Drovandi, Christopher C., Nott, David J., & Pagendam, Daniel E.
(2017)
New insights into history matching via sequential Monte Carlo.
[Working Paper]
(Unpublished)

New Insights into History Matching via Sequential Monte Carlo

Christopher C. Drovandi*, David J. Nott† and Daniel E. Pagendam‡

*School of Mathematical Sciences,
Australian Centre of Excellence for Mathematical and Statistical Frontiers,
Queensland University of Technology, Brisbane, Australia 4001
email: c.drovandi@qut.edu.au

† Department of Statistics and Applied Probability,
National University of Singapore

‡ CSIRO Data61, Dutton Park, Australia, 4102

October 4, 2017

Abstract

The aim of the history matching method is to locate non-implausible regions of the parameter space of complex deterministic or stochastic models by matching model outputs with data. It does this via a series of waves where at each wave an emulator is fitted to a small number of training samples. An implausibility measure is defined which takes into account the closeness of simulated and observed outputs as well as emulator uncertainty. As the waves progress, the emulator becomes more accurate so that training samples are more concentrated on promising regions of the space and poorer parts of the space are rejected with more confidence. Whilst history matching has proved to be useful, existing implementations are not fully automated and some ad-hoc choices are made during the process, which involves user intervention and is time consuming. This occurs especially when the non-implausible region becomes small and it is difficult to sample this space uniformly to generate new training points. In this article we develop a sequential Monte Carlo (SMC) algorithm for implementation which is semi-automated. Our novel SMC approach reveals that the history matching method yields a non-implausible distribution that can be multi-modal, highly irregular and very difficult to sample uniformly. Our SMC approach offers a much more reliable sampling of the non-implausible space, which requires additional computation compared to other approaches used in the literature.

Keywords: emulator, Gaussian process, history matching, hydrology, Markov chain Monte Carlo, Markov processes.

1 Introduction

As practitioners strive to develop more realistic models, the computational burden of calibrating them to observed data typically increases. The development and implementation of such models often results in some computer code that is not tractable to handle in a conventional way. Complex models arise in many different fields such as biology (e.g. Vo et al. (2015)), climate (e.g. Holden et al. (2015)), cosmology (e.g. Vernon et al. (2014)) and many other disciplines. In the case of deterministic models, it may be very expensive to solve the model even for a single parameter configuration. Stochastic models may be expensive to simulate, and may only produce noisy likelihood estimates. History matching (e.g. Craig et al. (1997)) is a method for determining non-implausible regions of the parameter space of a complex computer model where simulation from the model is very expensive. A non-implausible parameter value is defined to be one for which there currently is not evidence that simulation outputs generated using that value will not match observed outputs. The method proceeds in a series of waves and uses emulation to determine thriftily without model simulation the current non-implausible region. From this region a training sample can be formed, model simulation performed, followed by another application of emulation. After each wave the non-implausible volume gets smaller as the emulator is more accurate over smaller volumes and more training points are generally placed closer to where simulated outputs match observed outputs.

Whilst history matching has proven to be useful in quickly eliminating large portions of the implausible parameter space, user intervention is often required during the process and various ad-hoc decisions are made. In this paper we develop a novel sequential Monte Carlo (SMC) algorithm for history matching. By doing this, we obtain a more generic, automated and reliable algorithm for exploring the non-implausible space. As a by-product of the latter, we demonstrate the complexity of the distribution implied by the non-implausible region generated from history matching and that the SMC method offers a principled approach for exploring this space.

The paper is structured as follows. Details of the history matching approach and some variants of how it is implemented are provided in Section 2. Section 3 describes the novel SMC framework for history matching. Numerical examples to illustrate the ideas in this paper are shown in Section 4. The paper concludes with a discussion in Section 5.

2 History Matching

In this section we describe the history matching method and some of the issues with current implementations of it.

2.1 The Method

Let $\theta \in \Theta \subseteq \mathbb{R}^p$ be the parameter of interest of a model $\mathcal{M}$ where p is the number of parameters. We denote the 'prior' distribution of the parameter as $\pi(\theta)$. In a Bayesian sense, the prior could be formed from expert opinion and/or historical data or the prior could simply be some hyper-rectangle that defines the search space. The model $\mathcal{M}$ may be deterministic or stochastic. We denote the solution of the deterministic system as $y_\theta = \mathcal{M}(\theta)$ and generation from the stochastic system as $y_\theta = \mathcal{M}(\theta, u)$ where u are the random numbers required in the stochastic model and $y_\theta \in \mathcal{Y} \subseteq \mathbb{R}^n$ where n is the number of outputs. We use the term 'model simulation' for both deterministic and stochastic models. The observed output is denoted as $y_{\text{obs}} \in \mathcal{Y}$.

For simplicity we assume there is a single output $n = 1$. The single output could be some scalar y_{obs}. Alternatively, when y_{obs} is vector-valued, the output could be some distance $\rho_\theta = ||y_\theta - y_{\text{obs}}||$ between simulated and observed outputs, or the 'distance' could be quantified through a chosen likelihood function, $f(y_{\text{obs}}|\theta)$ or a 1-1 transformation of it such as the log-likelihood. If the exact likelihood computation is not feasible, an approximate likelihood could be used, such as a stochastic estimator of the likelihood. In Section 5 we discuss how our SMC algorithm for history matching could be extended to multiple outputs.

```
  Input  : A prior distribution $\pi(\theta)$ where $\theta \in \Theta$.
  Output: A region $\Theta_s \subset \Theta$ that is deemed as non-implausible.
1 Set the wave counter to $w = 1$
2 Generate $N_w$ training samples $\{\theta_j\}_{j=1}^{N_w} \sim \pi(\theta)$ using a space filling design and
    simulate the model at each $\theta_j$ to generate the collection of outputs $\{y_{\theta_j}\}_{j=1}^{N_w}$.
3 Fit an emulator $E_w$ to the training data $\{\theta_j, y_{\theta_j}\}_{j=1}^{N_w}$.
4 Use the emulator $E_w$ to define an implausibility function $\mathcal{I}_w(\theta)$. If $\mathcal{I}_w(\theta) > c_w$ for
    some chosen $c_w$ then $\theta$ is deemed as implausible by emulator $E_w$.
5 Use all emulators $\{E_r\}_{r=1}^{w}$ to define the non-implausible region
    $\Theta_w = \{\theta \in \Theta| \cap_{r=1}^{w} \mathcal{I}_w(\theta) < c_w\}$.
6 Increase wave counter $w = w + 1$.
7 Generate $N_w$ training samples $\{\theta_j\}_{j=1}^{N_w}$ from $\Theta_w$ and simulate the model at each $\theta_j$ to
    generate the collection of outputs $\{y_{\theta_j}\}_{j=1}^{N_w}$.
8 If the stopping rule is satisfied then finish otherwise return to Line 3.
```

Algorithm 1: Steps involved in the history matching algorithm.

The steps involved in history matching are shown in Algorithm 1. The first step involves generating a training sample from the prior $\pi(\theta)$ and simulating from the model at these points. It is common in the history matching literature to use an approach with improved space filling properties compared to pseudo-random samples. For example, latin hypercube sampling (Iman, 2008) or quasi-Monte Carlo (QMC, Niederreiter (1992)) are popular choices.

An emulator is then fitted to the training sample. A common choice is the Gaussian process (GP), but in principle any emulator can be used. In the numerical results in Section 4 we use a GP and assume that the reader is familiar with them (see Rasmussen and Williams (2006) for details on GPs). The fitted emulator can then divide the parameter space into implausible and non-implausible regions. The practitioner must decide upon an implausibility measure, $\mathcal{I}(\theta)$, to inform the split. In particular, an untested θ is deemed as currently non-implausible if $\mathcal{I}(\theta) < c$ for some chosen cut-off c. One common choice (e.g. Andrianakis et al. (2015)) is as follows

$$\mathcal{I}(\theta) = \frac{|y_\theta - y_{\mathrm{obs}}|}{\sqrt{s_{m,\theta}^2 + s_{e,\theta}^2 + s_d^2}}, \tag{1}$$

where $s_{m,\theta}$ is the estimated standard deviation of the model output, $s_{e,\theta}$ is a standard deviation that arises from emulation uncertainty and s_d is an additional standard deviation to account for the fact that the model might be misspecified in some way. The model output standard deviation $s_{m,\theta}$ is zero if the model is deterministic. In the case of a stochastic model, the model output standard deviation can be estimated easily from the training data if it is assumed that it is independent of θ. If this is not a reasonable assumption, then K independent simulations can be performed for each θ in the training sample, and a surface can be fitted to the empirical standard deviations to predict for untested θ. When K simulations are performed, the output y_θ is an average of the K simulations. Based on the 3σ rule (Pukelsheim, 1994), $c = 3$ may be a reasonable cut-off.

The implausibility measure could also be formed from the log-likelihood function of the underlying model. For deterministic models, for example, the observed values may be subject to a Gaussian measurement error. In latent variable stochastic models, an unbiased estimate of the likelihood could be obtained via importance sampling methods (Andrieu and Roberts, 2009), which may be expensive to obtain. In such instances we could consider the implausibility function as $\mathcal{I}(\theta) = -\log f(y_{\mathrm{obs}}|\theta) - r \times s_{e,\theta}^2$ and recognise that the log-likelihood, $\log f(y_{\mathrm{obs}}|\theta)$, may not be the exact value but rather only an estimate of it. It is natural to consider the log-likelihood as opposed to the likelihood as it is probable that the log-likelihood can be more accurately emulated. Wilkinson (2014) consider even taking the log a second time if the log-likelihood values differ widely for different θ in the training sample. The value of r may be considered as an exploration parameter. The larger it is the more we are inclined to explore where the emulator is uncertain. The smaller it is the more we exploit regions where the emulator predicts the log-likelihood to be large. Finally, if we simply take the distance between observed and simulated data we might consider $\mathcal{I}(\theta) = \rho_\theta - r \times s_{e,\theta}^2$. Based on the Gaussian assumption of the GP prediction we may take $r = 3$.

