Bayesian estimation for stochastic gene expression using multifidelity models

Huy D. Vo,[*,†] Zachary Fox,[‡] Ania Baetica,[¶] and Brian Munsky[*,†,‡]

†*Department of Chemical and Biological Engineering, Colorado State University, Fort Collins, CO*

‡*Keck Scholars, School of Biomedical Engineering, Colorado State University, Fort Collins, CO*

¶*Department of Biochemistry and Biophysics, University of California San Francisco, San Francisco, CA*

E-mail: Huy.Vo@colostate.edu; Munsky@colostate.edu

Abstract

The finite state projection (FSP) approach to solving the chemical master equation (CME) has enabled successful inference of discrete stochastic models to predict single-cell gene regulation dynamics. Unfortunately, the FSP approach is highly computationally intensive for all but the simplest models, an issue that is highly problematic when parameter inference and uncertainty quantification takes enormous numbers of parameter evaluations. To address this issue, we propose two new computational methods for the Bayesian inference of stochastic gene expression parameters given single-cell experiments. First, we present an adaptive scheme to improve parameter proposals for Metropolis-Hastings sampling using full FSP-based likelihood evaluations. We then formulate and verify an Adaptive Delayed Acceptance Metropolis-Hastings (ADAMH) algorithm to utilize with reduced Krylov-basis projections of the FSP. We test and

compare both algorithms on three example models and simulated data to show that the ADAMH scheme achieves substantial speedup in comparison to the full FSP approach. By reducing the computational costs of parameter estimation, we expect the ADAMH approach to enable efficient data-driven estimation for more complex gene regulation models.

Introduction

An important goal of quantitative biology is to elucidate and predict the mechanisms of gene expression. Evidence increasingly suggests that gene expression processes are inherently stochastic with substantial cell-to-cell variability.[1–3] In an isogenic population with the same environmental factors, much of these fluctuations can be attributed to intrinsic chemical noise, which is captured well by the chemical master equation (CME).[4] Predictive models for gene expression dynamics can be identified by fitting the solution of the CME to the empirical histogram of single-cell data at several experimental conditions or time-points.[5–8]

The finite state projection (FSP),[9] which approximates the dynamics of the CME with a finite system of linear ODEs, provides a framework to analyze full distributions of stochastic gene expression models with computable error bounds. It has been observed that the full distribution-based analyses using the FSP perform well, even when applied to realistically small experimental datasets on which summary statistics-based fits may fail.[10] On the other hand, the FSP requires solving a large system of ODEs that grows quickly with the complexity of the gene expression network under consideration. Our present study borrows from model reduction strategies in other complex systems fields to alleviate this issue by reducing the computational cost of FSP-based parameter estimation.

There has been intensive research on efficient computational algorithms to quantify the uncertainty in complex models.[11] A particularly promising approach is to utilize multifidelity algorithms to systematically approximate the original system response. In these approximations, surrogate models or meta-models allow for various degrees of model fidelity (e.g., error

compared to the exact model) in exchange for reductions in computational cost. Surrogate models generally fall into two categories: response surface and low-fidelity models.[12,13] We will focus on the second category that consists of reduced-order systems, which approximate the original high-dimensional dynamical system using either simplified physics or projections onto reduced order subspaces.[11,14,15] Reduced-order modeling has already begun to appear in the context of stochastic gene expression. When all model parameters are known, the CME can be reduced by system-theoretic methods,[16,17] sparse-grid/aggregation strategies,[18,19] tensor train representations[20–22] and hierarchical tensor formats.[23] Model reduction techniques have also been applied to parameter optimization by Waldherr and Hassdonk[24] who projected the CME onto a linear subspace spanned by a reduced basis, and Liao et al.[25] who approximated the CME with a Fokker-Planck equation that was projected onto the manifold of low-rank tensors.[26] While these previous works clearly show the promise of reduced-order modeling, there remains a vast reservoir of ideas from the broader computational science and engineering community that remain to be adapted to the quantitative analysis of stochastic gene expression.

