution, is a multivariate normal distribution [9,10]:

$$q(\boldsymbol{x}'^{(i)}) \sim \mathcal{N}\left(\boldsymbol{x}'^{(i)} \mid \hat{\mu}, \kappa\hat{\Sigma}\right), \tag{13}$$

where $\hat{\mu}$ and $\hat{\Sigma}$ represent the sample mean and covariance of the non-implausible samples from the final wave of HM, $\{\boldsymbol{x}_w^{(i)}\}_{i=1}^{N}$, respectively. The parameter κ is a hyperparameter that controls the convergence of the proposal distribution.

To ensure a final set of N equally weighted samples and improve estimation accuracy, [10] proposed a re-sampling step based on the normalized weights after importance sampling. This additional step enhances the alignment of the samples with the target posterior distribution.

Although importance sampling with re-sampling can effectively adjust the sample system from HM to approximate the true posterior, it has some limitations. A relatively large sample size is often required when the true posterior is highly concentrated or skewed, making it difficult to adapt the HM-generated samples efficiently in a single step. Furthermore, in a dynamical framework, BHM would require running the entire algorithm from scratch if new data becomes available, which prevents the reuse of previously computed information—even when the new data is similar to the existing observations.

To address these limitations and enable dynamic Bayesian calibration, we propose integrating a sequential Monte Carlo (SMC) sampler with history matching. This approach allows for the direct update of previously formed distributions as new data arrives, significantly improving computational efficiency and adaptability in real-time applications.

3. Sequential Monte Carlo sampler

3.1. Sequential Monte Carlo

Sequential Monte Carlo (SMC) [19] methods combine importance sampling with Monte Carlo techniques to address sequential sampling problems, that is, scenarios where the target distribution evolves over time or depends on a growing set of observations. Unlike sequential inference methods such as the Kalman filter [20], SMC does not depend on linearity or Gaussianity assumptions. Instead, it leverages information from the likelihood function to handle non-linear and non-Gaussian models effectively. Basic SMC algorithms sample from a sequence of probability distributions whose dimensionality increases over time. These distributions can represent the sequence of posterior distributions for a system's state path from time 1 to time t, conditioned on observations collected up to time t.

Each iteration of an SMC algorithm involves three essential steps [21]: (1) Sampling the new state from a proposal distribution; (2) Reweighting each particle; (3) Resampling particles. The resampling step is crucial to address the *weight degeneracy* problem [19], when the particle approximation becomes dominated by a few particles as time progresses, leading to poor representation of the target distribution. This occurs because particles with low weights at a given time step have a negligible chance of gaining importance in future steps. Resampling removes low-weight particles and replicates high-weight particles to maintain particle diversity and improve approximation accuracy.

Basic SMC algorithms commonly use traditional resampling techniques, such as multinomial resampling [22], which generate N equally weighted particles after each resampling step. However, as time progresses, the discrepancy between the importance proposal and the target distribution increases, and successive resampling steps can deplete the diversity of earlier particle paths, causing the *path degeneracy* problem. To address this, advanced resampling techniques such as adaptive resampling [23], dynamic resampling [24], and rejection resampling [25] have been proposed. These methods perform resampling only when necessary, thereby improving estimation accuracy.

In the algorithm proposed in this paper, we use rejection resampling because it does not require hyperparameter tuning. This makes it more straightforward and efficient to implement.

3.2. Sequential Monte Carlo sampler

Building on SMC methods, [26] introduced the Sequential Monte Carlo sampler, a methodology for sampling from a sequence of distributions defined on a common space, such as a series of bridging distributions between a prior and its corresponding posterior. The SMC sampler constructs a sequence of artificial joint target distributions whose marginals correspond to the desired target distributions at each time step. This framework allows the application of standard SMC methods to approximate these targets effectively.

The artificial joint distribution at time t is constructed using an auxiliary variable technique and backward Markov kernels L_{t-1} with densities $L_{t-1}(\boldsymbol{x}_t, \boldsymbol{x}_{t-1})$. The joint distribution is defined as:

$$\widetilde{\pi}_t(\boldsymbol{x}_{1:t}) = \widetilde{p}_t(\boldsymbol{x}_{1:t}), \tag{14}$$

where

$$\widetilde{p}_t(\boldsymbol{x}_{1:t}) = p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)\prod_{i=1}^{t-1} L_i(\boldsymbol{x}_{i+1}, \boldsymbol{x}_i),$$

and its marginal $\int \widetilde{p}_t(\boldsymbol{x}_{1:t})\mathrm{d}\boldsymbol{x}_{1:t-1}$ recovers the target $p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)$. Here, L_i are backward Markov kernels, and $\bar{p}_t(\boldsymbol{x}_t|\boldsymbol{z}_t)$ denotes the unnormalized part of $p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)$. The joint distribution can also be expressed as:

$$\widetilde{\pi}_t(\boldsymbol{x}_{1:t}) = \frac{\bar{p}_t(\boldsymbol{x}_t|\boldsymbol{z}_t)\prod_{i=1}^{t-1} L_i(\boldsymbol{x}_{i+1}, \boldsymbol{x}_i)}{Z_t}, \tag{15}$$

where Z_t is the normalizing constant of $p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)$.

At time step t, given a set of weighted particles $\{W_{t-1}^{(i)}, X_{t-1}^{(i)}\}_{i=1}^N$ from time $t-1$, the importance proposal for the path $\boldsymbol{x}_{1:t}$ is denoted as $\widetilde{q}_t(\boldsymbol{x}_{1:t})$. A forward Markov kernel K_t with density $K_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t)$ is applied to move the particles. The unnormalized weights are computed as:

$$w_t(\boldsymbol{x}_{1:t}) = W_{t-1}(\boldsymbol{x}_{1:t-1})\widetilde{w}_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t), \tag{16}$$

$$\widetilde{w}_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t) = \frac{\bar{p}_t(\boldsymbol{x}_t)L_{t-1}(\boldsymbol{x}_t, \boldsymbol{x}_{t-1})}{\bar{p}_{t-1}(\boldsymbol{x}_{t-1})K_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t)}, \tag{17}$$

where $\widetilde{w}_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t)$ is the incremental weight. The optimal importance proposal is the target distribution at time t, but this is generally impractical to sample from directly.

MCMC can be used to define K_t, ensuring it has the target distribution at time t as its invariant distribution. This strategy is effective if K_t mixes rapidly or if adjacent target distributions are similar. The corresponding backward kernel L_{t-1} can be defined as:

$$L_{t-1}(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}) = \frac{p_t(\boldsymbol{x}_{t-1}|\boldsymbol{z}_{t-1})K_t(\boldsymbol{x}_{t-1}, \boldsymbol{x}_t)}{p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)}, \tag{18}$$

and the unnormalized weight becomes:

$$w_t(\boldsymbol{x}_{1:t}) \propto W_{t-1}(\boldsymbol{x}_{1:t-1})\frac{\bar{p}_t(\boldsymbol{x}_t)}{\bar{p}_{t-1}(\boldsymbol{x}_{t-1})}. \tag{19}$$

The SMC sampler offers significant advantages over traditional MCMC methods due to its flexibility. It allows customization of initial sampling, intermediate distributions, and importance proposals. For instance, a well-designed sequence of intermediate distributions can mitigate weight degeneracy, while an effective importance proposal improves exploration of the target distribution. Additionally, SMC samplers are naturally parallelisable because particles are independent during sampling, making them computationally efficient.