The next step involves generating uniformly from the non-implausible parameter space, which we denote as $\Theta_1 = \{\theta \in \Theta | \mathcal{I}(\theta) < c\}$. This produces a new training sample that is concentrated in more promising parts of the parameter space relative to the initial training sample from the prior. This essentially re-starts the history matching process, where an emulator is fitted to the new training sample. The new emulator is likely to lead to more accurate predictions as the training sample is less diffuse and the outputs simulated from the model less variable across Θ_1. The history matching method continues in this fashion in waves, where we denote the wave counter as w. At wave w the implausibility function is denoted $\mathcal{I}_w(\theta)$ with a cut-off c_w potentially depending on the wave. After w waves, the non-implausible parameter space is denoted as $\Theta_w = \{\theta \in \Theta | \cap_{r=1}^{w} \mathcal{I}_w(\theta) < c_w\}$. That is, for an untested θ to be deemed as non-implausible for wave $w + 1$ it must be deemed as non-implausible for all previous w waves.

The waves continue until a user-specified stopping rule is met. Andrianakis et al. (2017) discuss some commonly used stopping rules; the entire space may be deemed implausible, the emulator's uncertainty is small enough and/or a sufficient number of points that match the observed data have been collated. Given that the stopping rule selected is likely to be problem and practitioner dependent, we do not attempt to address the issue in this paper.

2.2 Issues with History Matching

Despite the ability of history matching to relatively quickly identify parts of the parameter space that may be consistent with the observed data, it does have at least two issues:

1. The cut-off values c_w may not be easy to select in practice and there is no existing automated method for doing so.

2. Sampling uniformly from Θ_w as w increases becomes increasingly difficult.

Regarding issue 1, as was already mentioned, a sensible cut-off value for the implausibility measure in (1) is 3. However, this value assumes that we can accurately compute the quantity in (1). This would involve being able to model $s_{m,\theta}$ and s_d accurately, which may be difficult to do, especially for the model misspecification term s_d (especially when this term may depend on θ). Sometimes the s_d term is simply ignored. If the implausibility measure is not calculated accurately enough then a cut-off value of $c = 3$ may lead to the non-implausible parameter space reducing too quickly or not quickly enough. Many authors (e.g. Vernon et al. (2014), Wilkinson (2014) and Andrianakis et al. (2015)) report instances of having to manually change these cut-off values after each wave and/or resort to differing numbers of training samples for each wave. Regarding the log-likelihood implausibility measure, Wilkinson (2014) suggest a cut-off value that depends on the maximum log-likelihood value/estimate in the training sample minus 10 log-likelihood points, but this value may not be appropriate for all applications and may also need to change over the waves. If the implausibility measure is based on some distance between observed and simulated data, then an appropriate cut-off value is generally unclear and will need to change throughout the algorithm as distances get generally smaller. Thus an approach able to select the thresholds in a more automated way would be welcome.

Regarding issue 2, uniform samples from Θ_w can be obtained perfectly by rejection sampling; continually sampling from $\pi(\theta)$ until $\theta \in \Theta_w$ (see, for example, Wilkinson (2014)). However, as w increases, the acceptance rate of this rejection sampler decreases rapidly. It may become unacceptably small if the volume of the parameter space consistent with the observed data is a tiny fraction of the volume of the original parameter space. Once the rejection sampler becomes too inefficient various authors resort to ad-hoc approaches for generating training samples from subsequent waves, choices that do not preserve the uniform distribution on Θ_w. For example, Andrianakis et al. (2015) proposes 20 parameter values centered on a subset of parameter values from a previous wave that are deemed non-implausible. An approach that can reliably sample the non-implausible space uniformly is of interest to ensure that the parameter space is comprehensively explored.

Williamson and Vernon (2013) consider a more sophisticated approach to the sampling problem based on evolutionary Monte Carlo. The approach is similar to parallel tempering in that several chains are run in parallel. One of the chains has the desired target, the uniform distribution over Θ_w, as its limiting distribution, whilst the other chains increasingly relax the constraints so that they can more freely search the parameter space. The method has local moves to update each chain and also proposes swaps between chains. Although the approach has the desired target as its limiting distribution, its implementation is not straightforward and there are many tuning parameters such as the number of chains, the target distribution of each chain, local proposal distributions and chain swapping mechanisms.

Andrianakis et al. (2017) provide another serious attempt to uniformly sample the non-implausible region. The method is based on slice sampling. However, since each co-ordinate of the input is updated separately, this approach may be inefficient if there is dependence between inputs implied by the data. Nonetheless, the approach of Andrianakis et al. (2017), or some adaptation of it, could be used in our SMC framework described next.

3 SMC Framework for History Matching

Here we propose to place the history matching method into the SMC framework. This allows us to take advantage of the extensive research on efficient SMC algorithms (e.g. Del Moral et al. (2006), Fearnhead and Taylor (2013) and South et al. (2016)), which are naturally adaptive.

SMC involves sampling from a sequence of distributions that smoothly evolves between a distribution that is easy to sample from and finishing at the distribution of interest, sometimes referred to as the target distribution. For history matching, we define the sequence of distributions as

$$p_w(\theta) \propto \pi(\theta) \prod_{k=1}^{w} \mathbb{I}(\mathcal{I}_k(\theta) \leq c_k), \tag{2}$$

where $\pi(\theta)$ defines the distribution that samples are initially drawn from, for example a hyper-rectangle or some other distribution that has been informed from experts or historical data. Our approach fits within the framework of Chopin (2002), but see Del Moral et al. (2006) for a more general framework for SMC methods. Assume that we have a collection of weighted samples or 'particles', $\{W_w^i, \theta_w^i\}_{i=1}^M$ from $p_w(\theta)$. To push the particle set to the next target, a re-weighting step is required. A simple importance sampling argument leads to the following update of the weights

$$W_{w+1}^i \propto W_w^i \mathbb{I}(\mathcal{I}_{w+1}(\theta_w^i) \leq c_{w+1}),$$

so that the values of the samples θ_w^i themselves remain unchanged. Assuming that $W_w^i = 1/M$ for $i = 1, \ldots, M$, the weights W_{w+1} will either be proportional to a constant or equal to zero. Thus placing the history matching approach into the SMC framework allows us to select the implausibility cut-off at each wave c_{w+1} so that a certain proportion, α, have a non-zero weight. This is equivalent to ensuring that the effective sample size (ESS), often measured by $1/\sum_{i=1}^M (W_{w+1}^i)^2$, is roughly αM. A similar approach is adopted by Drovandi and Pettitt (2011a) and Del Moral et al. (2012) in the SMC algorithms developed for approximate Bayesian computation. Drovandi et al. (2016) develop a similar idea for calibrating differential equations to population data in the presence of parameter uncertainty.

After the re-weighting step the ESS drops to roughly αM. Resampling M times from the surviving particles allows the ESS to return to M. However, the drawback is that some particles will be duplicated. A diverse sample from each of the targets is desired. This can be achieved by applying an MCMC kernel to each of the resampled particles. Given that an MCMC kernel may reject proposals, it is generally advised to apply say R iterations of the MCMC kernel. We implement the adaptive strategy promoted by South et al. (2016), which involves running one MCMC iteration on each particle, estimating the average MCMC acceptance rate, and using that to determine R.

The selection of an efficient MCMC kernel is critical to the success of the SMC algorithm. SMC has the distinct advantage over serial-based algorithms such as standard MCMC in that the population of particles can be harnessed to help build a useful MCMC proposal. The simplest approach is a multivariate normal random walk (perhaps after transforming each parameter to the real line) with a covariance matrix calibrated from the weighted or re-sampled particle set (e.g. Drovandi and Pettitt (2011a)). However, given the complexity of the distributions that arise from history matching (illustrated in Section 4), we find that this standard approach is infeasible due to the low MCMC acceptance rates generated and hence unattractively large values of R.

The slice sampling approach of Andrianakis et al. (2017) could be used in the MCMC step of our method. However, we find some success for the examples in this paper with the following approach. A kernel density estimate (kde) using the Epanechnikov kernel is calibrated to each component of θ (i.e. the marginals) based on a subset of the M re-sampled particles. If the parameter space is bounded then the kde is also restricted to that support. The cumulative distribution function of the fitted kdes can be used to transform each of the marginals to roughly standard uniform, which can be transformed again to roughly standard normal via the standard normal quantile function. A multivariate normal random walk is then applied to this transformed space. We find that the distribution on the transformed space is significantly more regular compared to the transformed space, in the sense that heavy tails can be reduced and modes brought closer together. Once a proposal is generated on the transformed space, a sample on the original space requires evaluating the quantile function for each of the fitted marginals. Unfortunately the quantile function of the kde has no explicit form, but it can be approximated numerically. All of these options and functionality are provided in the `ksdensity` function in Matlab.

We note that the requirement to approximate the quantile function of the kde for each of the marginals substantially slows down the MCMC step relative to other standard transforms such as the log (positive parameters) or the logistic (bounded parameters). However, we find that in the challenging examples in Section 4 the approach we adopt is critical to maintain a reasonable value of R throughout the algorithm. More detail and comparison is provided in Section 4. In typical applications of history matching where the simulation of the model is very expensive, the MCMC step of the SMC algorithm, which does not involve any model simulation, may remain relatively fast. However, one of the messages of this paper is that a significant and perhaps unavoidable amount of effort needs to be spent on ensuring the non-implausible part of the space is well represented at each wave.

Because the sequence of distributions (2) implied by SMC history matching involves indicator functions, it is possible to reject MCMC proposals early based on the prior and proposal densities before checking for implausibility from the fitted emulators. This idea has also been used in the context of ABC (Picchini, 2014). Furthermore, once a proposal has been deemed implausible by one of the emulators, there is no need to check for implausibility with the remaining emulators.

After each wave, the SMC process generates M particles from $p_w(\theta)$. The training sample for the next wave can be obtained by sub-sampling $N << M$ particles from this set (with duplicates ignored for deterministic models). The SMC history matching algorithm developed in this paper is summarised in Algorithm 2.

