The BayesChemEng algorithm

Ji Hyun Bak

(Dated: June 20, 2019)

∗ This is an extract of the algorithm-related part of the Supporting Information for a paper, "Bayesian Inference of Aqueous Mineral Carbonation Kinetics for Carbon Capture and Utilization" (Na et al., 2019), provided for convenience along with the Matlab code repository (https://github.com/jihyunbak/BayesChemEng). Please cite Na et al. (2019) if you find the algorithm useful.

In this document, we describe an effective inference method developed to address a set of challenges that arise in systems with large number of components and interactions. Our focus is on a chemical engineering system: for example, one may need to understand the parameters of a model plant before building and running a real one at a larger scale. That is, we are considering a problem where data acquisition is costly, the model is also complicated (parameter space is multi-dimensional), and even a simulation is computationally heavy. Because the plant needs to operate under a variety of conditions, multiple different design variables (the input settings) should be considered; but since the model system is already costly to operate, the test experiments are most likely not repeated many times at a given set of design variables. So there is also a need to integrate information from multiple datasets collected at very different conditions. Our method performs a Bayesian inference of the posterior distribution, using an iterative algorithm to estimate the scale of underlying fluctuation in the responses at the same time.

A. Bayesian inference formulation

Here we describe how we formulate and infer the Bayesian posterior distribution over the parameter space. Our goal is to construct the posterior distribution $\mathcal{P}(\boldsymbol{\theta})$ over the parameters $\boldsymbol{\theta}$, given the observed data, according to the Bayes rule:

$$\mathcal{P}(\boldsymbol{\theta}) \equiv p(\boldsymbol{\theta}|\mathrm{data}) \propto p(\mathrm{data}|\boldsymbol{\theta}) \cdot p(\boldsymbol{\theta}) \tag{1}$$

where $p(\mathrm{data}|\boldsymbol{\theta})$ is the likelihood of observing the data from a model characterized by $\boldsymbol{\theta}$, and $p(\boldsymbol{\theta})$ is the prior distribution. We will discuss how to compute the likelihood and the prior distribution in this section.

A1. Problem setup

Let $\mathbf{x}$ be a set of design variables for an experiment, and $\mathbf{y}$ the observed response. Specifically, we consider the case where the response is a time series: $\mathbf{y} = \{y_1, y_2, \cdots, y_t, \cdots, y_T\}$, written in the form of a T-dimensional vector. Each experiment may report multiple response types, $\{\mathbf{y}_1, \cdots, \mathbf{y}_D\}$, where each response $\mathbf{y}_d$ is a T-dimensional vector. We have $D = 2$ in our problem: $\mathbf{y}_1$ measures the CO_2 removal rate, and $\mathbf{y}_2$ the pH of the solution. In our experiments, the response length T is fixed for $\mathbf{y}_1$ and $\mathbf{y}_2$ from the same experiment, because they are measured simultaneously.

We consider discrete sets of design variables $\{\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(M)}\}$, that correspond to multiple experiments. We have $M = 16$ experiments; M cannot be very large in most cases, because experiments are costly. The length of response T may be different for different experiments.

With $M = 16$ experiments and $D = 2$ response types, we have $M \times D = 32$ datasets; by a single "dataset" we mean $\{\mathbf{x}^{(m)}, \mathbf{y}_d^{(m)}\}$ at a specific (m, d), where $m = 1, \cdots, M$ and $d = 1, \cdots, D$.

Modeled and observed responses. We can model the system using a set of parameters, $\boldsymbol{\theta}$, which are not directly measured and need to be inferred from the experimental data $\{\mathbf{x}, \mathbf{y}\}$. For a fixed response type d and

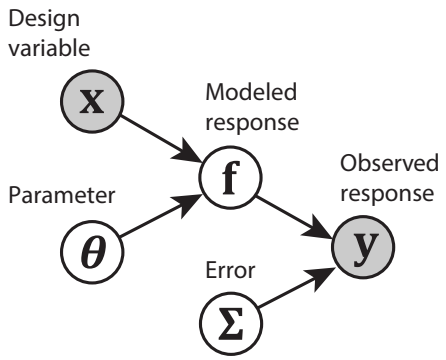

Figure A1: A schematic diagram of the generating model. The two observables, $\mathbf{x}$ and $\mathbf{y}$, are shaded. Reprint of Figure 4a in the main paper (Na et al., 2019).

a fixed experiment $\mathbf{x} = \mathbf{x}^{(m)}$, the model is given in a form

$$\mathbf{y}_{\text{model}} = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}). \tag{2}$$

In the current problem, we have $K = 8$ parameter elements, written in a K-dimensional parameter vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \cdots, \theta_K)$. Each parameter element is varied freely within the physically relevant range.