4. Sequential Bayesian history matching

This paper introduces Sequential Bayesian History Matching (SBHM), a framework that combines SMC samplers with HM under the Bayesian perspective to address the challenge of model updating when a constant stream of data is available. To achieve robust initial sampling, we propose reliability-based HM with SMC to identify the non-implausible space in the first time step. Once initial non-implausible samples are obtained from an importance proposal, they are transitioned to the true posterior of the first set of observations by applying the SMC sampler with intermediate distributions. For subsequent observations, the SMC sampler iteratively updates particles from the previous posterior to the new posterior, ensuring that previously acquired information is utilized. This avoids running BHM from the beginning every time new data is observed, to avoid wasteful computation.

This section provides a detailed description of the target posteriors in dynamic Bayesian calibration and the technical components of SBHM, including initial sampling, intermediate distributions, stopping rules, and importance proposals.

4.1. Target posteriors

In a scenario when the model parameters evolve in time, one can leverage the information from the samples at the current time step to update the parameters for the next step, based on new data. At time t, the relationship between an observation and the output of the emulator is analogous to Eq. (6), and is expressed as:

$$\boldsymbol{z}_t = \mathbb{E}(\hat{\eta}_t(\boldsymbol{x}_t)) + \gamma(\boldsymbol{x}_t) + \boldsymbol{\delta}_t + \boldsymbol{\epsilon}_t, \tag{20}$$

All sources of uncertainty are assumed to follow normal distributions. A sequence of posterior distributions conditioned on the observations at each time t, namely $\{p_t(\boldsymbol{x}_t|\boldsymbol{z}_t)\}_{t=1}^{T}$. can be constructed applying Bayes' theorem.

Even in the absence of explicit prior knowledge about the evolution of the unknown parameters, sequential sampling can efficiently update the previously formed posterior distribution. This is particularly effective in scenarios where the new observation $\boldsymbol{z}_t$ is strongly correlated with the prior state $\boldsymbol{z}_{t-1}$.

4.2. Reliability-based history matching

One of the main challenges posed by HM, especially for complex high-dimensional simulators, is that the non-implausible space can be orders of magnitude smaller than the original parameter space [1]. This means that, at every new wave, producing non-implausible samples in a progressively shrinking space becomes progressively more difficult. To address this, a subset simulation (SuS)-based HM approach was proposed in [27]. SuS is an advanced Monte Carlo algorithm originally developed in [28] for rare event estimation in engineering reliability analysis. It samples from failure events by designing a sequence of nested conditional events.

SuS can be seen as a special case of a Sequential Monte Carlo (SMC) sampler [29]. Instead of conditional distributions between failure events, the intermediate target distributions are set as the distributions on the failure events themselves.

At each wave w, assume an implausibility measure I_w with a corresponding cut-off value c_w. To efficiently identify the non-implausible space, we define a sequence of intermediate distributions $\{\mathbb{1}(I_{w,l} < c_{w,l})\}_{l=1}^{L}$, where $c_w = c_{w,L} < \cdots < c_{w,1}$. To move samples between failure events, we use iterations of the adaptive Metropolis algorithm in the SMC-based SuS algorithm proposed in [29]. The pseudo-code for the modified SuS-based HM is given in Algorithm 1.

4.3. Intermediate distributions

After the first experimental data is observed, SuS-based HM is run to detect the corresponding non-implausible space. After that, a sequence of intermediate distributions is constructed, leading to the posterior:

$$p_{1,r}(\boldsymbol{x}_1; \boldsymbol{z}_1) \propto \pi(\boldsymbol{x}_1)^{1-\varphi_{1,r}} \bar{p}_1(\boldsymbol{x}_1)^{\varphi_{1,r}}, \tag{21}$$

where $r = 0, 1, \ldots, r_1$ and $0 = \varphi_{1,0} < \cdots < \varphi_{1,r_1} = 1$.

These intermediate distributions enable a smooth transition from a relatively flat distribution (*e.g.*, the prior over the non-implausible space) to the final posterior distribution at time 1. Performing importance sampling sequentially with such intermediate distributions helps ensure convergence, even with a relatively small sample size.

When new data arrives, the corresponding intermediate distributions can be designed to move particles from time $t-1$ to time t if the new observation is highly correlated with the previous information:

$$p_{t,r}(\boldsymbol{x}_t; \boldsymbol{z}_t) \propto \bar{p}_{t-1}(\boldsymbol{x}_{t-1})^{1-\varphi_{t,r}} \bar{p}_t(\boldsymbol{x}_t)^{\varphi_{t,r}}, \tag{22}$$

where $r = 0, 1, \ldots, r_t$ and $0 = \varphi_{t,0} < \cdots < \varphi_{t,r_t} = 1$.

These intermediate distributions gradually approach the posterior at time t, facilitating efficient particle movement. To achieve this, the tempering hyperparameters $\{\varphi_{t,r}\}_{r=0}^{r_t}$ are adjusted by controlling their rates $\{s_{t,r} = \varphi_{t,r} - \varphi_{t,r-1}\}_{r=1}^{r_t}$, also referred to as the progress rates, where $\varphi_{t,0} = 0$ [30].

For the initial rate $s_{t,1}$, a high value (*e.g.*, 1) is set, and it decreases until the effective sample size (ESS) [31] exceeds a predefined lower bound, that is:

$$\mathrm{ESS}_{t,1}(\varphi_{t,1}) = \left[\sum_{i=1}^{N} \left(W_t^{(i)}(\varphi_{t,1})\right)^2\right]^{-1} > \alpha N, \tag{23}$$

where $W_t^{(i)}(\varphi_{t,1})$ denotes the normalized weight associated with $\varphi_{t,1}$. The hyperparameter $\alpha \in (0, 1)$ is chosen by the user to control the quality of the corresponding ESS [31,32]. Since MCMC methods typically produce correlated samples, the ESS is a measure to approximate the number of samples needed to produce a sample of independent draws which contains the equivalent amount of information.

In theory, if the particles are moved smoothly, maintaining a constant progress rate ensures that the ESS remains within a reasonable range [30]. For subsequent iterations, the rate is kept constant unless:

- The tempering hyperparameter $\varphi_{t,r}$ reaches or exceeds 1, or

Algorithm 1 SuS-Based History Matching

Require: Experimental observation z, sample size N, prior distribution $\pi(\boldsymbol{x})$ for the unknown parameters, cut-off value c_w and level probability p in SuS.