The larger the value of M the more chance there is of the non-implausible space being well represented. On the other hand, the computing time of the MCMC step will increase linearly with M.

```
Input  : The number of SMC particles $M$, the 'prior' distribution $\pi(\theta)$, the desired
         probability $1-c$ of moving a particle in the MCMC step, the proportion $\alpha$
         of SMC particles to keep at each iteration that also helps to define the
         implausibility cut-off and the number of training samples $N$ for fitting the
         emulator at each wave.
Output: A collection of parameter values $\{\theta^i\}_{i=1}^M$ that are deemed as
         non-implausible.
Set wave counter $w = 0$
Set $W_0^i = \frac{1}{M}$ for $i = 1, \ldots, M$
Simulate $\theta_0^i \overset{\text{iid}}{\sim} \pi(\theta)$ for $i = 1, \ldots, M$
Simulate $N$ training samples from $\pi(\theta)$ using a space filling design and fit an
  emulator $E_w$
while stopping rule not met do
    Set $w = w + 1$
    Compute implausibility measure $\mathcal{I}_w(\theta_{w-1}^i)$ for $i = 1, \ldots, M$
    Determine implausibility cut-off value $c_w$ based on the $\alpha$ quantile of implausibility
      values $\{\mathcal{I}_w(\theta_{w-1}^i)\}_{i=1}^M$
    Resample floor($(1-\alpha)M$) particle values from the surviving particles (i.e. those
      with $\mathcal{I}_w(\theta_{w-1}^i) \leq c_w$) to replenish the SMC population
    Set $\theta_w^i = \theta_{w-1}^i$ for $i = 1, \ldots, M$
    Form a suitable MCMC proposal distribution from the resampled particles
    Move $\{\theta_w^i\}_{i=1}^M$ with one iteration of an MCMC kernel targetting $p_w(\theta)$ (see
      Algorithm 3)
    Compute $R_t = \left\lceil \frac{\log(c)}{\log(1-p_{\text{acc}})} \right\rceil$ where $p_{\text{acc}}$ is the acceptance probability of the above
      move step
    for $k = 1$ to $R_t$ do
        Move $\{\theta_w^i\}_{i=1}^M$ with one iteration of an MCMC kernel targetting $p_w(\theta)$ (see
          Algorithm 3)
    end
    Sample $N$ particles from $\{\theta_w^i\}_{i=1}^M$ without replacement (removing duplicates for
      deterministic models). Fit emulator $E_w$ based on this training sample
end
```

Algorithm 2: SMC History Matching Algorithm.

```
    Input  : Collection of re-sampled particles ${\theta^i}_{i=1}^M$ from $p_w(\theta)$, wave counter $w$,
             fitted emulators $\{E_1, E_2, \ldots, E_w\}$, the implausibility threshold cut-offs
             $c_1, c_2, \ldots, c_w$ and the 'prior' distribution $\pi(\theta)$. For simplicity of notation we
             omit the wave index $w$ from $\theta$.
    Output: Collection of particles $\{\theta^i\}_{i=1}^M$ from $p_w(\theta)$ that are better diversified.
 1  Fit a kde to each marginal of $\theta$ based on the collection of particles $\{\theta^i\}_{i=1}^M$
 2  Use the cdf of the fitted kde and the normal quantile function to transform each
      margin to roughly standard normal. Denote the transformed particles as $\{z_w^i\}_{i=1}^M$
 3  Estimate covariance matrix from $\{z^i\}_{i=1}^M$, $\hat{\Sigma}$
 4  for $i = 1, \ldots, M$ do
 5      Proprose $z^* \sim \mathcal{N}(z^i, \hat{\Sigma})$
 6      Use the fitted kdes to transform each margin of $z^*$ back to the original scale
          denoted $\theta^*$
 7      Compute the proposal density
          $q(\theta^*|\theta^i) = \exp\left(\sum_{k=1}^p \log \hat{f}_k(\theta^*[k]) - \sum_{k=1}^p \log \mathcal{N}(z^*[k]; 0, 1)\right)$ where $\theta[k]$ denotes
          the $k$th component of $\theta$ and $\hat{f}_k(\cdot)$ denotes the fitted kde for the $k$th marginal
 8      Compute the proposal density in the other direction $q(\theta^i|\theta^*)$
 9      Compute the first part of the Metropolis-Hastings ratio
          $r = \min(\pi(\theta^*)q(\theta^i|\theta^*)/\pi(\theta^i)q(\theta^*|\theta^i))$
10      if $\mathcal{U}(0, 1) > r$ then
11          reject early and go to the next iteration of the for loop at line 4
12      end
13      for $j = 1, \ldots, w$ do
14          Compute $\mathcal{I}_j(\theta^*)$ from fitted emulator $E_j$
15          if $\mathcal{I}_j(\theta^*) > c_j$ then
16              reject early and go to the next iteration of the for loop at line 4
17          end
18      end
19      Accept $\theta^i = \theta^*$
20  end
```

Algorithm 3: MCMC kernel used within the generic SMC history matching method in Algorithm 2. The proposal distribution for the MCMC algorithm is specific to this paper. For simplicity of notation we omit the wave index w from θ.

One advantage of the SMC sampling approach over the evolutionary Monte Carlo method of Williamson and Vernon (2013) is that the information from the SMC population of particles at the current wave can be harnessed to help facilitate uniform sampling of the non-implausible region at the next wave, whilst the method of Williamson and Vernon (2013) needs to be re-started at each wave. Further, the SMC approach allows for adaptive choice of the cut-offs. Finally, the approach of Williamson and Vernon (2013) is an MCMC algorithm and is thus less suited to parallel computing compared to our SMC approach. Parallel computing architectures are critical in history matching applications so that model simulations can be performed in parallel for different inputs. Therefore it is useful that our SMC algorithm for uniform sampling of the non-implausible space can take advantage of parallel computing.

4 Examples

Below we illustrate our ideas on a toy problem, and non-trivial applications in hydrology and biology. Both of the substantive applications do not necessarily require history matching as the model simulation is not expensive relative to typical applications of history matching. However, they are sufficient for us to illustrate the messages of this paper. The hydrology example involves a deterministic model and the implausibility measure is based on a distance between observed and simulated data. The SMC history matching approach is compared with an SMC optimisation routine. The biological example involves a stochastic model for an autoregulatory gene network, and the implausibility measure is based on an unbiased likelihood estimator. In this case we compare the output and efficiency with a more standard Bayesian SMC method.

4.1 Toy Problem

Here the objective is to minimise the following function taken from one of the test functions in Molga and Smutnicki (2005):

$$y = -\sin(x_1)\sin(x_1^2/\pi)^2 - \sin(x_2)\sin(2x_2^2/\pi)^2,$$

where $\theta = (x_1, x_2) \in (0, \pi) \times (0, \pi)$. Figure 1 provides a visualisation of the function.

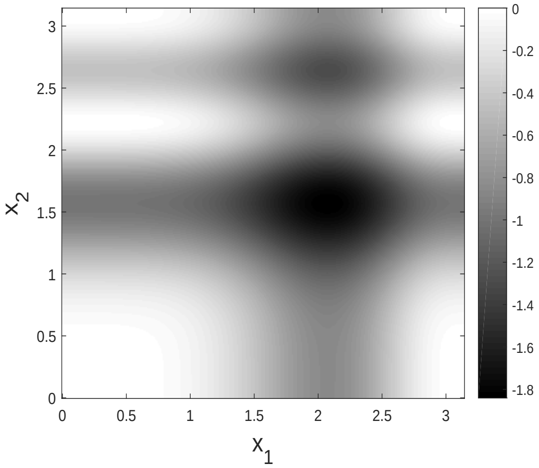

Figure 1: Visualisation of the toy function. Black regions show where the function value is smallest.

For each wave we randomly select $N = 50$ training samples for fitting the GP. The history matching method is solved using brute force by starting with 2^{20} QMC samples over the bounded support. The process starts by selecting $N = 50$ samples at random from this initial set and fitting a GP. The implausibility for each sample is computed as $\mathcal{I}(\theta) = y_p(\theta) - r \times s_p(\theta)$ where $y_p(\theta)$ and $s_p(\theta)$ is the mean prediction of the function and the standard deviation from the currently fitted GP, respectively. The exploration parameter is set at $r = 3$. The cut-off for implausibility at wave w is given by $\mathcal{I}_w(\theta) > c_w$ where c_w is selected adaptively such that exactly half of the surviving samples satisfy the cut-off. Thus after each wave half of the original 2^{20} samples is lost. Nine waves are used so that there are still a significant number of the original samples that satisfy all of the waves. In effect we have perfectly uniform samples from each of the non-implausible regions defined after each wave. The non-implausible regions are shown in black in Figure 2. The training samples are shown as crosses, and are effectively taken uniformly at random from the black region in the previous wave (the black region for wave 0 is the entire space of θ).

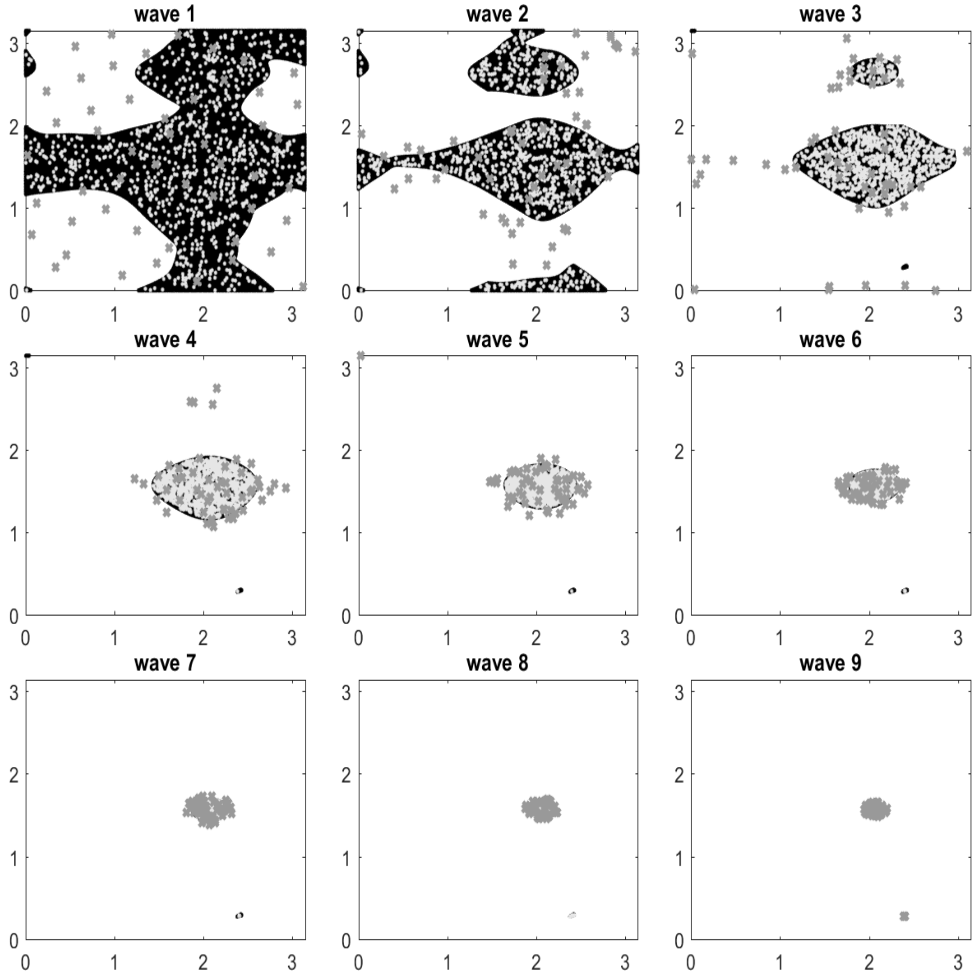

Figure 2: Results for the SMC approach to sample from the sequence of distributions obtained by brute force for the toy example. The black regions denote the non-implausible regions after each wave and the light grey dots are the SMC samples. The grey crosses are the training samples for fitting the GP during the brute force process.