In this paper, we introduce two efficient algorithms, which are based on the templates of the adaptive Metropolis algorithm[27] and the delayed acceptance Metropolis-Hastings (DAMH[28,29]) algorithm, to sample the posterior distribution of gene expression parameters given single-cell data. The adaptive Metropolis approach automatically tunes parameter proposal distributions to more efficiently search spaces of unnormalized and correlated parameters. The DAMH provides a two-stage sampling approach that uses a cheap approximation to the posterior distribution at the first stage to quickly filter out unlikely parameters. Improvements to the DAMH allow algorithmic parameters to be updated adaptively and automatically by the DAMH chain.[30,31] The DAMH has been applied to the inference of stochastic chemical kinetics parameters from time-course data.[32] Our algorithm is a modified version of DAMH that is specifically adapted to improve Bayesian inference from population snapshots of single-cell data, such as data arising from flow cytometry or fixed-cell

microscopy experiments. We employ parametric reduced order models using Krylov-based projections,[33,34] which give an intuitive means to compute expensive FSP-based likelihood evaluations.[35,36] To improve the accuracy and the DAMH acceptance rate, we allow the reduced model to be refined during parameter space exploration. The resulting method, which we call the ADAMH-FSP-Krylov algorithm, is tested on three common gene expression models. We also provide a theoretical guarantee and numerical demonstrations that the proposed algorithms converge to equivalent target posterior distributions.

The organization of the paper is as follows. We review the background on the FSP analysis of single-cell data, and basic Markov chain Monte Carlo (MCMC) schemes in the *Background* section. In the *Materials and Methods* section, we introduce our method to generate reduced FSP models, as well as our way of monitoring and refining their accuracy. These reduced models give rise to an approximation to the true likelihood function, which is then employed to devise an Adaptive Delayed Acceptance Metropolis-Hastings with FSP-Krylov reduced models (ADAMH-FSP-Krylov). We make simple adjustments to the existing ADAMH variants in the literature to prove convergence, and we give the mathematical details in the supplementary materials. We provide empirical validation of our methods on three gene expression models, and we compare the efficiency and accuracy of the approaches in the *Numerical Results* section. Interestingly, we find empirically that the reduced model learned through the ADAMH run could fully substitute the original FSP model in a Metropolis-Hastings run without incurring a large difference in the sampling results. Finally, we conclude with a discussion of future work and the potential of computational science and engineering tools to analyze stochastic gene expression.

Background

Stochastic modeling of gene expression and the chemical master equation

Consider a well-mixed biochemical system with $N \geq 1$ different chemical species that are interacting via $M \geq 1$ chemical reactions. Assuming constant temperature and volume, the time-evolution of this system can be modeled by a continuous-time Markov process.[4] The state space of the Markov process consists of integral vectors $\boldsymbol{x} \equiv (x_1, \ldots, x_N)^T$, where x_i is the population of the ith species. Each reaction channel, such as the transcription of an RNA species, is characterized by a *stoichiometric* vector $\boldsymbol{\nu}_j$ $(j = 1, \ldots, M)$ that represents the change when the reaction occurs; if the system is in state $\boldsymbol{x}$ and reaction j occurs, then the system transitions to state $\boldsymbol{x} + \boldsymbol{\nu}_j$. Given $\boldsymbol{x}(t) = \boldsymbol{x}$, the propensity $\alpha_j(\boldsymbol{x}; \boldsymbol{\theta})dt$ determines the probability that reaction j occurs in the next infinitesimal time interval $[t, t + dt)$, where $\boldsymbol{\theta}$ is the vector of model parameters.

Since the state space is discrete, we can index the states as $\boldsymbol{x}_1, ..., \boldsymbol{x}_n, \ldots$. The time-evolution of the probability distribution of the Markov process is the solution of the linear system of differential equations known as the chemical master equation (CME):

$$\begin{cases} \frac{d}{dt}\boldsymbol{p}(t) = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{p}(t), \quad t \in [0,\, t_f] \\ \boldsymbol{p}(0) = \boldsymbol{p}_0 \end{cases}, \tag{1}$$

where the probability mass vector $\boldsymbol{p} = (p_1, p_2, \ldots)^T$ is such that each component, $p_i = P(t, \boldsymbol{x}_i) = \text{Prob}\{\boldsymbol{x}(t) = \boldsymbol{x}_i\}$, describes the probability of being at state $\boldsymbol{x}_i$ at time t, for $i = 1, \ldots, n$. The vector $\boldsymbol{p}_0 = \boldsymbol{p}(0)$ is an initial probability distribution and $\boldsymbol{A}(\boldsymbol{\theta})$ is the infinitesimal generator of the Markov process. Here, we have made explicit the dependence of $\boldsymbol{A}$ on the model parameter vector $\boldsymbol{\theta}$, which is often inferred from experimental data.

Finite State Projection

The state space of the CME could be infinite or extremely large. To alleviate this problem, the finite state projection (FSP[9]) was introduced to truncate the state space to a finite size. In the simplest FSP formulation, the state space is restricted to a hyper-rectangle

$$\boldsymbol{H} = \{0, \ldots, n_1\} \times \cdots \times \{0, \ldots, n_N\}, \tag{2}$$

where the n_k are the maximum copy numbers of the chemical species.