In general, the experimental data are not described exactly by the model. There is always some discrepancy between the actual response $\mathbf{y}$ and the modeled response $\mathbf{y}_{\text{model}}$, such that:

$$\mathbf{y} = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) + \boldsymbol{\epsilon}. \tag{3}$$

We assume that the error $\boldsymbol{\epsilon}$ is distributed according to a multivariate normal distribution, with a $T \times T$ covariance matrix Σ:

$$\boldsymbol{\epsilon} = \mathbf{y} - \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) \sim \mathcal{N}(\mathbf{0}, \Sigma). \tag{4}$$

This is a simple and reasonable approach when not much is known about the true model; it is also commonly used in related problems (Kastner et al., 2013; Mosbach et al., 2012).

A2. Likelihood

The gaussian error assumption Equation. (4) means that the likelihood of a given dataset (a pair of design variable $\mathbf{x}$ and response $\mathbf{y}$) depends not only on the parameter $\boldsymbol{\theta}$, but also on the covariance matrix Σ that characterizes the error distribution. We can write down the joint likelihood as

$$p(\text{data}|\boldsymbol{\theta}, \Sigma) = p(\{\mathbf{x}, \mathbf{y}\}|\boldsymbol{\theta}, \Sigma) \approx \mathcal{N}(\boldsymbol{\epsilon}|\mathbf{0}, \Sigma). \tag{5}$$

The log of this joint likelihood is written simply as

$$\log p(\text{data}|\boldsymbol{\theta}, \Sigma) \approx -\frac{1}{2}\boldsymbol{\epsilon}^\top \Sigma^{-1} \boldsymbol{\epsilon} - \frac{1}{2}\log\det\Sigma + const. \tag{6}$$

where $\boldsymbol{\epsilon} = \mathbf{y} - \mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$, and the constant terms are independent of $\boldsymbol{\theta}$ or Σ.

Likelihood as a function of θ. The covariance matrix Σ is a parameter of the problem, in the sense that it affects the likelihood of data. However, we are not directly interested in knowing the values of elements in Σ; we are only interested in estimating $\boldsymbol{\theta}$. In other words, Σ is a *nuisance parameter*. Eventually, we are interested in the likelihood as a function of $\boldsymbol{\theta}$ only. This is achieved by marginalizing over Σ:

$$p(\text{data}|\boldsymbol{\theta}) = \int d\Sigma\, p(\text{data}|\boldsymbol{\theta}, \Sigma)\, p(\Sigma). \tag{7}$$

If we fix Σ and let the likelihood be a function of $\boldsymbol{\theta}$, the function $p(\text{data}|\boldsymbol{\theta}, \Sigma)$ is a *conditional* likelihood. To clearly indicate this, it is useful to abbreviate the log conditional likelihood as $L(\boldsymbol{\theta}|\Sigma) \equiv \log p(\text{data}|\boldsymbol{\theta}, \Sigma)$.

A fully Bayesian approach would be to consider the entire space of $(\boldsymbol{\theta}, \Sigma)$, and to marginalize over Σ after obtaining the joint posterior. Indeed, some of the previous works treated each element of Σ as the a parameter of the system and used a non-informative prior (Kastner et al., 2013; Mosbach et al., 2012). Here, we use a more practical approach to estimate the error covariance matrix directly from the data, and keep $\Sigma = \hat{\Sigma}_{m,d}$ fixed for each specific dataset. In other words, we assume

$$p(\text{data}|\boldsymbol{\theta}) \approx p(\text{data}|\boldsymbol{\theta}, \hat{\Sigma}). \tag{8}$$

Our approach is closely related to the empirical Bayes methods (George and Foster, 2000; Robbins, 1956), in the sense that parameters that have least impact to data are directly estimated, instead of being integrated over. We would not call $\hat{\Sigma}$ the hyperparameter of the model, because it governs the likelihood, and not the prior distribution of $\boldsymbol{\theta}$. That said, we note that our viewpoint is equivalent to assuming that the prior on Σ is highly localized around a "true" error covariance matrix $\hat{\Sigma}$ for each dataset, such that the log likelihood is reduced to

$$L(\boldsymbol{\theta}) \equiv \log p(\text{data}|\boldsymbol{\theta}) = \log \int d\Sigma \, \exp\left[L(\boldsymbol{\theta}|\Sigma)\right] \, p(\Sigma|\hat{\Sigma}) \approx L(\boldsymbol{\theta}|\hat{\Sigma}). \tag{9}$$

In this extended sense, estimation of $\hat{\Sigma}$ can be regarded as the estimation of hyperparameters for the prior on Σ. The direct estimation of $\hat{\Sigma}$ for a given dataset is described in the next section.