What is immediately noticeable even from this very simple example is that the non-implausible region defined from the history matching procedure can be significantly disconnected, which creates a probability distribution that can be highly multi-modal and irregular, making it difficult to sample efficiently and reliably. This is because the non-implausible regions exist where the emulator predicts low function values and/or where there is high uncertainty in the prediction. There is a small non-implausible island at the top left of the space which exists even after wave 4, where the true function is relatively high and thus not of interest. This part of the space can only be discarded once a training sample is placed near that region, which can be seen in wave 5.

We now fix the sequence of distributions implied by the GP fits based on perfect uniform sampling and explore the ability of SMC for sampling this sequence. Here $M = 5$K particles are used. A random subset of 2.5K of these samples for each wave are shown as light grey dots in Figure 2. It is clear that the SMC procedure is quite successful at uniformly sampling the black regions. The acceptance rate of the MCMC step remains reasonable throughout the algorithm, between 40-60%. In contrast, by wave 9, the acceptance rate of the brute force approach (i.e. perfect sampler) is 0.2%.

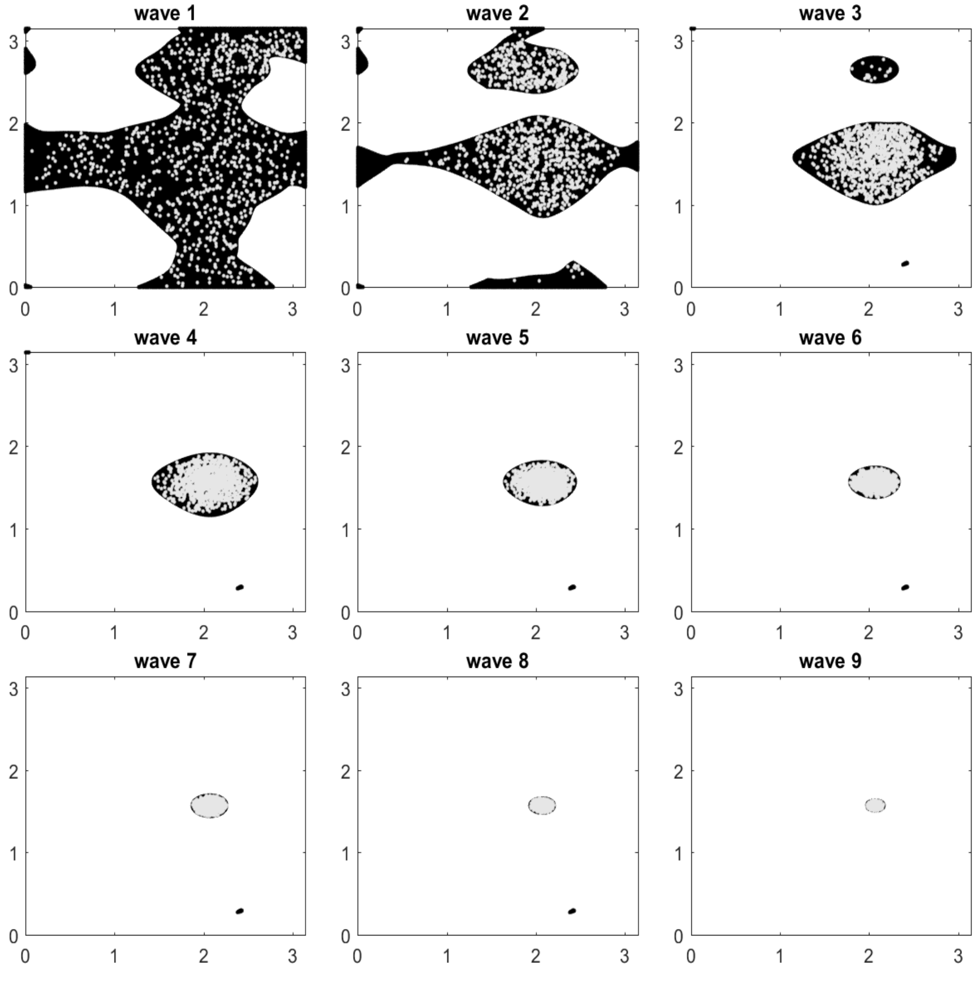

Figure 3: Results for the adhoc approach with the logit transformation to sample from the sequence of distributions obtained by brute force for the toy example. The black regions denote the non-implausible regions after each wave and the light grey dots are samples generated by a more standard approach that might be used in history matching.

We compare the output of the SMC approach with a more standard approach that might be adopted in the history matching literature. A logit transformation is applied to each component of θ to make the parameter space unbounded. Then, the sampling distribution is specified as a multivariate normal distribution with a mean and covariance that is calculated from the existing $M = 5\text{K}$ particles satisfying the constraint for the next wave. Sampling from this normal distribution continues until $N = 5\text{K}$ samples are generated that satisfy all waves to date. The samples from this approach after each wave are shown in light grey dots in Figure 3. It is evident from these plots that this approach under-represents and over-represents in different spots within the black region. Some of the black regions are essentially ignored, which implies that these regions may remain completely unexplored by the history matching algorithm, which could potentially be useful regions of the parameter space. We also use some ideas from this paper and apply the marginal cdf transform (rather than logit transform) and fit a multivariate normal distribution on this space. The results are shown in Figure 4. The results are better than for the logit transform, but ultimately it remains clear that uniform sampling is not achieved. The acceptance rate of the SMC approach is typically lower than the adhoc approaches, since an MCMC kernel may reject a proposal even if it satisfies all relevant waves. However, this is a necessary price to pay to guarantee more reliable sampling of the non-implausible space.

The example here in two dimensions already demonstrates the multi-modal and irregular sequence of distributions that can be generated from the history matching process, and one can imagine how much more complex this sequence might be in higher dimensions. The next example, which involves seven parameters, provides some insight into this.

4.2 Hydrology Example

Here the objective is to find parameter values of a Rainfall-runoff model (RRM) that lead to good predictions of a stream flow time-series given input time series of rainfall and potential evapotranspiration. RRMs are hydrological models that conceptualise the process by which precipitation across a catchment is transformed into water flowing in a river or stream (surface water). The mechanics of an RRM are typically described by a set of water balance equations (differential or difference equations), involving a number of conceptual reservoirs or stores of water in a catchment. In this example, we consider a simple, spatially-lumped (in the sense that the conceptual water stores and precipitation are not modelled as spatially-distributed processes) RRM used by Schoups and Vrugt (2010) and Schoups et al. (2010), which is derived from the FLEX model of Fenicia et al. (2007).

The RRM consists of four conceptual reservoirs of water in a catchment: (i) an interception reservoir that accounts for precipitation intercepted by vegetation; (ii) a soil water reservoir (the unsaturated zone); (iii) a fast-reacting reservoir with relatively short residence time, so that water in this store appears as surface water relatively quickly; and (iv) a slow-reacting reservoir, for which there is a relatively long residence time before water in this store emerges as surface water. The stores identified in (iii) and (iv) give rise to surface water components that are sometimes referred to by hydrologists as quickflow and slowflow (or baseflow).

The states of the four reservoirs (interception, unsaturated, fast and slow) at time t are denoted I_t, U_t, F_t and S_t respectively, and in order to remain independent of the catchment area, these

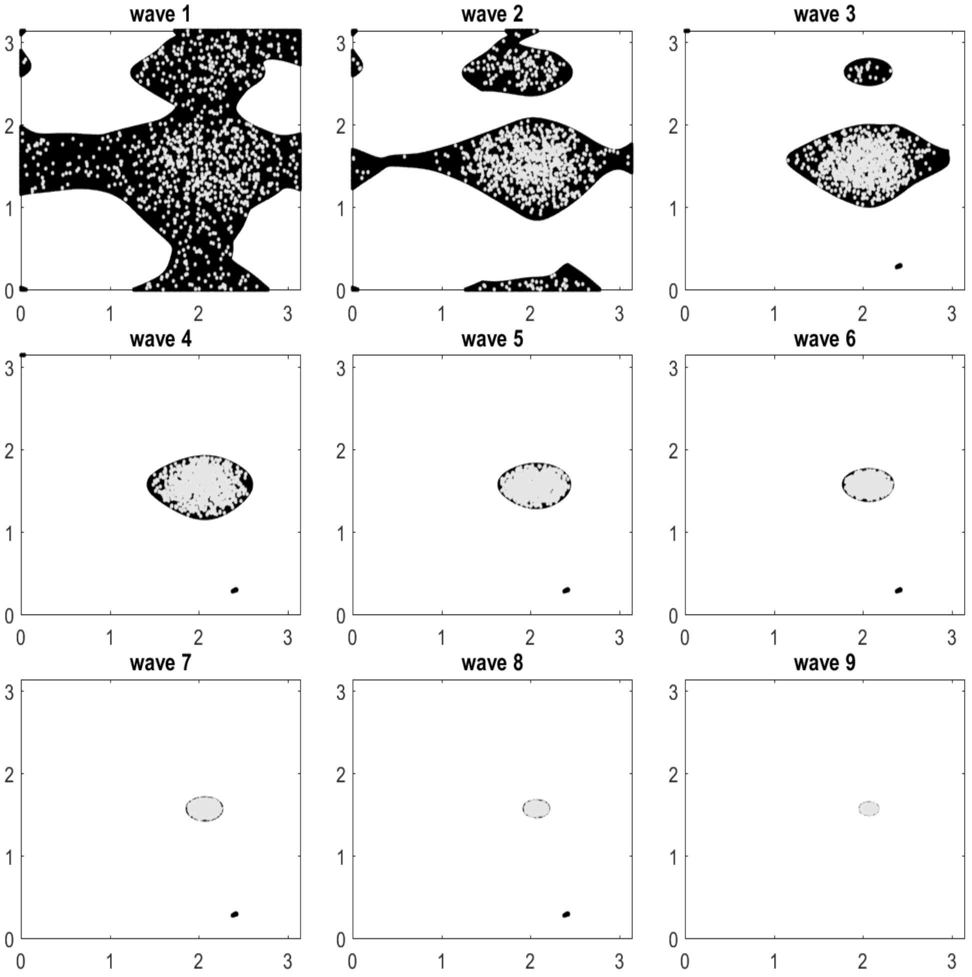

Figure 4: Results for the adhoc approach with the kde transformation applied to the marginals to sample from the sequence of distributions obtained by brute force for the toy example. The black regions denote the non-implausible regions after each wave and the light grey dots are generated with the described ad-hoc approach.