1. **Initialisation:** Set wave counter to $w = 1$. Draw N initial samples $\left\{\boldsymbol{x}_{1,1}^{(i)}\right\}_{i=1}^{N}$ from the prior $\pi(\cdot)$. Train an initial Gaussian process emulator GP_1 using the training runs $\mathcal{D} = \{\boldsymbol{x}_i, \eta(\boldsymbol{x}_i)\}_{i=1}^{N}$.
2. **Iterate until stopping rule:**
   (a) Set SuS level $l = 1$ and cut-off value $c_{w,1} = +\infty$.
   (b) **While intermediate cut-off value $c_{w,l}$ is greater than c_w:**
      i. **Update intermediate cut-off value:** Rank samples in increasing order based on their implausibility. Denote the reordered samples as $I_w\left(\boldsymbol{x}_{w,l}^{(1)}; \boldsymbol{z}\right) \leq \cdots \leq I_w\left(\boldsymbol{x}_{w,l}^{(N)}; \boldsymbol{z}\right)$. Define the intermediate cut-off value $c_{w,l}$ as the midpoint between $I_w\left(\boldsymbol{x}_{w,l}^{(Np)}; \boldsymbol{z}\right)$ and $I_w\left(\boldsymbol{x}_{w,l}^{(Np+1)}; \boldsymbol{z}\right)$.
      ii. **Reweight:** Compute weights:
         • If $l > 1$:
$$w_{w,l}^{(i)} \propto \frac{\mathbb{1}\left(I_w\left(\boldsymbol{x}_{w,l}^{(i)}; \boldsymbol{z}\right) < c_{w,l}\right)}{\mathbb{1}\left(I_w\left(\boldsymbol{x}_{w,l}^{(i)}; \boldsymbol{z}\right) < c_{w,l-1}\right)}.$$
         • Otherwise:
$$w_{w,l}^{(i)} \propto \frac{\mathbb{1}\left(I_w\left(\boldsymbol{x}_{w,l}^{(i)}; \boldsymbol{z}\right) < c_{w,l}\right)}{\mathbb{1}\left(I_{w-1}\left(\boldsymbol{x}_{w-1,L_w}^{(i)}; \boldsymbol{z}\right) < c_{w-1,L_w}\right)}.$$
         where $\mathbb{1}(\cdot)$ is an indicator function.
      iii. **Resample:** Perform rejection resampling to obtain equally weighted particles $\left\{\boldsymbol{x}_{w,l}^{(i)}\right\}_{i=1}^{N}$.
      iv. **Sampling:** Move particles using a Markov kernel K, whose invariant distribution is $\pi(\boldsymbol{x})\mathbb{1}\left(I_w\left(\boldsymbol{x}_{w,l}^{(i)}; \boldsymbol{z}\right) < c_{w,l}\right)$, i.e., $\boldsymbol{x}_{w,l+1}^{(i)} \sim K\left(\boldsymbol{x}_{w,l}^{(i)}, \cdot\right)$.
      v. Set $l = l + 1$.
   (c) Update the emulator GP_w using the training runs $\mathcal{D} = \{\boldsymbol{x}_{w,l}^{(i)}, \eta(\boldsymbol{x}_{w,l}^{(i)})\}_{i=1}^{N}$. Set $w = w + 1$, and update the implausibility measure I_w and cut-off value c_w for wave w.
3. **Output:** N samples $\left\{\boldsymbol{x}_{w-1,l}^{(i)}\right\}_{i=1}^{N}$.

• The ESS falls outside the predefined range:

$$\mathrm{ESS}_{t,r}(\varphi_{t,r}) = \left[\sum_{i=1}^{N}\left(W_t^{(i)}(\varphi_{t,r})\right)^2\right]^{-1} \in [\alpha N, \beta N], \tag{24}$$

where $\alpha < \beta$ are thresholds, and $W_t^{(i)}(\varphi_{t,r})$ is the normalized weight for $\varphi_{t,r}$.

If the tempering hyperparameter $\varphi_{t,r}$ reaches or exceeds 1, we set $s_{t,r} = 1 - \varphi_{t,r-1}$. Otherwise:

- If the ESS falls below the lower threshold αN, the tempering parameter $\varphi_{t,r}$ is adjusted such that $\mathrm{ESS}_{t,r}(\varphi_{t,r}) \approx \alpha N$, using iterative bisection on $(0, s_{t,r-1}]$.
- If the ESS exceeds the upper threshold βN, the parameter $\varphi_{t,r}$ is adjusted such that $\mathrm{ESS}_{t,r}(\varphi_{t,r}) \approx \beta N$, using iterative bisection on $(s_{t,r-1}, 1 - \varphi_{t,r-1}]$.

This adaptive adjustment ensures efficient particle movement while maintaining diversity and accuracy.

4.4. Stopping rules

When new data is obtained, it may be highly uncorrelated to the previous one. In such cases, SMC samplers may require more computational effort than starting HM anew. To decide whether to proceed with SMC or reinitiate HM, we propose a stopping rule based on the ESS.

Assume that at time $t-1$, we have a set of weighted particles $\{W_{t-1}^{(i)}, X_{t-1}^{(i)}\}_{i=1}^{N}$, where $W_{t-1}^{(i)}$ is the normalized weight of the ith particle. To evaluate the performance of these particles for the posterior distribution associated with the new observation at time t,

the ESS is calculated as:

$$\mathrm{ESS}_t = \left[\sum_{i=1}^{N}\left(W_t^{(i)}\right)^2\right]^{-1} \in [1, N], \tag{25}$$

where $W_t^{(i)}$ is the normalized weight of the ith particle with respect to the posterior distribution at time t. A higher ESS indicates that the particle system provides a more efficient representation of the target distribution.

If the ESS falls below a threshold $N_{min} = p_{min}N$, where $p_{min} \in (0, 1)$ is a user-defined parameter, starting HM anew is preferred. On the other hand, if the ESS meets or exceeds this threshold, the previous and new posteriors are considered sufficiently similar to justify continuing with the SMC sampler to move particles toward the new posterior distribution.

4.5. Importance proposals

An essential component of an SMC sampler is the choice of the importance proposal. Once samples from the non-implausible space are identified by HM, a multivariate normal distribution $\mathcal{N}(\cdot|\hat{\mu}, \kappa\hat{\Sigma})$ can be used as the initial importance proposal, as suggested by [29]. Here, $\hat{\mu}$ and $\hat{\Sigma}$ are the sample mean and covariance matrix, respectively, of the non-implausible samples obtained from the last wave of HM, and κ is a scaling factor (*e.g.*, $\kappa = 3$) that controls the proposal spread.

When new data arrives, if the new posterior contains isolated or disconnected modes compared to the previous posterior, more complex MCMC proposals that allow transitions between modes, such as the mode-jumping proposal by [33], can be used. Otherwise, simpler and computationally efficient proposals, such as a standard random walk or independent Metropolis–Hastings, are suitable. In scenarios where the posterior at each time step is expected to have a single mode, we use adaptive Metropolis–Hastings as outlined in [29] to move particles efficiently.

The pseudo-code of sequential Bayesian History Matching is given in Algorithm 2.

Algorithm 2 Sequential Bayesian History Matching

Require: Experimental observations $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_t, \ldots$, sample size N.

1. **Initialisation:** Set $t = 1$.
   (a) Run the modified SuS-based HM algorithm (Algorithm 1) for the observation $\boldsymbol{z}_t$.
   (b) Store $\{\boldsymbol{x}_{t,0}^{(i)}\}_{i=1}^{N}$, the N particles in the final non-implausible space for this observation.
2. **As new data arrives:**
   (a) Set $t = t + 1$ and compute the Effective Sample Size (ESS) for the new observation using Equation (25).
   (b) If the ESS is below the threshold N_{min}, go to Step 1 to initiate a new HM. Otherwise, proceed with Step 2(c) for the new observation $\boldsymbol{z}_t$.
   (c) Set $r = 0$, $\varphi_{t,-1} = 0$, and initialize $\boldsymbol{x}_{t,0}^{(i)} \leftarrow \boldsymbol{x}_{t-1,0}^{(i)}$. Train an initial Gaussian process emulator $GP_{t,0}$ for the observation $\boldsymbol{z}_t$ using a subset of the dataset $\mathcal{D} = \{\boldsymbol{x}_{t,0}^{(i)}, \eta_t(\boldsymbol{x}_{t,0}^{(i)})\}$.
   (d) **Repeat until** $\varphi_{t,r-1} = 1$**:**
      i. **Hyperparameter adaptation:** Determine the tempering hyperparameter $\varphi_{t,r}$ and the next intermediate distribution by controlling the ESS.
      ii. **Reweight:** Reweight the particles $\{\boldsymbol{x}_{t,r}^{(i)}\}_{i=1}^{N}$ for the intermediate distribution.
      iii. **Resample:** Perform rejection resampling to obtain new equally weighted particles $\{\boldsymbol{x}_{t,r}^{(i)}\}_{i=1}^{N}$.
      iv. **MCMC Mutation:** Move particles using a Markov kernel K, whose invariant distribution matches the current target:

$$\boldsymbol{x}_{t,r+1}^{(i)} \sim K(\boldsymbol{x}_{t,r}^{(i)}, \cdot), \quad i = 1, \ldots, N.$$

      v. **Update Emulator:** Increment $r = r + 1$ and retrain the Gaussian process emulator $GP_{t,r}$ with a subset of the updated dataset $\mathcal{D} = \{\boldsymbol{x}_{t,r}^{(i)}, \eta_t(\boldsymbol{x}_{t,r}^{(i)})\}_{i=1}^{N}$.
3. **Output:** Return the final set of particles $\{\boldsymbol{x}_{t,r}^{(i)}\}_{i=1}^{N}$.

5. Numerical experiments

To demonstrate the broad utility of the proposed algorithm, this section presents two numerical case studies from distinct domains, engineering and healthcare. The engineering case study includes a simple three-degree-of-freedom mass–spring system using simulated data. This example illustrates the advantages of SBHM and the role of the key hyperparameter α. In the healthcare case study, a patient-specific respiratory model with read data is presented. Since this is a model with a more complex structure and a greater number of unknown parameters, it demonstrates the potential of SBHM in real-world applications.

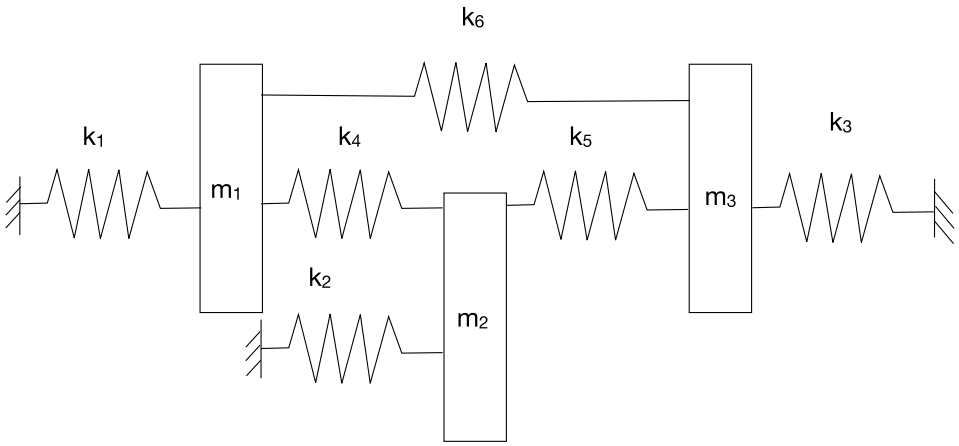

Fig. 2. Three degree-of-freedom mass–spring.

Table 1
Model fits (output with estimated parameters) of HM and HM-SMC for the initial observation; N = 1000. Units are rad/s.

Natural frequencies	True	Observed	Initial	HM	HM-SMC
1	0.1600	0.1598	0.1805	0.1584	0.1606
2	0.3200	0.3198	0.3240	0.3181	0.3187
3	0.4500	0.4523	0.4079	0.4525	0.4525

Table 2
Parameter fits (mean ± standard deviation) of HM and HM-SMC for the initial observation; N = 1000. Units are N/m.

Parameter	True	Initial	HM	HM-SMC
k_1	1	2.0009 (± 0.1933)	0.9724 (± 0.0726)	1.0557 (± 0.1067)
k_6	3	1.9984 (± 0.1979)	3.0479 (± 0.0522)	3.0268 (± 0.0609)

5.1. Three degree-of-freedom mass-spring system

The three-degree-of-freedom mass–spring system introduced in [6] is used here, see Fig. 2. The true values of parameters for this system are $m_i = 1.0\,\text{kg}$ $(i = 1, 2, 3)$, $k_i = 1.0\,\text{N/m}$ $(i = 1, 2, 3, 4, 5)$ and $k_6 = 3.0\,\text{N/m}$. The value of k_1 and k_6 are assumed as unknown and a prior $\mathcal{N}(2.0, 0.04)$ N/m is defined for both k_1 and k_6. The nominal values of the natural frequencies of the three modes of the system are $\omega_1 = 0.16\,\text{rad/s}$, $\omega_2 = 0.32\,\text{rad/s}$ and $\omega_3 = 0.45\,\text{rad/s}$. The measurement error and model discrepancy are assumed to have Gaussian distributions with zero mean and 0.001 standard derivation. An initial observation is generated by adding measurement error and model discrepancy to the true values of the natural frequency. History matching and History matching with sequential Monte Carlo sampler (HM-MSC), *i.e.*, the initialization of SBHM, were performed to generate samples for the unknown parameters k_1 and k_6. Mean of these samples in each algorithm is chosen as its corresponding estimated parameters.

The natural frequency, parameters estimation and distributions of samples for the initial observation are shown in Table 1, Table 2 and Fig. 3 , respectively.

From Table 1, it can be seen that compared with the output of the model with initial parameters, the output with calibrated parameters produced by HM or HM-SMC is more aligned with observed (and true target) data. Tables 1 and 2 show that HM-SMC can reset the samples from HM to further approach the target data. Moreover, Fig. 3 compares the kernel density estimate from HM with the marginal distribution produced by HM-SMC. While HM identifies high-probability regions, its samples do not provide a smooth target posterior distribution, which should be approximately Gaussian in this example. On the other hand, HM-SMC not only achieves a more accurate point estimate but also yields posterior samples.

Additionally, Fig. 4 presents a kernel density estimation of the final distributions of three natural frequencies f_1, f_2 and f_3, which are considerably narrower and closer to the target range than the prior distributions.

During their operational lifetime spring stiffnesses can vary, for instance, due to prolonged static loads, cyclic fatigue, and other mechanical factors. The changes are particularly pronounced if the system is regularly exposed to elevated temperatures and extreme environments. Changes can also result from wear and tear, including operation outside of nominal conditions and damage. Therefore, it is important to not treat the unknown stiffness parameters as fixed. Thus, efficient updating in the presence of streaming data is crucial. To illustrate how SBHM can deal with such situations, two scenarios are considered: one in which the stiffnesses undergo small changes and another in which the changes are comparatively larger. In Case 1, the new true values of k_1 is updated to $0.95\,\text{N/m}$ while k_6 remains unchanged. In Case 2, the new true values of k_1 and k_6 are updated to $0.9\,\text{N/m}$ and $2.9\,\text{N/m}$, respectively.