Likelihood for multiple datasets. With all $M \times D$ datasets, the total likelihood is

$$p(\text{data}|\boldsymbol{\theta}) = \int d\Sigma_{1,1} \cdots \int d\Sigma_{M,D} \, p(\text{data}|\boldsymbol{\theta}, \{\Sigma_{m,d}\}) \prod_{m,d} p(\Sigma_{m,d}), \tag{10}$$

where $\Sigma_{m,d}$ is the error covariance matrix for the specific dataset (m, d), and we use a shorthand $\{\Sigma_{m,d}\} = \{\Sigma_{1,1}, \cdots, \Sigma_{M,D}\}$ to mean the set of all Σ's. Now it becomes clearer that, because each experiment may have a different error statistics and therefore carry a different amount of information, it is important that Σ is considered separately for each individual dataset.

The integral can be greatly simplified if we can assume that each response type is independent of one another, and that experiments at different design variables are also independent. In this limit, the conditional likelihood is conveniently separable:

$$p(\text{data}|\boldsymbol{\theta}, \{\Sigma_{m,d}\}) = \prod_{m=1}^{M} \prod_{d=1}^{D} p(\{\mathbf{x}^{(m)}, \mathbf{y}_d^{(m)}\}|\boldsymbol{\theta}, \Sigma_{m,d}). \tag{11}$$

Consequently, the total log likelihood $L(\boldsymbol{\theta}) \equiv \log p(\text{data}|\boldsymbol{\theta})$ is written in a straightforward sum of dataset-specific log likelihoods:

$$L(\boldsymbol{\theta}) = \log \prod_{m=1}^{M} \prod_{d=1}^{D} \int d\Sigma_{m,d} \, p(\{\mathbf{x}^{(m)}, \mathbf{y}_d^{(m)}\}|\boldsymbol{\theta}, \Sigma_{m,d}) \, p(\Sigma_{m,d}) \tag{12}$$

$$= \sum_{m=1}^{M} \sum_{d=1}^{D} \log \int d\Sigma_{m,d} \, p(\{\mathbf{x}^{(m)}, \mathbf{y}_d^{(m)}\}|\boldsymbol{\theta}, \Sigma_{m,d}) \, p(\Sigma_{m,d}) \tag{13}$$

$$\equiv \sum_{m=1}^{M} \sum_{d=1}^{D} L_{m,d}(\boldsymbol{\theta}), \tag{14}$$

where $L_{m,d}(\boldsymbol{\theta})$ is the single-dataset log likelihood for the (m, d)-th dataset. The (possibly different) leverage of each dataset is taken into account indirectly through the error covariance matrix Σ, which determines how sharp or broad the likelihood is. Therefore, when adding up the log likelihoods from multiple datasets, care should be taken to identify an appropriate $p(\Sigma)$ for each dataset.

A3. Prior

We use a simple range-constraint prior $p(\boldsymbol{\theta})$. That is, we consider a hypercube $\mathcal{C}$ in the parameter space, and assume that the prior distribution is uniform within the hypercube:

$$\begin{aligned} p(\boldsymbol{\theta}) &\propto 1 \quad \text{if} \quad \boldsymbol{\theta} \in \mathcal{C}; \\ &= 0 \quad \text{otherwise.} \end{aligned} \tag{15}$$

The normalization within the hypercube is not important for our current purpose, because the Metropolis-Hastings sampler only cares about the ratio between two probability values.

The prior distribution is represented by a uniform sampling of parameters within the prior hypercube $\mathcal{C}$. We combined the Central Composite Design (CCD), defined by the two ends and a midpoint along each dimension of the parameter space, and the Latin Hypercube Sampling (LHS) (McKay et al., 1979) that generates a well-separated random sample. The CCD samples $1 + 2K + 2^K$ points from the parameter space, with 1 center point, $2K$ points at the two ends of each axis, and 2^K points at the corners of the hypercube. Because the LHS can sample an arbitrary number of points, we can fix the total number of samples according to the tradeoff between accuracy and computational cost. In this case, with $K = 8$, we used $N_{\text{prior}} = 1,000$ samples in total (273 from CCD, and 727 from LHS). This sampling of the prior is used for constructing the surrogate model, as well as for obtaining the first estimate of the scale for the error covariance, as described below.