The infinite-dimensional matrix $\boldsymbol{A}$ and vector $\boldsymbol{p}$ in eq. (1) are replaced by the corresponding submatrix and subvector. When the bounds n_k are chosen sufficiently large and the propensities satisfy some regularity conditions, the gap between the FSP and the original CME is negligible and computable.[9,37] Throughout this paper, we assume that the bounds n_k have been chosen appropriately and that the FSP serves as a high-fidelity model of the gene expression dynamics of interest. Our goal is to construct lower-fidelity models of the FSP using model order reduction and incorporate these reduced models in the uncertainty analysis for gene expression parameters.

Bayesian inference from single-cell data

Data from smFISH experiments[5,8,38,39] consist of several snapshots of many independent cells taken at discrete times $t_1, \ldots, t_T$. The snapshot at time t_i records gene expression in n_i cells, each of which can be collected in the data vector $\boldsymbol{c}_{j,i},\ j = 1, \ldots, n_i$ of molecular populations in cell j at time t_i. Let $\boldsymbol{p}(t, \boldsymbol{x}|\boldsymbol{\theta})$ denote the entry of the FSP solution corresponding to state $\boldsymbol{x}$ at time t, with model parameters $\boldsymbol{\theta}$. The FSP-based approximation to the log-likelihood of the data set $\mathcal{D}$ given parameter vector $\boldsymbol{\theta}$ is given by

$$L(\mathcal{D}|\boldsymbol{\theta}) = \sum_{i=1}^{T} \sum_{j=1}^{n_i} \log p(t_i, \boldsymbol{c}_{j,i}|\boldsymbol{\theta}). \tag{3}$$

It is clear that when the FSP solution converges to the true solution of the CME, the FSP-based log-likelihood converges to the true data likelihood. The posterior distribution of model parameters $\boldsymbol{\theta}$ given the data set $\mathcal{D}$ then takes the form

$$f_{posterior}(\boldsymbol{\theta}|\mathcal{D}) \propto \exp(L(\mathcal{D}|\boldsymbol{\theta}))f_0(\boldsymbol{\theta}),$$

where f_0 is the prior density that quantifies prior knowledge and beliefs about the parameters. When f_0 is a constant, the parameters that maximize the posterior density are equivalent to the maximum likelihood estimator. However, we also want to quantify our uncertainty regarding the accuracy of the parameter fit, and the MCMC framework provides a way to address this by sampling from the posterior distribution.

For convenience, we limit our current discussion to models and inference problems that have the following characteristics:

1. The matrix $\boldsymbol{A}(\boldsymbol{\theta})$ can be decomposed into

$$\boldsymbol{A}(\boldsymbol{\theta}) = \sum_{j=1}^{M} g_j(\boldsymbol{\theta})\boldsymbol{A}_j, \tag{4}$$

   where g_j are continuous functions and $\boldsymbol{A}_j$ are independent of the parameters.

2. The support of the prior is contained in a bounded domain of the form

$$\Theta = [\theta_1^{\min}, \theta_1^{\max}] \times \ldots \times [\theta_d^{\min}, \theta_d^{\max}]. \tag{5}$$

The first assumption means that the CME matrix depends "linearly" on the parameters, ensuring the efficient assembly of the parameter-dependent matrix. In particular, the factors $\boldsymbol{A}_j$ can be computed and stored in the offline phase before parameter exploration and only a few (sparse) matrix additions are required to compute $\boldsymbol{A}(\boldsymbol{\theta})$ in the online phase. When there are nonlinear dependence on parameters, more sophisticated methods such as the Discrete

Empirical Interpolation method[40] could be applied, but we leave this development for future work in order to focus more on the parameter sampling aspect. Nevertheless, condition (4) covers an important class of models, including all models defined by mass-action kinetics. The second assumption means that the support of the posterior distribution is a bounded and well-behaved domain (in mathematical terms, a compact set). This allows us to derive convergence theorems more straightforwardly. In practice, condition (5) is not a severe restriction since it can be interpreted as the prior belief that physical parameters cannot assume infinite values.