storages are measured in the same units as rainfall (mm). The interception reservoir intercepts precipitation (measured) which enters at rate P_t (mm/day), and can fill to a maximum storage capacity of $I_{\max}$. Because some of the precipitation over the catchment is retained in the interception reservoir, the effective precipitation rate, P_t^e (mm/day), entering the soil water store, is less than the measured precipitation and is calculated as $P_t^e = P_t - (I_{\max} - I_t)$. The interception reservoir also loses water to the atmosphere via evapotranspiration at a rate that is calculated as $E_t^I = \min(E_t^p, I_t)$ (mm/day), where E_t^p is the potential evapotranspiration rate (mm/day), typically calculated based on environmental factors (temperature, solar radiation, wind speed etc) using the Penman-Monteith equation (Monteith, 1965). The water balance for the interception reservoir is therefore

$$\frac{dI_t}{dt} = P_t - E_t^I - P_t^e.$$

Our model assumes that the soil water store, U_t, has a maximum capacity of $U_{\max}$ (mm), and its water balance is governed by the equation

$$\frac{dU_t}{dt} = P_t^e - Q_t^f - E_t^a - Q_t^s,$$

where P_t^e is the effective rainfall rate at time t as previously defined, Q_t^f is the runoff rate (mm/day) that enters the fast-reacting reservoir, E_t^a is the actual evapotranspiration (mm/day) and Q_t^s (mm/day) is the percolation rate (mm/day) into the slow-reacting reservoir. The fluxes Q_t^f, Q_t^s and E_t^a are modelled as

$$\begin{aligned} Q_t^f &= P_t^e f\left(\tfrac{U_t}{U_{\max}}; \alpha_F\right), \\ E_t^a &= (E_t^p - E_t^I) f\left(\tfrac{U_t}{U_{\max}}; \alpha_E\right), \quad \text{and} \\ Q_t^s &= Q_{\max}^s f(\tfrac{U_t}{U_{\max}}; \alpha_S), \end{aligned}$$

where $f(U;\alpha) = \frac{1-e^{-\alpha U}}{1-e^{-\alpha}}$ is a sigmoidal function that is monotonically increasing in U. $Q_{\max}^s$ is the parameter defining the maximum percolation rate into the slow-reacting reservoir, and α_F, α_E and α_S ($\neq 0$) are parameters governing the rate of change of the sigmoidal function for each of the three fluxes. The fast-reacting reservoir and slow reacting reservoirs have water balances, respectively of

$$\begin{aligned} \frac{dF_t}{dt} &= Q_t^f - Q_t^F \quad \text{and} \\ \frac{dS_t}{dt} &= Q_t^s - Q_t^S, \end{aligned}$$

where $Q_t^F = K_F F_t$ (mm/day) and $Q_t^S = K_S S_t$ (mm/day). Finally, streamflow is then modelled as $Q_t = Q_t^F + Q_t^S$ and can be transformed to a volume per unit time (consistent with observations of flow) by multiplying by the catchment area.

In its entirety, the RRM has eight parameters, but following the approach used by Schoups and Vrugt (2010) we fix α_S to 1×10^{-6} resulting in a percolation rate that is effectively linearly related to the storage U_t. Fixing this parameter, truncates the parameter vector to $\theta = (I_{\max}, U_{\max}, Q_{\max}^s, \alpha_F, \alpha_E, K_F, K_S)^\top$.

Table 1: Parameters of the rainfall-runoff model and prior uncertainty ranges.

Parameter	Symbol	Lower	Upper	Units
Maximum interception	$I_{\max}$	1	10	mm
Soil water storage capacity	$U_{\max}$	10	1000	mm
Maximum percolation rate	$Q^{s}_{\max}$	0	100	mm/day
Evaporation parameter	α_E	1E-6	100	–
Runoff parameter	α_F	-10	10	–
Time constant, fast reservoir	K_F	0	10	days
Time constant, slow reservoir	K_S	0	150	days

Schoups and Vrugt (2010) calibrate the model above to daily precipitation, evaporation and streamflow data for the Guadalupe River basin at Spring Branch, Texas, USA. These authors devised plausible ranges for the seven parameters outlined above, which we provide in Table 1.

To quantify the discrepancy between the data and simulation, denoted $\rho_p(\theta)$, we consider the following relative distance

$$\rho_p(\theta) = \sum_{t=1}^{T} \frac{(y^t_{\text{obs}} - y^t_\theta)^2}{y^t_{\text{obs}}},$$

where y^t_{obs} and y^t_θ are the observed and simulated streamflow at time t. The number of time points is T, where $T = 1827$ here. Matlab code for this hydrology model is available as Example 6 in the DREAM package (Vrugt, 2016), which is available at `http://www.pc-progress.com/en/onlinehelp/dream1/DREAM_Suite.html?Demoexamples.html`.

For the history matching procedure, the implausibility for each sample is computed as $\mathcal{I}(\theta) = \rho_p(\theta) - r \times s_d(\theta)$ where $\rho_p(\theta)$ and $s_p(\theta)$ is the mean prediction of the function and the standard deviation from the currently fitted GP, respectively. The exploration parameter is set at $r = 3$. The cut-off for implausibility at wave w is given by $\mathcal{I}_w(\theta) > c_w$ where c_w is selected adaptively such that exactly half of the surviving samples satisfy the cut-off.

Even for moderate dimensional problems the brute force approach to history matching is infeasible. Thus we explore our SMC approach for this task. We use $M = 10$K particles in the SMC and $N = 1$K training samples for the GP at each wave. For illustrative purposes we use 29 waves.

The SMC procedure appears to be successful in performing the history matching. However, the acceptance rates of the MCMC step are around 7-18%, which are significantly smaller than that for the toy example above. The evolution of the MCMC acceptance probability over the waves is shown as the solid plot in Figure 5. The acceptance rate of 7% might be considered small for a seven dimensional parameter space, which indicates that the non-implausible region is difficult to explore effectively. This is confirmed in Figure 6, which shows bivariate plots of the SMC samples after various waves. It is evident that even the marginal and bivariate distributions are complex, with samples also appearing in disconnected regions. In particular, some of the boundaries of the parameter space cannot be ruled out as implausible early in the process.

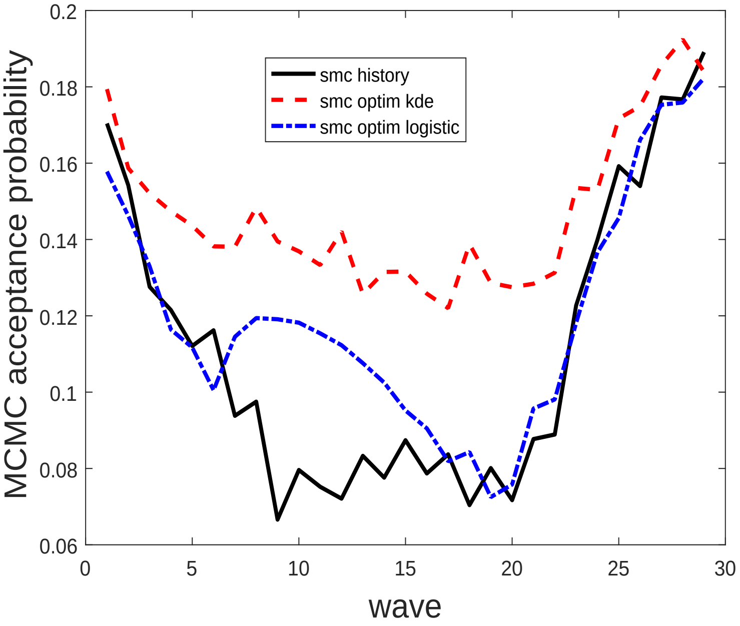

Figure 5: Acceptance probability of the MCMC step when applying the SMC history matching method (solid), SMC optimisation method with kde transforms for the marginals (dash) and the SMC optimisation method with logistic transforms for the marginals (dot-dash).

It turns out that a careful design of the MCMC proposal is required for the SMC procedure to perform well enough. We tried also using the logistic function and the cdf of the beta distribution (after scaling) to transform the marginals. However, given the significant multimodality present, we found that these approaches resulted in a very small MCMC acceptance probability that decreased rapidly. The value of R became too large for the SMC sampler to be computationally feasible. This further demonstrates the significantly complex sampling problem that arises from history matching.