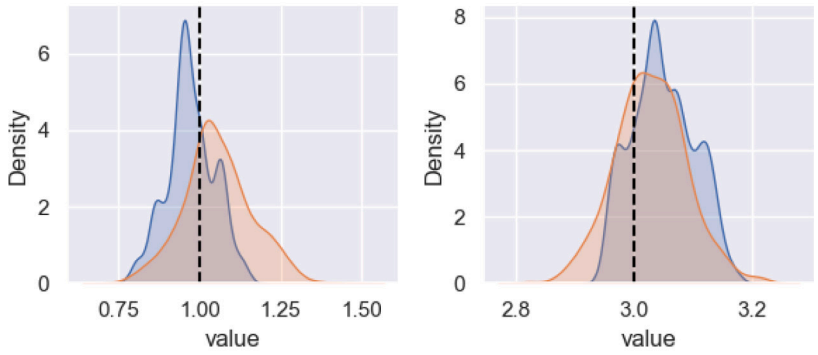

Fig. 3. The kernel density estimate for parameters k_1 (left) and k_6 (right) produced by the last iteration of HM (blue fill) and the marginal distribution for parameters produced by HM-SMC (orange fill) for the initial observation. The dotted black lines denote the true value of parameters; $N = 1000$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

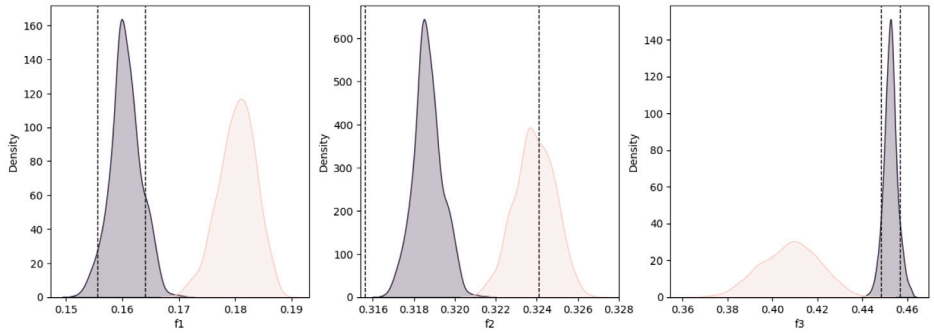

Fig. 4. The kernel density estimation of the initial output distribution produced by the first iteration of HM (light), and that one generated by HM-SMC (dark) for the initial observation. Dashed lines in Figures show the target range..

Table 3
Case 1: Model fits (output with estimated parameters) of HM-SMC and SBHM with different α for the new observation; N = 1000. Units are rad/s.

Natural frequencies	True	Observed	HM-SMC	SBHM (0.85)	SBHM (0.9)
1	0.1578	0.1579	0.1577	0.1581	0.1579
2	0.3180	0.3178	0.3180	0.3180	0.3180
3	0.4495	0.4500	0.4499	0.4501	0.4498

Table 4
Case 1: Parameter fits (mean ± standard deviation) of HM-SMC and SBHM with different α for the new observation; N = 1000. Units are N/m.

Parameter	True	HM-SMC	SBHM (0.85)	SBHM (α : 0.9)
k_1	0.95	0.9458 (± 0.1028)	0.9605 (± 0.1141)	0.9531 (± 0.0991)
k_6	3.0	3.0090 (± 0.0626)	3.0081 (± 0.0664)	3.0061 (± 0.0691)

For both cases, parameter updates were estimated by using a new HM-SMC and SBHM with a stopping threshold of $p_{min} = 0.7$. The natural frequency and parameters estimation for Case 1 are shown in Tables 3 and 4, respectively. Similarly, the natural frequency and parameters estimation for Case 2 are shown in Tables 5 and 6, respectively. The average computational time of the two algorithms for different cases are shown in Table 7.

Combining Table 3, 4, 5 and 6, we can observe that updating previously formed distributions via SBHM can achieve comparable ($\alpha = 0.85$) or even better ($\alpha = 0.9$) accuracy compared with running a new HM-SMC from scratch for a new observation under both smaller and larger changes. Table 7 demonstrates that SBHM can outperform new HM-SMC runs in terms of speed by leveraging previous information. Higher α requires SBHM to achieve higher effective sample size at each iteration and hence gets better accuracy but increases the computational cost of SBHM, while lower α reduces its accuracy (sometimes slightly worse than running a new HM-SMC) but save time. Thus, users should tune α to balance accuracy and analysis time based on their needs. Moreover, Table 7 displays the average computational time of SBHM and HM-SMC for different changes on parameters. The results indicate that

Table 5
Case 2: Model fits (output with estimated parameters) of HM-SMC and SBHM with different α for the new observation; N = 1000. Units are rad/s.

Natural frequencies	True	Observed	HM-SMC	SBHM (0.85)	SBHM (0.9)
1	0.1564	0.1548	0.1558	0.1556	0.1560
2	0.3176	0.3190	0.3178	0.3175	0.3175
3	0.4431	0.4443	0.4443	0.4438	0.4444

Table 6
Case 2: Parameter fits (mean ± standard deviation) of HM-SMC and SBHM with different α for the new observation; N = 1000. Units are N/m.

Parameter	True	HM-SMC	SBHM (0.85)	SBHM (0.9)
k_1	0.9	0.8761 (± 0.1054)	0.8713 (± 0.1258)	0.8848 (± 0.1109)
k_6	2.9	2.9267 (± 0.0583)	2.919 (± 0.0724)	2.9260 (± 0.0630)

Table 7
Average computational time of SBHM and HM-SMC for different changes on parameters with different α; N = 1000. Units are s.

Case	HM-SMC	SBHM (0.85)	SBHM (0.9)
1	45.3	14.5	21.9
2	43.8	21.3	37.8

Table 8
Parameter ranges for the lung model. The units for resistors and capacitors are (mmHg*s)/mL and mL/mmHg, respectively.

	R_1	R_2	R_3	R_4	C_1	C_2	C_3	C_4
Lower bound	0.001	0.001	0.001	0.001	1.5	1.5	2	50
Upper bound	0.004	0.004	0.004	0.005	3	3	6	200

larger parameters changes lead to higher computational costs since SBHM needs more iterations to capture these changes. Setting an appropriate stopping threshold is, therefore, critical for maintaining the efficiency of the algorithm. When new data becomes nearly irrelevant to previous data, it would be more efficient to run a new HM-SMC compared with continuing with the current SMC sampler in SBHM.

5.2. A clinical model

We update an 8-parameter version of the patient-specific cardio-respiratory model for intensive care unit (ICU) patients as presented in [34]. The model is flexible and can be modified to suit various clinical scenarios. For this study, we refine the respiratory system component, focusing specifically on the lung. The model incorporates three key components of the lung: trachea, bronchi, and alveoli, which are commonly described in the literature [35–37]. Additionally, a ventilation component is introduced to address the needs of ICU patients.

Each component is modeled with one resistor and one capacitor, resulting in a total of eight unknown parameters. The initial ranges of these parameters are provided in Table 8. In our fluid system, resistors directly control flow rates, while capacitors temporarily store and release pressurized fluid, indirectly influencing the flow.

The lung simulator takes an observed pressure curve as input and generates a flow curve as output, which can be monitored by clinicians at the ICU. Thus, for this example, the model is updated every time a new observation (that is a new curve) is observed. Based on prior beliefs provided by the modeler, the measurement error and model discrepancy for each time point of a flow curve are both set to 50 mL/s.