The Metropolis-Hastings and the adaptive Metropolis algorithms

The Metropolis-Hastings (MH) Algorithm[41,42] is one of the most popular methods to sample from a multivariate probability distribution (Algorithm 1). The basic idea of the MH is to generate a Markov chain whose limiting distribution is the target distribution. To do so, the algorithm includes a probabilistic acceptance/rejection step. More precisely, let f denote the target probability density. Assume the chain is at state $\boldsymbol{\theta}_i$ at step i. Let $\boldsymbol{\theta}'$ be a proposal from the pre-specified proposal density $q(.|\boldsymbol{\theta}_i)$. The DAMH computes a first-step acceptance probability of the form

$$\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}') = \min\left(1, \frac{f(\boldsymbol{\theta}')}{f(\boldsymbol{\theta}_i)} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)}\right),$$

to decide whether to accept $\boldsymbol{\theta}'$ as the next state of the chain. If $\boldsymbol{\theta}'$ fails to be promoted, the algorithm moves on to the next iteration with $\boldsymbol{\theta}_{i+1} := \boldsymbol{\theta}_i$.

There could be many choices for the proposal density q (for example, see the survey of Roberts and Rosenthal[43]). We will consider only the symmetric case where q is a Gaussian, that is,

$$q(\boldsymbol{\theta}'|\boldsymbol{\theta}) \propto \exp\left(-\frac{1}{2}(\boldsymbol{\theta}' - \boldsymbol{\theta})^T \boldsymbol{\Sigma} (\boldsymbol{\theta}' - \boldsymbol{\theta})\right),$$

where $\boldsymbol{\Sigma}$ is a positive definite matrix that determines the covariance of the proposal distribution. With this choice, the MH reduces to the original Metropolis Algorithm.[41] For gene

expression models, the MH has been combined with the FSP for parameter inference and model selection in several studies.[8,10]

The appropriate choice of $\boldsymbol{\Sigma}$ is crucial for the performance of the Metropolis algorithm. Haario et al.[27] proposes an Adaptive Metropolis (AM) algorithm in which the proposal $\boldsymbol{\Sigma}$ is updated at every step using the values visited by the chain. This is the version that we will implement for sampling the posterior distribution with the full FSP model. In particular, let $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_i$ be the samples accepted so far, the AM updates the proposal covariance using the formula

$$\boldsymbol{\Sigma} = \boldsymbol{\Sigma}_i := \begin{cases} \boldsymbol{\Sigma}_0, \ i < n_0 \\ s_d \operatorname{Cov}(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_i) + 10^{-6} s_d \boldsymbol{I}_d, \ i \geq n_0 \end{cases} .$$

Here, the function Cov returns the sample covariances. The constant s_d is assigned the value $(2.4)^2/d$ following Haario et al.[27] The matrix $\boldsymbol{\Sigma}_0$ is an initial choice for the Gaussian proposal density, and n_0 is the number of initial steps without proposal adaptations. Using the adaptive Metropolis allows for more efficient search over un-normalized and correlated parameters spaces and eliminates the need for the user to manually tune the algorithmic parameters. In the numerical results that we will show, the adaptive Metropolis results in reasonable acceptance rates ($19\% - 23.4\%$). Although non-adaptive MH algorithms have been consider in the past,[8,10] to the best of our knowledge, this is the first adaptive MH algorithm to be proposed for Bayesian inference of gene expression models.

Materials and Methods

Delayed acceptance Metropolis-Hastings algorithm

Previous applications of the MH to gene expression have required 10^4 to 10^6 or more iterations per combination of model and data set,[10] and computational cost is a significant issue when sampling from a high-dimensional distribution whose density is expensive to evaluate. A practical rule of thumb for balancing between exploration and exploitation for

Algorithm 1 Metropolis-Hastings

Input:
Target density $f(.)$;
Initial parameter $\boldsymbol{\theta}_0$;
Proposal density $q(\cdot|\cdot)$;

1: **for** $i = 0, 1, \ldots,$ **do**
2: Draw $\boldsymbol{\theta}'$ from the proposal density $q(\cdot|\boldsymbol{\theta}_i)$
3: Compute the acceptance probability

$$\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}') = \min\left(1, \frac{f(\boldsymbol{\theta}')}{f(\boldsymbol{\theta}_i)} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)}\right)$$

4: With probability $\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, set $\boldsymbol{\theta}_{i+1} \leftarrow \boldsymbol{\theta}'$ (accept); otherwise $\boldsymbol{\theta}_{i+1} \leftarrow \boldsymbol{\theta}_i$ (reject).
5: **end for**

Output: samples $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$

a MH algorithm with the Gaussian proposal is to have an acceptance rate close to 0.234, which was derived by Roberts et al.[44] as the asymptotically optimal acceptance rate for random walk MH algorithms. Assuming the proposal density of Algorithm 1 is tuned to have an acceptance rate of approximately 23.4%, one could achieve significant improvement to computation time if one can quickly screen out the remaining rejected proposals without evaluating the expensive posterior density.