A4. Posterior

Finally, we can construct the posterior distribution from the likelihood and the prior. In terms of the log probabilities,

$$\log \mathcal{P}(\boldsymbol{\theta}) = L(\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) + const. \tag{16}$$

where $const.$ means terms that are independent of the parameter $\boldsymbol{\theta}$.

We note that in the case of multi-dataset inference, where the integrated posterior distribution is very sharp and often multi-modal, one may get a more informative representation of the parameter space by sampling a *tempered* distribution. This is done by introducing an inverse tempering factor β that scales the log posterior as $\beta \log \mathcal{P}(\boldsymbol{\theta})$; using an analogy to equilibrium distributions in statistical mechanics, the inverse of β plays the role of "temperature" such that a smaller β results in a broader, fuzzier distribution. $\beta = 1$ corresponds to the original posterior distribution, without artificial tempering. It is also possible to use the idea of simulated annealing, in which one starts from a higher temperature ($\beta < 1$) and gradually cool down to reach $\beta = 1$, although we did not use this in the current work.

A5. Use of a surrogate model

For a complicated kinetic model, calculation of the model response $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta})$ while varying the value of $\boldsymbol{\theta}$ can be computationally demanding. To make computations affordable, we approximate the modeled response using a quadratic hyper-surface:

$$\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}) \approx \tilde{\mathbf{f}}(\boldsymbol{\theta}) = c + \mathbf{b}^\top \boldsymbol{\theta} + \boldsymbol{\theta}^\top A \boldsymbol{\theta}. \tag{17}$$

Here A is a symmetric $K \times K$ matrix, $\mathbf{b}$ is a vector of length K, and c is a scalar, where K is the dimensionality of the parameter $\boldsymbol{\theta}$. For each experiment $\mathbf{x}$, we pre-compute the model responses $\mathbf{f}(\mathbf{x}, \boldsymbol{\theta}_n)$ at select parameter values $\{\boldsymbol{\theta}_n\} \sim p(\boldsymbol{\theta})$ that are distributed according to the prior distribution, i.e., uniformly within the hypercube. This allows us to determine the $1 + K + K(K+1)/2$ coefficients in the quadratic expression. The quadratic surface fit was performed separately for each dataset $\{\mathbf{x}^{(m)}, \mathbf{y}_d^{(m)}\}$, to obtain an approximate response function $\tilde{\mathbf{f}}^{(m,d)}$ such that $\mathbf{y}_d^{(m)} \approx \tilde{\mathbf{f}}^{(m,d)}(\mathbf{x}^{(m)}, \boldsymbol{\theta})$.

B. Estimating error covariance from data

Estimating the error covariance matrix Σ from data is in general not feasible, because we do not know the true model *a priori*. Nevertheless, we can obtain a rough estimate by looking at the statistics of the respective time series data, as well as based on the inferred distribution of model parameters.

B1. Parameterizing the covariance matrix

Because the covariance matrix can have a large ($\sim T^2/2$) number of degrees of freedom if fully varied, we first characterize Σ in terms of a smaller number of parameters. We use the separation strategy (Barnard et al., 2000), which is to decompose the covariance matrix as $\Sigma = SRS$. In this decomposition, S is the $T \times T$ diagonal matrix where each diagonal element $S_{tt} = \sigma_t$ is the standard deviation, and R is a $T \times T$ correlation matrix. We make two additional simplifications in this case. First, we assume that the standard deviation is uniform at all t, such that $S = \sigma I$. Second, we assume that the correlation is a function of the time difference only, such that $R_{ij} = k(t_i - t_j)$, where k is some kernel that specifies the correlation function. Here we use a simple covariance kernel in the form $k(\Delta t) \sim \exp(-|\Delta t|/\tau)$. Considering a constant τ for the moment, the (i, j)-th element of the covariance matrix is written as

$$\Sigma_{ij} = \sigma^2 \exp\left(-\frac{|t_i - t_j|}{\tau}\right); \tag{18}$$

this corresponds to the Matérn covariance function with $\nu = 1/2$. We also note that with the exponential kernel, the inverse correlation matrix Σ^{-1} is a tridiagonal matrix with a known analytical form (Rybicki and Press, 1995).

Now the covariance matrix is characterized by two parameters as $\Sigma = \Sigma(\sigma, \boldsymbol{\tau})$, in terms of the overall scale of fluctuation σ and the de-correlation timescale $\boldsymbol{\tau}$. Note that we are actually using a T-dimensional vector $\boldsymbol{\tau}$, rather than a single constant; this will be explained shortly below. The problem of estimating $\hat{\Sigma}$ is now reduced to identifying the best $(\hat{\sigma}, \hat{\boldsymbol{\tau}})$.