The multivariate normal distribution proposed by [9] is not used as the importance proposal following HM because the values of resistors and capacitors must remain positive. Instead, we adopt a uniform prior $\mathcal{U}(\cdot|\hat{\mu}, \kappa\hat{\Sigma}) \cap \mathcal{X}$, where $\kappa = 3$. GP emulators for curves, as described in [38], are used. Additionally, when new observations are introduced, the dimension of the output changes because the length of breath varies. Thus, two GP emulators are trained at each wave of SBHM: one for the previous observation and one for the new observation. An HM-SMC procedure, representing the initialization step of the SBHM algorithm, is used to obtain posterior samples for the initial observation. At each iteration, a maximin strategy [9], which selects points by maximizing the minimum distance among the current system of samples, is used to generate the training set for the GP emulators.

Fig. 5 illustrates the shrinking process in the parameter space for SuS-based HM. After three waves, the average standard deviation of the Gaussian process emulator on the validation dataset fell below that of the measurement error for a single point (*i.e.*, 50 mL/S). At this point, the algorithm terminates, outputting samples from the final non-implausible space. In the first wave, uniform samples within the ranges specified in Table 8. The maximum implausibility measure in Eq. (10) with a threshold of 6 was used during the second wave to eliminate regions, whilst the multivariate implausibility measure in Eq. (9) with a threshold of 3 was employed in the final wave to refine the non-implausible space further. Fig. 5 also shows how the final non-implausible space is significantly smaller than the original parameter space.

An SMC sampler was performed following HM to obtain posterior samples, with the mean of these samples treated as parameter estimates. Fig. 6 compares the model outputs using initial parameter estimates from the prior samples and those derived from

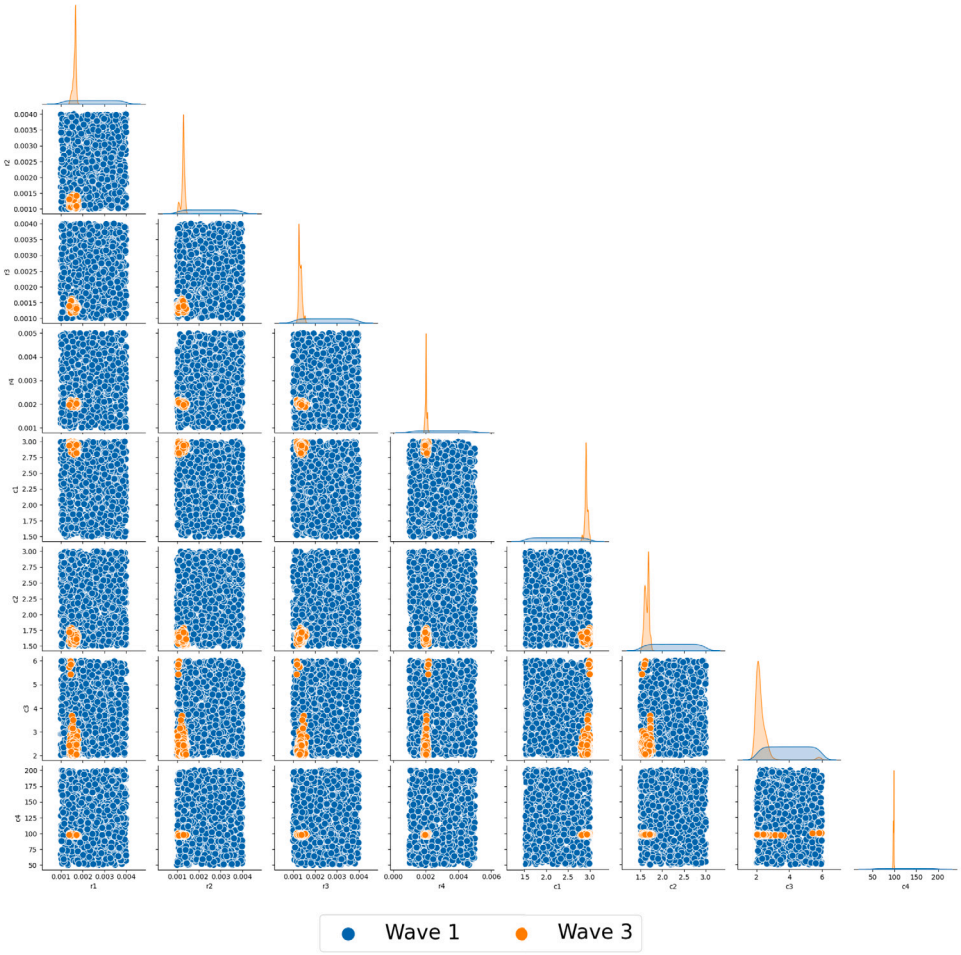

Fig. 5. Paired scatterplots for HM waves 1 and 3. The plots show the dramatic shrinkage of the parameter space. Initial samples, $N = 1000$.

posterior samples generated at the last iteration of HM-SMC. The unknown parameters primarily influence the global shape of the estimated flow curves, while local noise in these curves is determined by the input pressure. Fig. 6 shows that most of the model output lies within the defined uncertainty bounds. Moreover, the calibrated parameters obtained through HM-SMC produce outputs that align more closely with the observed flow compared to outputs generated with initial parameters. However, the observed mismatch near the minimum suggests that the model discrepancy in this region may have been underestimated. One potential explanation for this mismatch is the unaccounted offset between self-breathing and ventilator-induced breathing periods.

When a new flow curve is observed, there are two options: either run SBHM or run HM-SMC anew. Fig. 7 shows the calibrated outputs produced by both algorithms for the second observation (*i.e.* second curve). It can be seen that both methods delivered parameter configurations whose output matches the observed curve. This demonstrates that SBHM is capable of updating model parameters relying on information from previous observations. Moreover, the usage of one algorithm over the other one can be justified in terms of their computational complexity.

The computational complexity of the SuS-based HM with adaptive Metropolis is $\mathcal{O}(NLS)$, where L is the sum of the number of intermediate thresholds across waves ($L = \sum_{i=1}^{W} L_i$), and S is the number of adaptive Metropolis iterations. For an SMC sampler with rejection resampling, the complexity is $\mathcal{O}(NRS)$, where R is the number of intermediate distributions. After parallelization, the time complexity of SuS-based HM reduces to $\mathcal{O}(LS)$, while that of the SMC sampler with rejection resampling becomes $\mathcal{O}(RS)$. When the new observation is closely related to the previous one, the number of iterations R required to update parameters in SBHM is small. In such cases, SBHM requires less computational time compared to running a new HM-SMC. Experimental results confirm that SBHM achieves similar or better accuracy compared to HM-SMC, while consistently requiring less computational time when stopping rules are not met.