The delayed acceptance Metropolis-Hasting (DAMH)[28] seeks to alleviate the computational burden of rejections in the original MH by employing a rejection step based on a cheap approximation to the target density (cf. Algorithm 2). Specifically, let $f(.)$ be the density of the target distribution of the parameter $\boldsymbol{\theta}$. Let $f^{\star}_{\boldsymbol{\theta}}(.)$ be a cheap state-dependent approximation to f. At iteration i, let $\boldsymbol{\theta}'$ be a proposal from the current parameter $\boldsymbol{\theta}$ using a pre-specified proposal density $q(.|.)$. The DAMH promotes $\boldsymbol{\theta}'$ as a potential candidate for acceptance with probability

$$\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}') = \min\left(1, \frac{f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{\theta}')}{f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{\theta})} \frac{q(\boldsymbol{\theta}|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta})}\right).$$

If $\boldsymbol{\theta}'$ fails to be promoted, the algorithm moves on to the next iteration with $\boldsymbol{\theta}_{i+1} := \boldsymbol{\theta}_i$. If the $\boldsymbol{\theta}'$ passes the first inexpensive check, than a second acceptance probability is computed

using the formula

$$\beta(\boldsymbol{\theta}, \boldsymbol{\theta}') = \min\left(1, \frac{f(\boldsymbol{\theta}')}{f(\boldsymbol{\theta})} \frac{f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{f^{\star}_{\boldsymbol{\theta}}(\boldsymbol{\theta}')}\right),$$

and the DAMH algorithm accepts $\boldsymbol{\theta}'$ for the next step with probability β. In this manner, much computational savings can be expected if unlikely proposals are quickly rejected in the first step, leaving only the most promising candidates for careful evaluation in the second step. Christen and Fox show that the ADAMH converges to the target distribution under conditions that are easily met in practice.[28] However, the quality of the approximation $f^{\star}_{\boldsymbol{\theta}}$ affects the overall efficiency. Poor approximations lead to many false promotions of parameters that are rejected at the expensive second step. On the other hand, the first step may falsely reject parameters that could have been accepted using the accurate log-likelihood evaluation. This leads to subsequent developments that seek appropriate approximations and ways to adapt these approximations to improve the performance of DAMH in specific applications.[30,45] Specifically, the adaptive DAMH variant in Cui et al., 2014[45] formulates $f^{\star}_{\boldsymbol{\theta}}$ via reduced basis models that can be updated on the fly using samples accepted by the chain. The adaptive version in Cui et al., 2011,[30] allows adaptations for the proposal density and the error model, with convergence guarantees.[31] We will borrow these elements in our sampling scheme that we introduce below. However, the stochastic gene expression models that we investigate here differ from the models studied in those previous contexts, since our likelihood function incorporates intrinsic discrete state variability instead of external Gaussian noise.

Reduced-order models for the FSP dynamics

Projection-based model reduction

We approximate the full parameter-dependent FSP dynamics,

$$\frac{d}{dt}\boldsymbol{p}(t;\boldsymbol{\theta}) = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{p}(t;\boldsymbol{\theta}),\ \boldsymbol{p}(0) = \boldsymbol{p}_0,\ t \in [0, t_f], \tag{6}$$

Algorithm 2 Delayed Acceptance Metropolis-Hastings

Input:

Target density $f(.)$;
State-dependent density *approximation* $f^{\star}_{\boldsymbol{\theta}}(.)$;
Initial parameter $\boldsymbol{\theta}_0$;
Proposal density $q(\cdot|\cdot)$;

1: **for** $i = 0, 1, \ldots,$ **do**
2: Draw $\boldsymbol{\theta}'$ from the proposal density $q(\cdot|\boldsymbol{\theta}_i)$
3: Compute the first-stage acceptance probability

$$\rho(\boldsymbol{\theta}_i, \boldsymbol{\theta}') = \min\left(1, \frac{f^{\star}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta}')}{f^{\star}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta})} \frac{q(\boldsymbol{\theta}_i|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)}\right)$$

4: With probability $\rho(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, promote the value of $\boldsymbol{\theta}'$ to the next stage. Otherwise, set $\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta}$.
5: Compute the second-stage acceptance probability

$$\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}') = \min\left(1, \frac{f(\boldsymbol{\theta}')}{f(\boldsymbol{\theta}_i)} \frac{f^{\star}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta}_i)}{f^{\star}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta}')}\right)$$