B2. Estimating the scale of fluctuation σ

The scale of fluctuation σ is actually a property of the generating process, arising for example from the inherent instability in the many-body kinetics of the system, or from any measurement error. However, at the time of data analysis we do not have access to these sources of fluctuation. Instead, we will take a reverse viewpoint, and assume that σ should be consistent with the average scale of discrepancy between the modeled and the observed response. In other words, we calculate

$$\hat{\sigma}^2 = \frac{1}{T}\sum_{t=1}^{T}\left\langle (\epsilon_t)^2 \right\rangle = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{1}{N}\sum_{n=1}^{N}\left[\mathbf{y} - \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}_n)\right]_t^2\right) \tag{19}$$

where the average $\langle\cdot\rangle$ is taken over the inferred distribution of models, or equivalently the posterior distribution of parameters, sampled by a chain of parameters $\{\boldsymbol{\theta}_n\} \equiv \{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_N\}$.

This leads to a recursive loop, because the value of σ is used to construct the likelihood of data given each model, which is then needed to construct the posterior. We solve the problem by an iterative process, in which the algorithm alternates between performing (i) a posterior inference step, that samples a chain of parameters $\{\boldsymbol{\theta}_n\} \sim p(\boldsymbol{\theta}|\mathrm{data}; \hat{\sigma})$ using a given estimate of σ, and (ii) a error estimate step, which obtains a new estimate of σ using the parameter samples obtained from the previous step. The iterative algorithm will be described in more details in the next section.

The estimate of σ also extends naturally for the case where we infer a *single* posterior distribution out of multiple datasets. In that case, we use a single chain of parameters $\{\boldsymbol{\theta}_n\}$, which samples the multi-dataset posterior constructed by putting together the likelihood of each dataset, using Equation. (14), to estimate a $\hat{\sigma}_{(m,d)}$ for each dataset (m, d) involved in the inference. More specifically, we calculate for each dataset

$$\hat{\sigma}^2_{(m,d)} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{1}{N}\sum_{n=1}^{N}\left[\mathbf{y}_d^{(m)} - \mathbf{f}(\mathbf{x}^{(m)}, \boldsymbol{\theta}_n)\right]_t^2\right), \tag{20}$$

where $\mathbf{f}$ needs to be replaced with the dataset-specific $\tilde{\mathbf{f}}^{(m,d)}$ if surrogate models are used.

B3. Estimating the de-correlation timescale τ

Suppose that a given sampling of a time series response was fine enough to describe all the important features of the system. Now suppose that we design a new experiment that samples at a twofold higher rate, with

twofold more datapoints. Do we get more information? The answer is no, and this is because the sampled points are *correlated*. There is a characteristic timescale τ, such that two datapoints are truly independent only if they are separated more than τ in time.

In this particular system, we find that the timescale may not be homogeneous over the duration of the time series: often there is a narrow window of time in which the reaction happens more actively, after which the system saturates and stays almost constant for a long time. If we were to use a constant timescale for these systems, the contribution from the post-reaction observations may dominate the likelihood. Motivated by this, we extend to a non-homogeneous timescales and write this in a vector $\boldsymbol{\tau} = (\tau_1, \tau_2, \cdots, \tau_T)$, such that $\tau_i = \tau(t_i)$. In this case, the covariance matrix is $\Sigma_{ij} = \sigma^2 R_{ij}$, where R_{ij} is modified as (Rybicki and Press, 1995)

$$R_{ij} = \exp\left(-\int_{t_i}^{t_j} \frac{dt}{\tau(t)}\right) \approx \exp\left(-\frac{|t_i - t_j|}{\bar{\tau}_i}\right), \qquad i < j; \tag{21}$$

where the average timescale for the interval is given by the harmonic mean (assuming the timepoints are uniformly sampled)

$$\frac{1}{\bar{\tau}_i} = \frac{1}{|j-i|} \sum_{\ell=i}^{j} \frac{1}{\tau_\ell}, \qquad i < j. \tag{22}$$

Note that $\bar{\tau}_i$ also depends on the interval, although it is not written explicitly in the notation. This specifies all elements of R_{ij} with $i < j$; the other triangle with $i > j$ is symmetrically determined as $R_{ij} = R_{ji}$.