For comparison purposes we also run a standard SMC optimisation approach (i.e. no emulator) with $M = 10$K particles. The sequence of distributions is defined by the 0.5 quantile of the distances in the current particle set. The same type of MCMC kernel is used in the move step. We again perform 29 waves. The bivariate plots after certain iterations of this approach are shown in Figure 7. It is evident, when comparing to Figure 6, that relatively regular bivariate distributions of the parameters arise from the SMC optimisation. Hence, the complexity of the joint distributions as seen in Figure 6 is mostly an artifact of the history matching method rather than being a result of the hydrology model. The difficulty in sampling the parameter space is in part reflected in the acceptance probability of the MCMC kernel of the SMC history matching and optimisation approaches. Using the same type of proposal in the MCMC kernel, it is clear from Figure 5 that SMC optimisation has a much higher acceptance rate than history matching (compare the solid plot with the dash plot). Furthermore, the distributions arising from SMC optimisation are regular enough that it is not necessary to

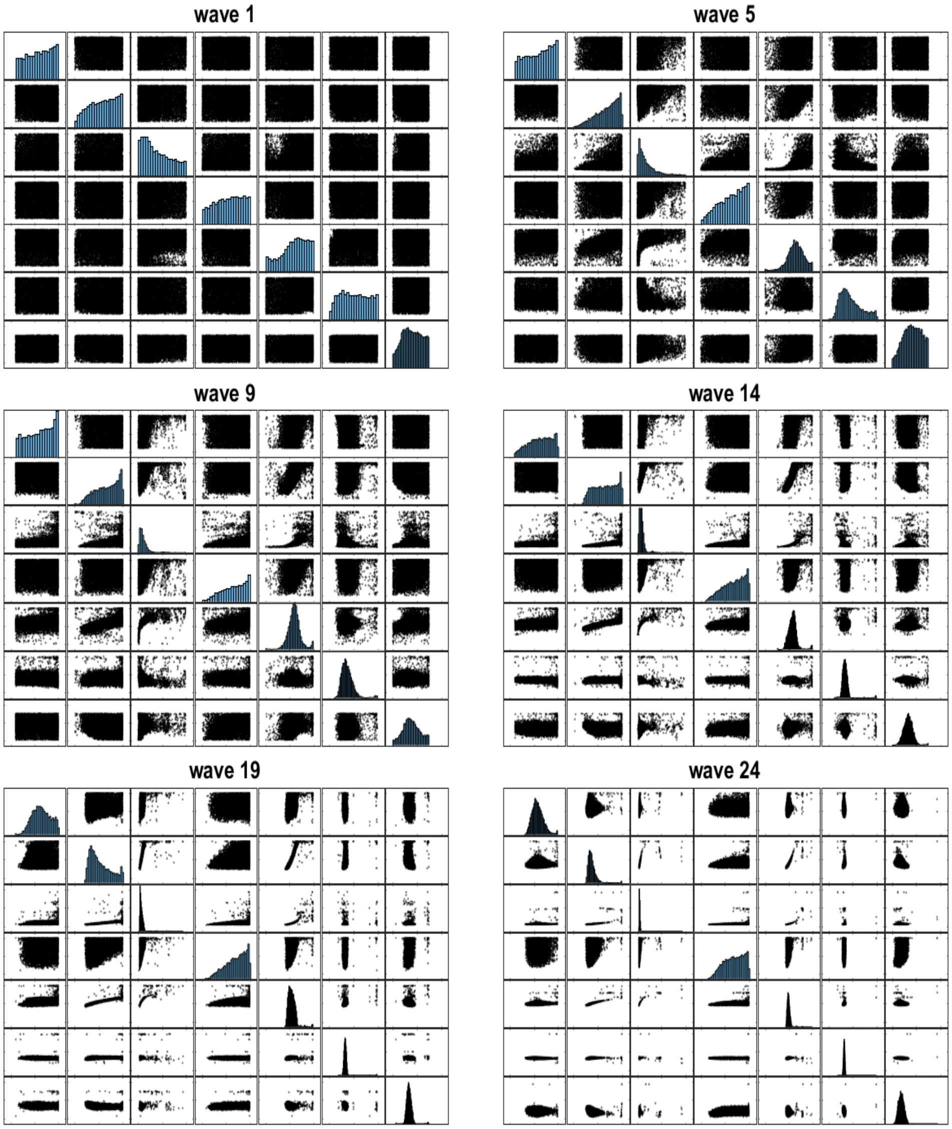

Figure 6: Bivariate scatterplots of the parameters (with marginal histograms along the diagonals) when the SMC history matching method is applied to the hydrology example.

resort to using kde fitting of the marginals as presented in Algorithm 3. Figure 5 shows that a simple logistic transform of the marginals leads to reasonable acceptance rates (dot-dash plot).

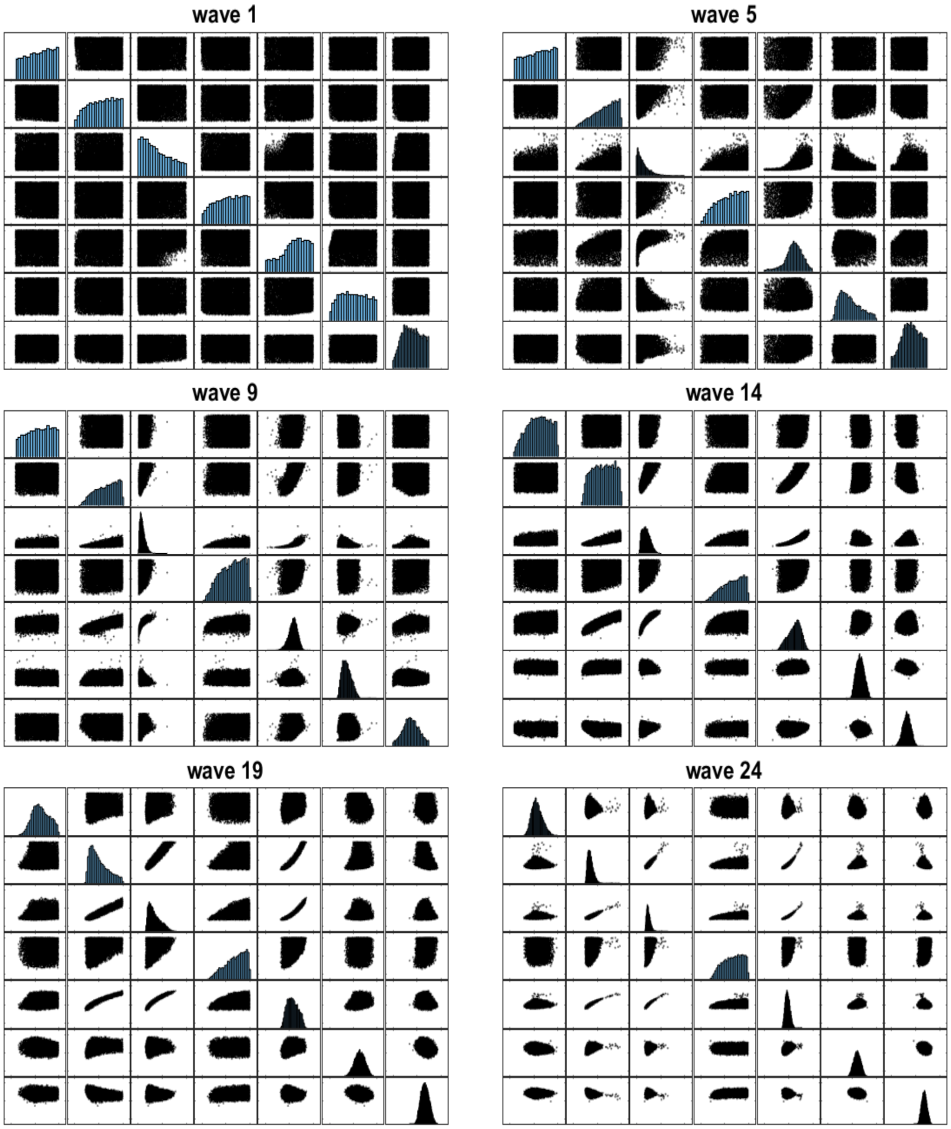

Figure 7: Bivariate scatterplots of the parameters (with marginal histograms along the diagonals) when SMC optimisation is applied to the hydrology example.

Of course these results do not imply that history matching should be dismissed; over the 29 waves, history matching uses more than two orders of magnitude less simulations of the model, which is critical in applications where history matching is adopted. However, the results do highlight the challenges associated with history matching and how SMC history matching can help to address those challenges.

The above hydrology example uses a deterministic model, whereas the next example considers a stochastic process and the implausibility measure depends on an unbiased likelihood estimator.

4.3 Gene Network Example

Golightly and Wilkinson (2005), and more recently Drovandi et al. (2017), consider a Markov jump process for an autoregulatory gene network consisting of four species DNA, RNA, P and P_2. See Golightly and Wilkinson (2005) for more details. The system contains eight possible reactions

$$\begin{aligned} \mathrm{DNA}+\mathrm{P}_2 &\xrightarrow{c_1\mathrm{DNA}\times\mathrm{P}_2} \mathrm{DNA}\cdot\mathrm{P}_2, & 2\mathrm{P} &\xrightarrow{c_5\mathrm{P}(\mathrm{P}-1)/2} \mathrm{P}_2, \\ \mathrm{DNA}\cdot\mathrm{P}_2 &\xrightarrow{c_2(k-\mathrm{DNA})} \mathrm{DNA}+\mathrm{P}_2, & \mathrm{P}_2 &\xrightarrow{c_6\mathrm{P}_2} 2\mathrm{P}, \\ \mathrm{DNA} &\xrightarrow{c_3\mathrm{DNA}} \mathrm{DNA}+\mathrm{RNA}, & \mathrm{RNA} &\xrightarrow{c_7\mathrm{RNA}} \emptyset, \\ \mathrm{RNA} &\xrightarrow{c_4\mathrm{RNA}} \mathrm{RNA}+\mathrm{P}, & \mathrm{P} &\xrightarrow{c_8\mathrm{P}} \emptyset, \end{aligned}$$

where k is a conservation constant (number of copies of the gene) and $\boldsymbol{c} = (c_1, \dots, c_8)$ are the rate constants governing the speed at which the system evolves. We consider the same scenario in Golightly and Wilkinson (2005) where data are simulated using rate values $\boldsymbol{c} = (0.1, 0.7, 0.35, 0.2, 0.1, 0.9, 0.3, 0.1)$, with $k = 10$, and initial species levels $(\mathrm{DNA}, \mathrm{RNA}, \mathrm{P}, \mathrm{P}_2) = (5, 8, 8, 8)$. We simulate equi-spaced data as the next 100 observations (on all species) recorded at 0.5 unit time intervals. Note that we assume these data are observed without error. As in Fearnhead et al. (2014), we take independent half-Cauchy priors for the parameters, with density $p(c_i) \propto 1/(1 + 4c_i^2)$, $c_i > 0$ for $i = 1, \dots, 8$. We remove the positivity constraint on the rate parameters by working on the log scale, that is, with $\theta_i = \log c_i$ for $i = 1, \dots, 8$.