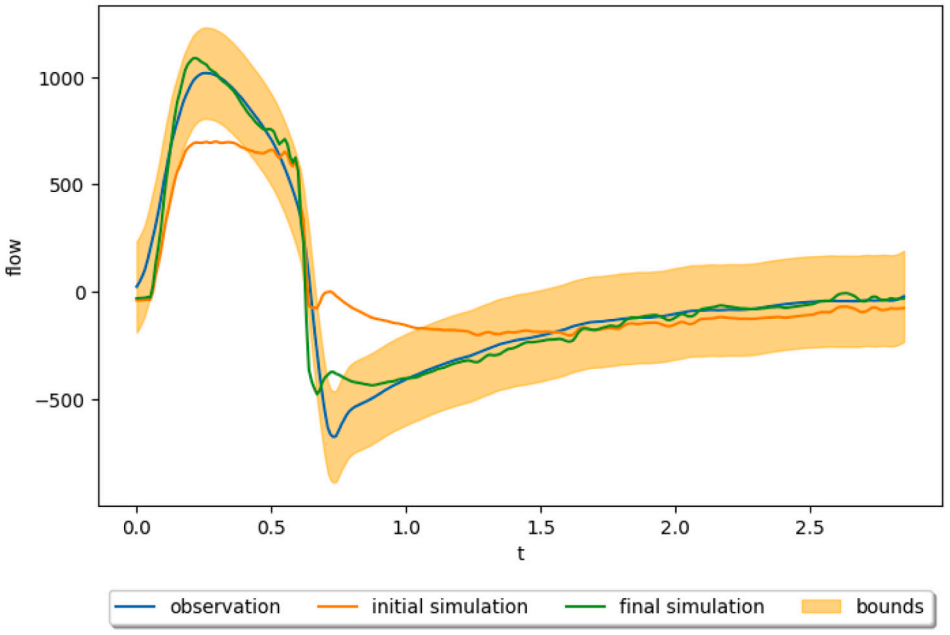

Fig. 6. Model fits (output with estimated parameters) for the first and last iterations of HM-SMC for the 1st breath, error bounds of the observation: $\pm 3\sqrt{V_{\delta,1} + V_{\epsilon,1}}$, $N = 1000$.

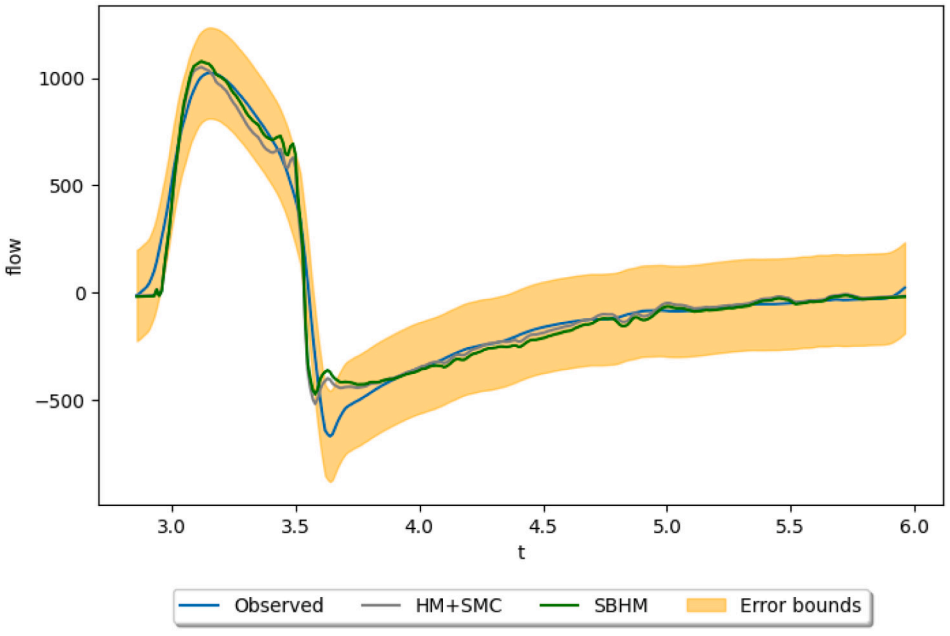

Fig. 7. Model fits of SBHM and HM-SMC for the 2nd breath, error bounds of the observation: $\pm 3\sqrt{V_{\delta,2} + V_{\epsilon,2}}$, $N = 1000$.

6. Discussion

This paper proposes an algorithm for sequential Bayesian history matching (SBHM), designed for dynamic model updating. The algorithm integrates two key techniques: history matching (HM) and sequential Monte Carlo (SMC) samplers. A subset-simulation-based HM procedure serves as a precursor during the initialization step, efficiently identifying high-probability regions of the posterior distribution. Subsequently, SMC samplers generate posterior samples by iteratively refining and adjusting the existing sample set to better approximate the target posterior distribution.

When new data becomes available, SBHM uses a stopping rule based on the effective sample size to determine whether to continue with the current SMC sampler. If the previously obtained information becomes nearly irrelevant due to new data, SBHM reverts to HM. Otherwise, SBHM directly updates the available samples. In contrast, existing calibration algorithms must restart entirely when new data arrives, regardless of the correlation between the new data and previous observations.

We presented two numerical case studies illustrating that the proposed algorithm can generate posterior samples and efficiently update the posterior distribution when new data is relevant. By appropriately tuning the hyperparameter α, SBHM can provide accurate approximations comparable to or better than a full restart, but with significantly reduced computational cost.

The numerical case studies presented in this paper resulted in uni-modal posteriors. Preliminary experiments with models having multi-modal posteriors that are not presented in the current paper, indicate that SBHM is capable of detecting multi-modality at the cost of stability. The study of multi-modal posteriors is certainly important and we defer this to future work. Another setting worth exploring is that which involves computationally expensive simulators with dimensions that vary over time. Investigations into efficient methods for training emulators as new observations become available are another important area of future research. In line with the ideas of transfer learning, this could include linking the newly trained emulator with previously trained ones to leverage existing knowledge and minimize computational cost.

CRediT authorship contribution statement

J. Cheng: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Formal analysis. **F.A. DiazDelaO:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **P.O. Hristov:** Writing – review & editing, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was funded by UKRI through the CHIMERA (Collaborative Healthcare Innovation through Mathematics, EngineeRing and AI) grant EP/T017791/1. This paper is dedicated to Prof. John Mottershead, who has been a model of excellence for many generations of researchers.

Data availability

Data will be made available on request.