We estimated the local de-correlation time, τ_i, at each timepoint t_i, by calculating the local rate of change in the response, and asking how long it takes at this rate until the response becomes significantly different from where it started. Specifically, we computed

$$\frac{1}{\hat{\tau}_i} = \frac{|\Delta y/\Delta t|_{t_i}}{\langle \Delta y \rangle}, \qquad \langle \Delta y \rangle = \Delta t \cdot \left\langle \left| \frac{\Delta y}{\Delta t} \right| \right\rangle, \tag{23}$$

where $\Delta y/\Delta t$ is the local rate of response change at a fixed timescale Δt, and $\langle \Delta y \rangle$ is the expected change of response over this Δt, averaged over the entire time series. We used $\Delta t \approx T/10$, which is to say that a change is "local" if it happens within a window of time shorter than about 10% of the total duration of observation. Finally, we added a lower cutoff to take into account the finite duration of observation, which protects $\hat{\tau}_i$ from diverging in the case of very flat (constant) signals:

$$\frac{1}{\hat{\tau}_i} \longrightarrow \frac{1}{T} + \frac{1}{\hat{\tau}_i}. \tag{24}$$

C. Efficient sampling of high-dimensional parameters

We use Markov chain Monte Carlo (MCMC) sampling to approximate the posterior distribution. Our sampling method is based on the Metropolis-Hastings algorithm (Metropolis et al., 1953), but involves multiple iterations to adjust the proposal distribution of the sampler; we discuss the details of the sampling process below. Once we have sampled a chain of parameters, $\{\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_N\}$, the response to a given design variable $\mathbf{x}$ that is predicted by the posterior is:

$$\hat{\mathbf{f}}(\mathbf{x}) = \langle \mathbf{f}(\mathbf{x}) \rangle = \frac{1}{N} \sum_{n=1}^{N} \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}_n). \tag{25}$$

C1. Re-parameterization

For each parameter of the kinetic model (say θ_i) whose prior range spans multiple orders of magnitudes, we re-parameterized it as $\theta_i = 10^{\mu_i}$, and inferred the value of the auxiliary parameter $\mu_i = \log_{10} \theta_i$ in our formulation. This was in order to ensure that the parameter space can be effectively sampled by a finite number of fixed-scale moves within each chain.

C2. Optimizing the sampler

One of the general and major challenges in MCMC sampling is to optimize the proposal distribution so that the sampled chain is "well-mixed". If the proposal distribution is too narrow (chain jumps in small steps), the resulting chain may stay in a limited region of the parameter space, not fully exploring the target distribution. On the other hand, if the proposal distribution is too wide (large steps), the proposed moves may escape the high-density region of the parameter space too quickly, again failing to explore the important features of the target distribution; such off-peak proposals are also more frequently rejected, wasting the trials.

Here we use a multivariate normal distribution $q(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}, \Sigma_{\rm prop})$ with a diagonal covariance matrix $\Sigma_{\rm prop} = {\rm diag}(ds_1^2, \cdots, ds_K^2)$, where $ds_i = (\mathbf{ds})_i$ is the "step size" in the i-th dimension.

Proposal distribution update. For the Metropolis-Hastings algorithm, it is known that maximal efficiency is achieved when the proposal and the resulting chain covariances are related as $ds_i = \alpha_K \cdot {\rm std}(\theta_i)$, where ${\rm std}(\theta_i) = \sqrt{\langle(\theta_i - \langle\theta_i\rangle)^2\rangle}$, and the ratio $\alpha_K = (2.38)/\sqrt{K}$ depends on the dimension of the parameter space K (Gelman et al., 1996; Roberts et al., 1997). It was suggested that by matching the proposal distribution adaptively to the accumulated covariance of the sampled parameters, it is possible to optimize the algorithm adaptively (Haario et al., 2001). Whereas the original idea of Haario et al. (2001) updated the proposal distribution at each trial within a single chain, here we use a semi-adaptive approach of updating at the end of every chain, and repeating the sampling process with multiple rounds of sampling. The semi-adaptive algorithm is an effective as well as economic choice in this case, because our inference naturally involves an iterative process for estimating the error scale. Also in this particular problem, we find that the chain behaves better when we adjust to a slightly smaller step size, which introduces an additional factor $\rho \leq 1$ in the following update rule; we used $\rho = 0.5$ for our analysis. The use of $\rho < 1$ is not justified or guaranteed to converge, but in this application, it seems to help ensure that the sampler collects enough sample (the acceptance rate is not too low) within a finite number of test samples during iteration. Put together, our update rule for the step size for the sampler was

$$ds_i \leftarrow \rho \cdot \alpha_K \cdot {\rm std}(\theta_i), \tag{26}$$

where ${\rm std}(\theta_i)$ means the standard deviation along the i-th dimension of the parameter space, across all parameter values sampled in the previous chain.