6: With probability $\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, set $\boldsymbol{\theta}_{i+1} \leftarrow \boldsymbol{\theta}'$ (accept); otherwise $\boldsymbol{\theta}_{i+1} \leftarrow \boldsymbol{\theta}_i$ (reject).
7: **end for**

Output: samples $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$

with a sequence of reduced-order dynamics,

$$\frac{d}{dt}\boldsymbol{q}^{(i)}(t;\boldsymbol{\theta}) = \boldsymbol{B}^{(i)}(\boldsymbol{\theta})\boldsymbol{q}^{(i)}(t;\boldsymbol{\theta}), \tag{7}$$

$$\boldsymbol{q}^{(i)}(t_{i-1};\boldsymbol{\theta}) = \left(\boldsymbol{\Phi}^{(i)}\right)^T \boldsymbol{\Phi}^{(i-1)}\boldsymbol{q}^{(i-1)}(t_{i-1};\boldsymbol{\theta}). \tag{8}$$

Here, $i = 1, \ldots, n_B$ indexes the user-specified subintervals $[t_{i-1}, t_i]$ with $t_0 = 0$. Each matrix $\boldsymbol{\Phi}^{(i)} \in \mathbb{R}^{n\times r_i}$, $r_i \leq n$, has orthonormal columns that span the subspace onto which we project the full dynamics. Equation (8) implies that the solution at a previous time interval will be projected onto the subspace of the next interval. While this introduces some extra errors, subdividing the long time interval helps to reduce the subspace dimensions for systems with complicated dynamics. Given an ordered set of reduced bases $\boldsymbol{\Phi} = \left(\boldsymbol{\Phi}^{(i)}\right)_{i=1}^{n_B}$, the approximations to the full distributions are given by

$$\boldsymbol{p}(t) \approx \boldsymbol{p}_{\boldsymbol{\Phi}}(t) = \boldsymbol{\Phi}^{(i)}\boldsymbol{q}^{(i)}(t), \quad t \in [t_{i-1}, t_i]. \tag{9}$$

Under assumption (4), the reduced system matrices $\boldsymbol{B}^{(i)}(\boldsymbol{\theta})$ in eq. (7) can be decomposed as

$$\boldsymbol{B}^{(i)}(\boldsymbol{\theta}) = \sum_{j=1}^{M} g_j(\boldsymbol{\theta})\boldsymbol{B}_j^{(i)}, \tag{10}$$

where $\boldsymbol{B}_j^{(i)} = (\boldsymbol{\Phi}^{(i)})^T \boldsymbol{A}_j \boldsymbol{\Phi}^{(i)}$. This decomposition allows us to assemble the reduced systems quickly with $O(r_i^2)$ complexity.

We build the reduced basis for the parameter-dependent dynamics by concatenation (see, e.g., Benner et al.[15]). Specifically, we assume that for any fixed parameter $\boldsymbol{\theta}$, we can construct a set $\{\boldsymbol{V}^{(i)}(\boldsymbol{\theta})\}_{i=1}^{n_B}$ of orthogonal basis matrices. We can sample different bases from a finite

set of 'training' parameters $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{n_{\text{train}}}$. Then, through the iterative updates

$$\boldsymbol{\Phi}^{(i,1)} = \boldsymbol{V}^{(i)}(\boldsymbol{\theta}_1), \tag{11}$$

$$\boldsymbol{\Phi}^{(i,j)} = \text{Gram-Schmidt}\left[\boldsymbol{\Phi}^{(i,j-1)} \quad \boldsymbol{V}^{(i)}(\boldsymbol{\theta}_j)\right], \tag{12}$$

we obtain the bases $\boldsymbol{\Phi}^{(i)} = \boldsymbol{\Phi}^{(i,n_{\text{train}})}$ that provide global approximations for the full dynamical system across the parameter domain. The operation Gram-Schmidt implies that the columns in $\boldsymbol{V}^{(i)}(\boldsymbol{\theta}_j)$ are orthogonalized against the columns in $\boldsymbol{\Phi}^{(i,j-1)}$ to produce a new matrix with orthonormal columns.

Krylov subspace approximation for single-parameter model reduction

Consider a fixed parameter combination $\boldsymbol{\theta}$. Let the time points $0 < t_1 < \ldots < t_B = t_f$ be given. Using a high-fidelity solver, we can compute the full solution at those time points, and we let $\boldsymbol{p}_i$ denote the full solution at time t_i. Our aim is to construct a sequence of orthogonal matrices $\boldsymbol{V}^{(i)} \equiv \boldsymbol{V}^{(i)}(\boldsymbol{\theta})$ with $i = 1 \ldots B$ such that the full model dynamics at the parameter $\boldsymbol{\theta}$ on the time interval $[t_{i-1}, t_i]$ is well-approximated by a projected reduced model on the span of $\boldsymbol{V}^{(i)}$.