References

[1] R.G. Bower, M. Goldstein, I. Vernon, Galaxy formation: a Bayesian uncertainty analysis, Bayesian Anal. 5 (2010) 619–669, http://dx.doi.org/10.1214/10-ba524.
[2] J.E. Mottershead, M. Friswell, Model updating in structural dynamics: A survey, J. Sound Vib. 167 (1993) 347–375, http://dx.doi.org/10.1006/jsvi.1993.1340.
[3] M.I. Friswell, J.E. Mottershead, H. Ahmadian, Finite–element model updating using experimental test data: Parametrization and regularization, Philos. Trans. R. Soc. Lond. A 359 (2001) 169–186, http://dx.doi.org/10.1098/rsta.2000.0719.
[4] J.E. Mottershead, M. Link, M.I. Friswell, The sensitivity method in finite element model updating: A tutorial, Mech. Syst. Sig. Process. 25 (2011) 2275–2296, http://dx.doi.org/10.1016/j.ymssp.2010.10.012.
[5] S. Bi, M. Beer, S. Cogan, et al., Stochastic model updating with uncertainty quantification: An overview and tutorial, Mech. Syst. Sig. Process. 204 (2023) 110784, http://dx.doi.org/10.1016/j.ymssp.2023.110784.
[6] C. Mares, J.E. Mottershead, M.I. Friswell, Stochastic model updating: Part 1—theory and simulated example, Mech. Syst. Sig. Process. 20 (2006) 1674–1695, http://dx.doi.org/10.1016/j.ymssp.2005.06.006.
[7] Q.B. Cheng, X. Chen, C.Y. Xu, et al., Improvement and comparison of likelihood functions for model calibration and parameter uncertainty analysis within a Markov chain Monte Carlo scheme, J. Hydrol. 519 (2014) 2202–2214, http://dx.doi.org/10.1016/j.jhydrol.2014.10.008.
[8] G.J. Gibson, E. Renshaw, Estimating parameters in stochastic compartmental models using Markov chain methods, Math. Med. Biol. 15 (1998) 19–40, http://dx.doi.org/10.1093/imammb/15.1.19.
[9] I. Andrianakis, I.R. Vernon, N. McCreesh, et al., Bayesian history matching of complex infectious disease models using emulation: A tutorial and a case study on HIV in uganda, PLoS Comput. Biol. 11 (2015) e1003968, http://dx.doi.org/10.1371/journal.pcbi.1003968.
[10] P. Gardner, C. Lord, R.J. Barthorpe, Bayesian history matching for structural dynamics applications, Mech. Syst. Sig. Process. 143 (2020) 106828, http://dx.doi.org/10.1016/j.ymssp.2020.106828.
[11] P.S. Craig, M. Goldstein, A.H. Seheult, et al., Pressure matching for hydrocarbon reservoirs: A case study in the use of Bayes linear strategies for large computer experiments, in: C. Gatsonis, J.S. Hodges, R.E. Kass, et al. (Eds.), Case Studies in Bayesian Statistics: Volume III, Springer, New York, 1997, pp. 37–93.
[12] M.C. Kennedy, A. O'Hagan, Bayesian calibration of computer models, J. R. Stat. Soc. B 63 (2001) 425–464, http://dx.doi.org/10.1111/1467-9868.00294.
[13] R.M. Gladstone, V. Lee, J. Rougier, et al., Calibrated prediction of pine island glacier retreat during the 21st and 22nd centuries with a coupled flowline model, Earth Planet. Sci. Lett. 333 (2012) 191–199, http://dx.doi.org/10.1016/j.epsl.2012.04.022.
[14] P. Gardner, T. Rogers, C. Lord, et al., Learning model discrepancy: A Gaussian process and sampling-based approach, Mech. Syst. Sig. Process. 152 (2021) 107381, http://dx.doi.org/10.1016/j.ymssp.2020.107381.

[15] F. Anterion, History matching: A one day long competition: Classical approaches versus gradient method, in: First International Forum on Reservoir Simulation, Alpbach, Austria, 1988.
[16] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (2000) 55–61, http://dx.doi.org/10.2307/1268522.
[17] J.E. Oakley, A. O'Hagan, Probabilistic sensitivity analysis of complex models: A Bayesian approach, J. R. Stat. Soc. B 66 (2004) 751–769, http://dx.doi.org/10.1111/j.1467-9868.2004.05304.x.
[18] F. Pukelsheim, The three sigma rule, Am. Stat. 48 (1994) 88–91, http://dx.doi.org/10.2307/2684253.
[19] A. Doucet, N. De Freitas, N. Gordon, An introduction to sequential Monte Carlo methods, in: A. Doucet, N. De Freitas, N. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer, New York, 2001, pp. 3–14.
[20] R.E. Kalman, A new approach to linear filtering and prediction problems, J. Basic Eng. 82 (1960) 35–45, http://dx.doi.org/10.1115/1.3662552.
[21] S. Hong, M. Bolic, P.M. Djuric, An efficient fixed-point implementation of residual resampling scheme for high-speed particle filters, IEEE Signal Process. Lett. 11 (2004) 482–485, http://dx.doi.org/10.1109/LSP.2004.826634.
[22] B. Efron, R.J. Tibshirani, An Introduction to the Bootstrap, CRC Press, New York, 1994.
[23] Z.Y. Wang, Z.T. Liu, W.Q. Liu, et al., Particle filter algorithm based on adaptive resampling strategy, in: Proceedings of 2011 International Conference on Electronic and Mechanical Engineering and Information Technology, Vol. 6, IEEE, Harbin, China, 2011, pp. 3138–3141.
[24] J. Zuo, Dynamic resampling for alleviating sample impoverishment of particle filter, IET Radar Sonar Navig. 7 (2013) 968–977, http://dx.doi.org/10.1049/iet-rsn.2013.0009.
[25] P. Del Moral, Feynman–Kac Formulae, Springer, New York, 2004.
[26] P. Del Moral, A. Doucet, A. Jasra, Sequential Monte Carlo samplers, J. R. Stat. Soc. B 68 (2006) 411–436, http://dx.doi.org/10.1111/j.1467-9868.2006.00553.x.
[27] Z. Gong, F. DiazDelaO, P.O. Hristov, et al., History matching with subset simulation, Int. J. Uncertain. Quantif. 11 (2021) http://dx.doi.org/10.1615/Int.J.UncertaintyQuantification.2021033543.
[28] S.-K. Au, J.L. Beck, Estimation of small failure probabilities in high dimensions by subset simulation, Probab. Eng. Mech. 16 (2001) 263–277, http://dx.doi.org/10.1016/S0266-8920(01)00019-4.
[29] J. Bect, L. Li, E. Vazquez, Bayesian subset simulation, SIAM/ASA J. Uncertain. Quantif. 5 (2017) 762–786, http://dx.doi.org/10.1137/16M1078276.
[30] S. Goshtasbpour, V. Cohen, F. Perez-Cruz, Adaptive annealed importance sampling with constant rate progress, in: International Conference on Machine Learning, PMLR, Honolulu, 2023, pp. 11642–11658.
[31] N. Kantas, A. Beskos, A. Jasra, Sequential Monte Carlo methods for high-dimensional inverse problems: A case study for the navier–stokes equations, SIAM/ASA J. Uncertain. Quantif. 2 (2014) 464–489, http://dx.doi.org/10.1137/130930364.
[32] Y. Zhou, A.M. Johansen, J.A. Aston, Toward automatic model comparison: An adaptive sequential Monte Carlo approach, J. Comput. Graph. Stat. 25 (2016) 701–726, http://dx.doi.org/10.1080/10618600.2015.1060885.
[33] H. Tjelmeland, B.K. Hegstad, Mode jumping proposals in MCMC, Scand. J. Stat. 28 (2001) 205–223, http://dx.doi.org/10.1111/1467-9469.00232.
[34] M. Cabeleira, D. Anand, S. Ray, et al., Comparing physiological impacts of positive pressure ventilation versus self-breathing via a versatile cardiopulmonary model incorporating a novel alveoli opening mechanism, Comput. Biol. Med. 180 (2024) 108960, http://dx.doi.org/10.1016/j.compbiomed.2024.108960.
[35] C. Ngo, S. Dahlmanns, T. Vollmer, et al., An object-oriented computational model to study cardiopulmonary hemodynamic interactions in humans, Comput. Methods Programs Biomed. 159 (2018) 167–183, http://dx.doi.org/10.1016/j.cmpb.2018.03.008.
[36] A. Albanese, L. Cheng, M. Ursino, et al., An integrated mathematical model of the human cardiopulmonary system: Model development, Am. J. Physiol.-Hear. Circ. Physiol. 310 (2016) H899–H921, http://dx.doi.org/10.1152/ajpheart.00230.2014.
[37] C. Liu, S. Niranjan, J. Clark Jr., et al., Airway mechanics, gas exchange, and blood flow in a nonlinear model of the normal human lung, J. Appl. Physiol. 84 (4) (1998) 1447–1469, http://dx.doi.org/10.1152/jappl.1998.84.4.1447.
[38] S. Coveney, C. Corrado, J.E. Oakley, et al., Bayesian calibration of electrophysiology models using restitution curve emulators, Front. Physiol. 12 (2021) 1120, http://dx.doi.org/10.3389/fphys.2021.693015.