Criteria for successful sampling. After each iteration, we check whether the sampled MCMC chain provides a good enough sampling of the target distribution. We considered two criteria to this end. The primary criterion is to have enough "fresh" samples in the chain. We formulate this condition in terms of the timescale of mixing, $n_{\rm mix}$, defined as the number of steps in which the autocorrelation function decays to $1/e$. (Note that the number of steps in the chain has nothing to do with the physical time; the term "timescale" is only used conventionally in the context of Markov chains.) This means that samples separated by less than $n_{\rm mix}$ steps are likely to be correlated. Therefore, a good sample of the target distribution is obtained only if $n_{\rm mix}$ is significantly smaller than the total number of samples in the chain. For a given MCMC chain, we checked if

$$n_{\rm mix} < n^*; \tag{27}$$

we used a cutoff value $n^* = 500$, which is $1/10$ of the chain length used for iteration ($N = 5000$), and $1/20$ of the chain length for a final sampling after the termination of the iteration ($N = 20000$).

Another useful criterion is to look for an "optimal" acceptance rate. It is known that in ideal cases, the Metropolis-Hastings sampler is asymptotically optimal when the acceptance rate (the ratio of accepted trials to all proposed trials in a MCMC chain) is $r = 0.234$ (Roberts et al., 1997) for multivariate distributions, although the optimal value depends on the form of target distribution (Roberts and Rosenthal, 2001), That said, it is also known the efficiency is relatively flat within a broader range, for example up to a factor of 2 from the optimal value (Rosenthal, 2011). Therefore, after sampling each chain, we checked if the acceptance rate falls into an optimal range of

$$r \in (0.1, 0.5), \tag{28}$$

which is a generous criterion. If $r \leq 0.1$, it means that most proposals are rejected, most likely because the step sizes are too large; if $r \geq 0.5$, it usually means that proposed steps are too small (and so adjacent samples are highly correlated), and therefore less informative of the shape of the target distribution.

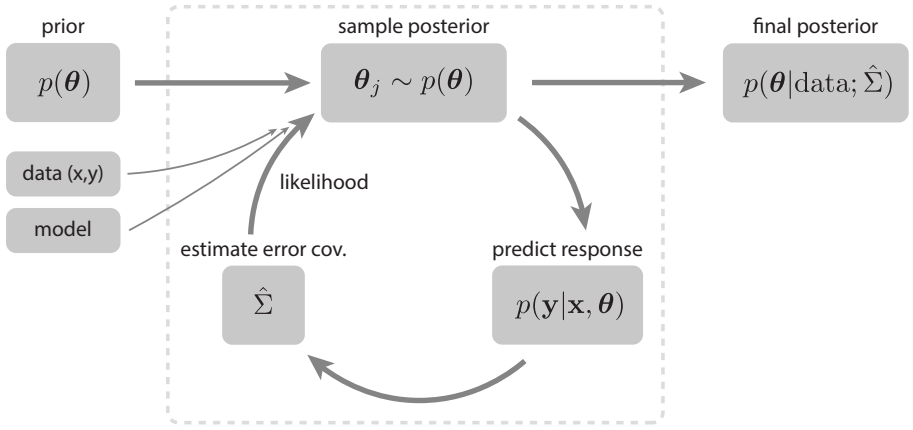

Figure D2: A schematics for the iterative algorithm. Reprint of Figure 4b in the main paper (Na et al., 2019).

D. Iterative sampling of posterior

Finally we outline the iterative sampling algorithm. The goal of the iterative sampling is to simultaneously infer the posterior distribution $\mathcal{P}(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\text{data}; \{\hat{\sigma}, \hat{\boldsymbol{\tau}}\})$ of the model parameters, as well as the best estimate of the error scale $\hat{\sigma}$.

- *Initialization:* For the first round of iteration, the error covariance is initialized by averaging over the samples of prior distribution, $\hat{\sigma}_0 = \hat{\sigma}_{\text{prior}}$. Step sizes were initialized to a constant fraction of the hypercube dimension; in this work, we started with $1/100$ of each hypercube dimension.