This model does not have a computationally tractable likelihood function. One approach to perform inference for such models is to assume that each species is observed with Gaussian error with a standard deviation of σ (Holenstein, 2009). The likelihood of the implied state space model can be estimated unbiasedly with a particle filter using J particles (Gordon et al., 1993). More accurate inferences are obtained with a low value of σ, with the correct posterior obtained in the limit as $\sigma \to 0$. However as σ decreases, more particles (J) are required to obtain an accurate likelihood estimate, increasing the computation. For illustration purposes, we use $\sigma = 0.6$ and $J = 6000$ here.

For the history matching procedure, the emulator is trained on values of $\log(-\log \hat{f}(y_{\text{obs}}|\theta))$ as the output, where $\hat{f}(y_{\text{obs}}|\theta)$ is the estimated likelihood. We take the log twice to improve the capacity of the emulator to provide a good fit (as advocated by Wilkinson (2014)). Since the prior is relatively vague, many datasets from the prior predictive distribution are far from the observed data. When some rate parameters are relatively large, simulation from the model can be very expensive. To overcome this, if any of the four species reaches a population size of 100 or the simulation is taking too long, the likelihood estimation procedure is terminated early and $\log \hat{f}(y_{\text{obs}}|\theta)$ is set to the smallest properly estimated log-likelihood initially drawn from the prior, which is relatively very small. This discourages exploring such areas of the parameter space in future waves. For parameter configurations that lead to an estimated likelihood of numerically 0, we also set the log-likelihood estimate to the small value mentioned above. The implausibility for each sample is computed as $\mathcal{I}(\theta) = y_p(\theta) - r \times s_d(\theta)$ where $y_p(\theta)$

and $s_p(\theta)$ is the mean prediction and the standard deviation of the predicted output from the currently fitted GP, respectively. The exploration parameter is set at $r = 3$. The cut-off for implausibility at wave w is given by $\mathcal{I}_w(\theta) > c_w$ where c_w is selected adaptively such that exactly half of the surviving samples satisfy the cut-off.

We perform 30 waves of our SMC history matching method. Boxplots of the output $\log(-\log \hat{f}(y_{\text{obs}}|\theta))$ from the training points at each wave are shown in Figure 8. Even after 20 waves, some of the training points generated by the SMC history matching method are not consistent with the data.

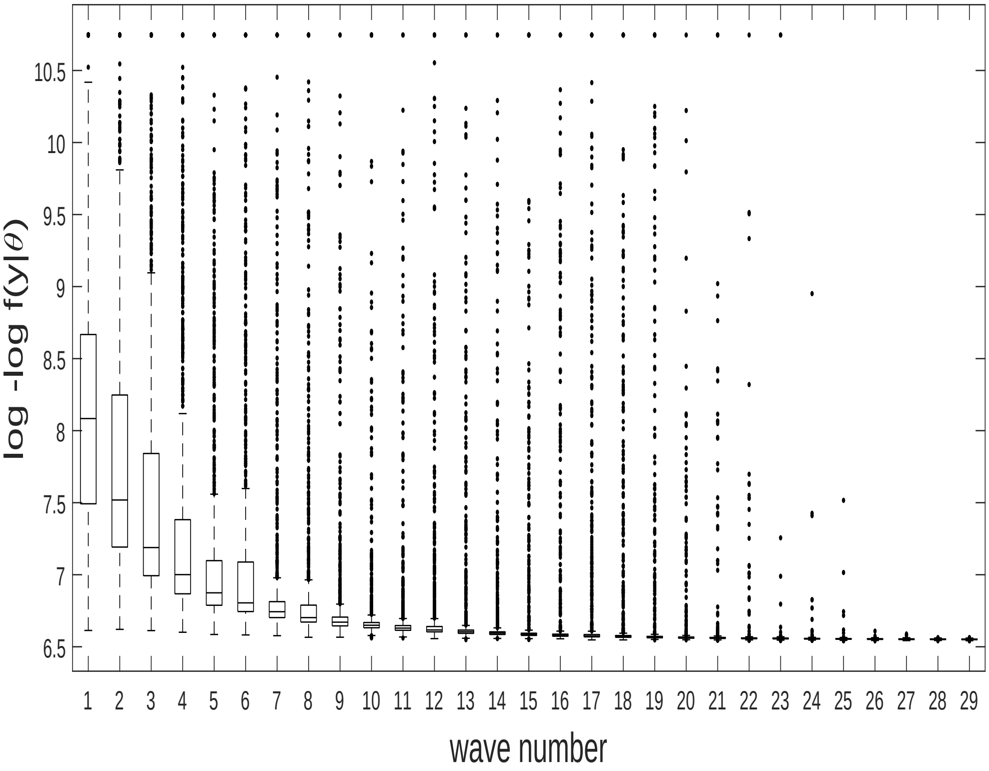

Figure 8: Boxplots of the output $\log(-\log \hat{f}(y_{\text{obs}}|\theta))$ from the training points at each wave for the gene network example. The largest value of the outliers typically corresponds to a parameter value that simulates data far from the observed data and/or can take an excessive time to simulate.

For comparison purposes we also run a standard SMC sampler with 1K particles to sample from the posterior distribution. The SMC sampler we use is based on likelihood annealing and uses the adaptive features described in South et al. (2016), which adaptively determines the annealing temperatures and the number of MCMC repeats in the move step as we find it works well enough here. Here we use a multivariate normal random walk proposal in the move step. The marginal posterior distributions, together with the marginal distributions obtained from SMC history matching after 19, 24 and 29 waves, are shown in Figure 9. The SMC procedure only requires 11 intermediate temperatures to reach the posterior, which are roughly 0.0004, 0.0014, 0.005, 0.0127, 0.028, 0.056, 0.104, 0.18, 0.30, 0.48 and 0.74. In contrast, even after 24 waves the distributions implied by SMC history matching are less precise than the posterior. This highlights that a relatively large number of waves are required to eliminate poor parts of the parameter space. Figure 10 plots the MCMC acceptance rate of the history matching and Bayesian SMC methods over the iterations. It is clear that the Bayesian SMC method produces a higher acceptance rate, even with a simpler proposal distribution. This again highlights the difficult sampling problem generated by history matching.

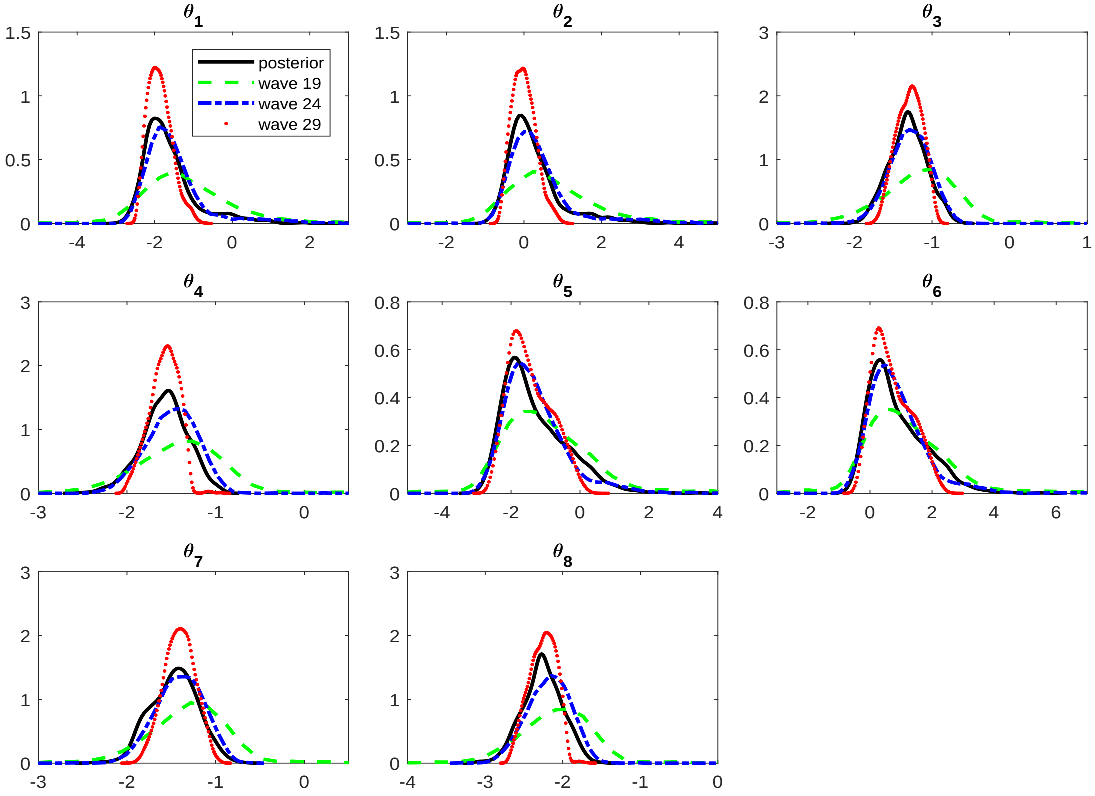

Figure 9: Marginal distributions obtained for the gene network example. Shown are the marginal posterior distributions from Bayesian SMC (solid), and the marginal distributions obtained from the output of SMC history matching after 19 (dash), 24 (dot-dash) and 29 (dot) waves.

Despite the above comparisons, history matching only uses 30K model simulations (stopping after wave 24, which might be considered reasonable, would result in 25K model simulations) whilst the Bayesian SMC method uses roughly 460K model simulations with only 1K SMC particles (SMC history matching had 10K particles).