A simple and effective way to construct the reduced bases is to choose $\boldsymbol{V}^{(i)}$ as the orthogonal basis of the Krylov subspace

$$K_{m_i}(\boldsymbol{\theta}, t_{i-1}) = \text{span}\left\{\boldsymbol{p}_i, \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{p}_i, \ldots \boldsymbol{A}(\boldsymbol{\theta})^{m_i-1}\boldsymbol{p}_i\right\}. \tag{13}$$

In order to determine the subspace dimension m_i, we use the error series derived by Saad[33] which we reproduce here using our notation as

$$\exp(\tau_i \boldsymbol{A}(\boldsymbol{\theta}))\boldsymbol{p}_{i-1} - \boldsymbol{V}^{(i)} \exp(\tau_i \boldsymbol{H}^{(i)})\boldsymbol{e}_1 = \sum_{k=1}^{\infty} h_{m_i+1,m_i} \boldsymbol{e}_{m_i}^T \varphi_k(\tau_i \boldsymbol{H}^{(i)})\boldsymbol{e}_1 \tau_i^{k-1} \boldsymbol{A}(\boldsymbol{\theta})^{k-1} \boldsymbol{v}_{m_i+1}^{(i)}. \tag{14}$$

Here, $\boldsymbol{v}_{m_i+1}^{(i)}$ and h_{m_i+1,m_i} are the outputs at step m_i of the Arnoldi procedure (Algorithm

10.5.1 in Golub and Van Loan[46]) to build the orthogonal matrix $\boldsymbol{V}^{(i)}$, where $\varphi_k(\boldsymbol{X}) = \int_0^1 \frac{1}{k!}\exp((1-s)\boldsymbol{X})\,\mathrm{d}s$ for any square matrix $\boldsymbol{X}$. The matrix $\boldsymbol{H}^{(i)} = \left(\boldsymbol{V}^{(i)}\right)^T \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{V}^{(i)}$ is the state matrix of the reduced-order system obtained via projecting $\boldsymbol{A}$ onto the Krylov subspace K_{m_i}. The terms $\boldsymbol{e}_{m_i}^T\varphi_k(\tau_i\boldsymbol{H}^{(i)})\boldsymbol{e}_1$ can be computed efficiently using Expokit (Theorem 1, Sidje[34]). We use the Euclidean norm of the first term of the series (14) as an indicator for the model reduction error. Given an error tolerance $\varepsilon_{\text{Krylov}}$, we iteratively construct the Krylov basis $\boldsymbol{V}^{(i)}$ with increasing dimension until the error per unit time step of the reduced model falls below the tolerance, that is,

$$|h_{m_i+1,m_i}\boldsymbol{e}_{m_i}^T\varphi_1(\tau_i\boldsymbol{H}^{(i)})\boldsymbol{e}_1| \leq \varepsilon_{\text{Krylov}}\tau_i. \tag{15}$$

Adaptive Delayed Acceptance Metropolis with reduced-order models of the CME

The approximate log-likelihood formula

The reduced bases described above allow us to find reduced-cost approximations $\boldsymbol{p} \approx \boldsymbol{p}_{\boldsymbol{\Phi}}$ to the full FSP dynamics. We can then approximate the full log-likelihood of single-cell data in equation (3) by the reduced-model-based log-likelihood

$$L_{\boldsymbol{\Phi}}^{\star}(\mathcal{D}|\boldsymbol{\theta}) = \sum_{i=1}^{T}\sum_{j=1}^{n_i}\log\max(\varepsilon_s, p_{\boldsymbol{\Phi}}(t_i, \boldsymbol{c}_{j,i}|\boldsymbol{\theta})), \tag{16}$$

where ε_s is a small constant, chosen to safeguard against undefined values. We need to include ε_s in our approximation since the entries of the reduced-order approximation are not guaranteed to be positive (not even in exact arithmetic). We aim to make the approximation to be accurate for parameters $\boldsymbol{\theta}$ with high posterior density, and crude on those with low density, which should be visited rarely by the Monte Carlo chain.