- *Iteration:* At each iteration $i = 1, \cdots, i_{\max}$, we construct a posterior distribution using the previous estimate of $\hat{\sigma}_{i-1}$; let $\mathcal{P}_i(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\text{data}; \{\hat{\sigma}_{i-1}, \hat{\boldsymbol{\tau}}\})$ be a shorthand for the posterior distribution at iteration i. Then we sample a chain of $N = 5000$ parameters, $\{\boldsymbol{\theta}\}_i \sim \mathcal{P}_i(\boldsymbol{\theta})$, using the Metropolis-Hastings algorithm and with the multivariate normal proposal distribution characterized by the step sizes $\mathbf{ds}_{i-1}$.

  After each sampling, we use Equation. (19) to obtain a new estimate of $\hat{\sigma}_i$, averaged over the samples $\{\boldsymbol{\theta}\}_i$. We also use Equation. (26) to adjust the step sizes according to the standard deviation of parameter values in $\{\boldsymbol{\theta}\}_i$, so that $\mathbf{ds}_i \propto \text{std}(\{\boldsymbol{\theta}\}_i)$. In addition, we scaled all step sizes uniformly by $\hat{\sigma}_i^2/\hat{\sigma}_{i-1}^2$, to account for the overall broadening/sharpening of the posterior distribution due to the change in $\hat{\sigma}$. The latter scaling is based on the assumption that the uncertainty in the parameter space (related to the optimal step size) would scale linearly with the uncertainty in the response space (related to σ^2); this assumption may not be true, but the adjustment is appears to be effective in our case. In the case of multi-dataset inference, the scaling factor is calculated with the combined $\hat{\sigma}_{\text{joint}}$ for the joint distribution, defined as $1/\hat{\sigma}_{\text{joint}}^2 = \sum_{m,d} 1/\hat{\sigma}_{(m,d)}^2$ (suppressing the iteration index i), a general result for a product of normal distributions.

- *Termination:* At the end of each iteration, we checked whether the sampling of the posterior is successful and stable: specifically, we check the two criteria Equation. (27) and Equation. (28), and terminate if they are simultaneously satisfied twice in a row. We assume that the estimate of σ would also have converged if the resulting posterior sample is good.

  Even if the chain was not completely stabilized, we stopped after a finite number of iterations. In this work, we tried up to $i_{\max} = 10$ iterations.

After the termination of the iterative process, we perform a final sampling with a longer chain, using $N = 20000$. This final sample of the posterior is then used for further analysis.

References

Barnard J, McCulloch R, and Meng XL. Modelling covariance matrices in terms of standard deviations and correlations with applications to shrinkage. *Statistica Sinica*, 10(4):1281–1311, 2000.

Gelman A, Roberts GO, Gilks WR, and others . Efficient Metropolis jumping rules. *Bayesian statistics*, 5 (599-608):42, 1996.

George E and Foster DP. Calibration and empirical Bayes variable selection. *Biometrika*, 87(4):731–747, 2000.

Haario H, Saksman E, and Tamminen J. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001.

Kastner CA, Braumann A, Man PL, Mosbach S, Brownbridge GP, Akroyd J, Kraft M, and Himawan C. Bayesian parameter estimation for a jet-milling model using Metropolis-Hastings and Wang-Landau sampling. *Chemical Engineering Science*, 89:244 – 257, 2013.

McKay MD, Beckman RJ, and Conover WJ. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

Metropolis N, Rosenbluth A, Rosenbluth M, and Teller A. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(1087), 1953.

Mosbach S, Braumann A, Man PL, Kastner CA, Brownbridge GP, and Kraft M. Iterative improvement of Bayesian parameter estimates for an engine model by means of experimental design. *Combustion and Flame*, 159(3):1303 – 1313, 2012.

Na J, Park S, Bak JH, Kim M, Lee D, Yoo Y, Kim I, Park J, Lee U, and Lee JM. Bayesian inference of aqueous mineral carbonation kinetics for carbon capture and utilization. *Industrial & Engineering Chemistry Research*, 58(19):8246–8259, 2019.

Robbins H. An empirical Bayes approach to statistics. In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, pages 157–163. University of California Press, 1956.

Roberts GO and Rosenthal JS. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16(4):351–367, 2001.

Roberts GO, Gelman A, and Gilks WR. Weak convergence and optimal scaling of random walk Metropolis algorithms. *Annals of Applied Probability*, 7(1):110–120, 1997.

Rosenthal JS. Optimal proposal distributions and adaptive MCMC. In *Handbook of Markov Chain Monte Carlo*, number 1, pages 93–112. Chapman and Hall CRC, 2011.

Rybicki GB and Press WH. Class of fast methods for processing irregularly sampled or otherwise inhomogeneous one-dimensional data. *Phys. Rev. Lett.*, 74:1060–1063, 1995.