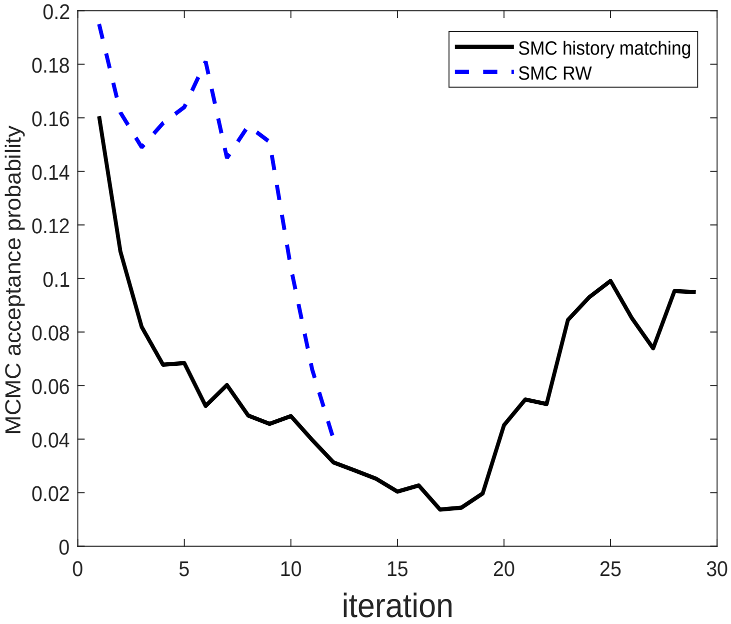

Figure 10: Acceptance probability of the MCMC step when applying the SMC history matching method (solid) and the Bayesian SMC method with a multivariate normal random walk (RW) proposal (dash) to the gene network example.

5 Discussion

In this paper we have developed a novel algorithm based on SMC for history matching that is adaptive and offers a principled method for sampling from the non-implausible space at each wave. Our algorithm reveals in greater detail the significant complexity of the probability distribution associated with the non-implausible space that arises from history matching.

We suggest not to be dismissive of history matching but advise practitioners that more care and computation may be required to help ensure that potentially important pockets of the parameter space are not ignored. Clearly history matching is an important method in the context of expensive simulators. McKinley et al. (2017) demonstrate in a complex stochastic epidemic model that history matching is able to determine parameter regions that lead to close matches with observed outputs whereas a more conventional SMC ABC algorithm that requires significantly more model simulations is not able to in a feasible amount of time.

Our method is semi-automatic in that it adaptively chooses the implausibility thresholds and the number of MCMC repeats, but the practitioner is still required to choose a suitable proposal distribution for the MCMC kernel. Even in our applications of moderate dimension we found that a simple multivariate normal random walk, which is commonly used in the SMC literature (e.g. Chopin (2002) and Drovandi and Pettitt (2011b)), is not efficient enough. Here we developed a method based on transforming the marginals with kde estimates before applying a multivariate normal random walk. Although we found some success with this approach, the MCMC acceptance rates were sometimes small, which significantly increased computational burden. An extension of the slice sampling method used in Andrianakis et al. (2017) may prove useful. The SMC approach of Schuster et al. (2017), called kernel SMC, uses estimated derivatives and local covariance estimates to improve sampler efficiency. In summary, more research in determining a suitable MCMC kernel that is able to explore irregular, multi-modal distributions without derivative information, which can arise from history matching, is required.

In this paper, the emulator for wave w is trained only on points sampled from the non-implausible space at that wave. However, it is possible to apply the emulator to all training points generated up to wave w. An alternative approach to training emulators is that rather than proceeding in waves of distinct training sets, training samples are sequentially appended to an existing training set stemming from a method called Bayesian optimisation (Mockus, 2012). We might be interested in placing additional training points where there is significant uncertainty in the emulator's prediction. It is important to note though, that with certain emulators such as GPs the prediction cost is $\mathcal{O}(N)$ where N is number of training points. See Holden et al. (2015) for further discussion on different emulator training strategies. We suggest that SMC might also be useful in this sequential training approach, and we leave that for future research.

To simplify the explanation of the ideas of this paper we assumed that only one output required emulation. However, in many realistic applications (e.g. Vernon et al. (2014) and Andrianakis et al. (2015)) there are multiple outputs which are emulated separately. One possible approach as described in Andrianakis et al. (2015) is to take the overall implausibility measure as the maximum implausibility over all outputs. In this case an appropriate implausibility cut-off threshold is even less clear, which further motivates our semi-automated method. Another approach is to introduce outputs in some sequential fashion, focussing initially on the outputs that are emulated accurately. It is possible to extend our SMC approach to accommodate this heuristic by modifying the sequence of distributions that SMC depends on to include an indicator highlighting which outputs have been introduced. One approach would sequentially introduce outputs one-at-a-time once the already introduced outputs are satisfied. The output to introduce could be adaptively chosen by selecting the one that is most accurately emulated out of the outputs not yet introduced. The implausibility thresholds for each introduced output can be determined adaptively using the same approach as in our paper.

Acknowledgements

Christopher Drovandi was supported by an Australian Research Council's Discovery Early Career Researcher Award funding scheme (DE160100741). David Nott was supported by a Singapore Ministry of Education Academic Research Fund Tier 2 grant (MOE2016-T2-2-135). Christopher Drovandi is grateful to Leah South for many discussions on sequential Monte Carlo methods and for providing some useful code.

References

Andrianakis, I., McCreesh, N., Vernon, I., McKinley, T. J., Oakley, J. E., Nsubuga, R. N., Goldstein, M., and White, R. G. (2017). Efficient history matching of a high dimensional individual-based HIV transmission model. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):694–719.

Andrianakis, I., Vernon, I. R., McCreesh, N., McKinley, T. J., Oakley, J. E., Nsubuga, R. N., Goldstein, M., and White, R. G. (2015). Bayesian history matching of complex infectious disease models using emulation: a tutorial and a case study on hiv in uganda. *PLOS Computational Biology*, 11(1):e1003968.

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725.

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika*, 89(3):539–552.

Craig, P. S., Goldstein, M., Seheult, A. H., and Smith, J. A. (1997). Pressure matching for hydrocarbon reservoirs: a case study in the use of Bayes linear strategies for large computer experiments. In *Case Studies in Bayesian Statistics*, pages 37–93. Springer.

Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436.

Del Moral, P., Doucet, A., and Jasra, A. (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, 22(5):1009–1020.

Drovandi, C. C., Cusimano, N., Psaltis, S., Lawson, B. A., Pettitt, A. N., Burrage, P., and Burrage, K. (2016). Sampling methods for exploring between-subject variability in cardiac electrophysiology experiments. *Journal of The Royal Society Interface*, 13(121):20160214.

Drovandi, C. C., Moores, M. T., and Boys, R. J. (2017). Accelerating pseudo-marginal MCMC using Gaussian processes. *To appear in Computational Statistics & Data Analysis.*

Drovandi, C. C. and Pettitt, A. N. (2011a). Estimation of parameters for macroparasite population evolution using approximate Bayesian computation. *Biometrics*, 67(1):225–233.

Drovandi, C. C. and Pettitt, A. N. (2011b). Using approximate Bayesian computation to estimate transmission rates of nosocomial pathogens. *Statistical Communications in Infectious Diseases*, 3(1):2.

Fearnhead, P., Giagos, V., and Sherlock, C. (2014). Inference for reaction networks using the linear noise approximation. *Biometrics*, 70(2):457–466.

Fearnhead, P. and Taylor, B. M. (2013). An adaptive sequential Monte Carlo sampler. *Bayesian Analysis*, 8(2):411–438.

Fenicia, F., Savenije, H. H., Matgen, P., and Pfister, L. (2007). A comparison of alternative multiobjective calibration strategies for hydrological modeling. *Water Resources Research*, 43:W03434.

Golightly, A. and Wilkinson, D. J. (2005). Bayesian inference for stochastic kinetic models using a diffusion approximation. *Biometrics*, 61(3):781–788.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *Radar and Signal Processing, IEE Proceedings F*, volume 140, pages 107–113.

Holden, P., Edwards, N., Hensman, J., and Wilkinson, R. (2015). ABC for climate: dealing with expensive simulators. *arXiv:1511.03475*.

Holenstein, R. (2009). *Particle Markov Chain Monte Carlo*. PhD thesis, The University Of British Columbia.

Iman, R. L. (2008). *Latin hypercube sampling*. Wiley Online Library.

McKinley, T., Vernon, I., Andrianakis, I., McCreesh, N., Oakley, J., Nsubuga, R., Goldstein, M., and White, R. (2017). Approximate Bayesian computation and simulation-based inference for complex stochastic epidemic models. *To appear in Statistical Science*.

Mockus, J. (2012). *Bayesian approach to global optimization: theory and applications*, volume 37. Springer Science & Business Media.

Molga, M. and Smutnicki, C. (2005). Test functions for optimization needs. *http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf*.

Monteith, J. (1965). Evaporation and environment. *Symposia of the Society for Experimental Biology*, 19:205–224.

Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*. SIAM.

Picchini, U. (2014). Inference for SDE models via approximate Bayesian computation. *Journal of Computational and Graphical Statistics*, 23(4):1080–1100.

Pukelsheim, F. (1994). The three sigma rule. *The American Statistician*, 48(2):88–91.

Rasmussen, C. E. and Williams, C. K. (2006). *Gaussian processes for machine learning*, volume 1. MIT press Cambridge.

Schoups, G., Vrugt, J., Fenicia, F., and van de Giesen, N. (2010). Corruption of accuracy and efficiency of Markov chain Monte Carlo simulation by inaccurate numerical implementation of conceptual hydrologic models. *Water Resources Research*, 46:W10530.

Schoups, G. and Vrugt, J. A. (2010). A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic, and non-Gaussian errors. *Water Resources Research*, 46:W10531.

Schuster, I., Strathmann, S., Paige, B., and Sejdinovic, D. (2017). Kernel sequential Monte Carlo. *arXiv:1510.03105v4*.

South, L. F., Pettitt, A. N., and Drovandi, C. C. (2016). Sequential Monte Carlo for static Bayesian models with independent MCMC proposals. *https://eprints.qut.edu.au/101729/15/101729.pdf*.

Vernon, I., Goldstein, M., and Bower, R. (2014). Galaxy formation: Bayesian history matching for the observable universe. *Statistical Science*, 29(1):81–90.

Vo, B. N., Drovandi, C. C., Pettitt, A. N., and Pettet, G. J. (2015). Melanoma cell colony expansion parameters revealed by approximate Bayesian computation. *PLOS Computational Biology*, 11(12):e1004635.

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.

Wilkinson, R. (2014). Accelerating ABC methods using Gaussian processes. *Journal of Machine Learning Research*, 33:1015–1023.

Williamson, D. and Vernon, I. (2013). Efficient uniform designs for multi-wave computer experiments. *https://arxiv.org/pdf/1309.3520.pdf.*