One can readily plug in the approximation (16) to the DAMH algorithm. Since $\exp(L_{\boldsymbol{\Phi}}^{\star}(\mathcal{D}|\boldsymbol{\theta})) > 0$ for all $\boldsymbol{\theta} \in \mathbb{R}_+^d$, the chain will eventually converge to the target posterior distribution (The-

orem 1 in Christen and Fox,[28] and Theorem 3.2 in Efendiev et al.[29]). On the other hand, a major problem with the DAMH is that the computational efficiency depends on the quality of the reduced basis approximation. Crude models result in high rejection rates at the second stage, thus increasing sample correlation and computation time. Therefore, it is advantageous to fine-tune the parameters of the algorithm and update the reduced models adaptively to ensure a reasonable acceptance rate. This motivates the adaptive version of the DAMH that we discuss next.

Delayed acceptance posterior sampling with infinite model adaptations

We propose an adaptive version of the DAMH for sampling from the posterior density of the CME parameters given single-cell data (Algorithm 3). We have borrowed elements from the adaptive DAMH algorithms in Cui et al.[30,45] The first step proposal uses an adaptive Gaussian similar to the adaptive Metropolis of Haario et al.,[27] where the covariance matrix is updated at every step from the samples accepted so far. Here, we generate the proposals in $\log_{10}$ space.

The reduced bases are updated as the chain explores the parameter domain. Instead of using a finite adaptation criterion to stop model adaptation as in Cui et al.,[45] we introduce an adaptation probability with which the reduced basis updates are considered. This means that there could be an infinite amount of model adaptations that occur with diminishing probability as the chain progresses. This idea is taken from the "doubly-modified example" in Roberts and Rosenthal.[47] The advantage of the probabilistic adaptation criteria is that it allows us to prove ergodicity for the adaptive algorithm. The mathematical proofs are presented in the Appendix.

The adaptation probability $a(i)$ is chosen to converge to 0 as the chain iteration index i increases. In particular, we use

$$a(i) = 2^{-i/I_0},$$

where I_0 is a user-specified constant. This formula means that the probability for an adapta-

tion to occur decreases by half after every I_0 chain iterations. In addition, we further restrict the adaptation to occur only when the error indicator is above a threshold at the proposed parameters. As a consequence of our model updating criteria, the reduced-order bases will be selected at points that are close to the support of the target posterior distribution.

Algorithm 3 ADAMH-FSP-Krylov

Input:
Prior density f_0;
Parameter-dependent CME matrix $\boldsymbol{A}(.)$;
Chain starting point $\boldsymbol{\theta}_0$, initial proposal covariance $\boldsymbol{C}_0$.
Basis update tolerance ε_{basis}, Krylov tolerance ε_{Krylov}, reduced model time partition $\mathcal{T} = \{t_k\}_{k=1}^{n_B}$;
Adaptation probability $\{a(i)\}_{i=0}^{\infty}$;
Maximum basis dimension m_{max}.

$\boldsymbol{\Phi}_0 = \text{GenerateKrylovBases}(\boldsymbol{A}(\boldsymbol{\theta}_0), \boldsymbol{p}_0, \mathcal{T}, \varepsilon_{Krylov})$;
for $i = 0, 1, \ldots$ **do**
  Compute the proposal $\boldsymbol{\theta}' = 10^{\boldsymbol{\psi}}$ where $\boldsymbol{\psi} \sim N(\log_{10}(\boldsymbol{\theta}_i), \boldsymbol{C}_i)$;
  With probability α, promote $\boldsymbol{\theta}'$, where

$$\alpha = \min\{1, \exp(L^{\star}_{\boldsymbol{\Phi}_i}(D|\boldsymbol{\theta}') + \log f_0(\boldsymbol{\theta}') - L^{\star}_{\boldsymbol{\Phi}_i}(D|\boldsymbol{\theta}_i) - \log f_0(\boldsymbol{\theta}_i))\}$$

  otherwise, $\boldsymbol{\theta}_{i+1} := \boldsymbol{\theta}_i$ and move on to the next iteration.
  if $\boldsymbol{\theta}'$ was promoted **then**
    With probability β, accept $\boldsymbol{\theta}'$ as the next sample $\boldsymbol{\theta}_{i+1}$. Otherwise, set $\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}$. Here,

$$\beta = \min\left\{1, \frac{\exp(L(D|\boldsymbol{\theta}'))}{\exp(L(D|\boldsymbol{\theta}_i))}\frac{\exp(L^{\star}_{\boldsymbol{\Phi}_i}(D|\boldsymbol{\theta}_i))}{\exp(L^{\star}_{\boldsymbol{\Phi}_i}(D|\boldsymbol{\theta}'))}\right\}$$

    Compute $\text{ErrorEst}(\boldsymbol{\theta}', \boldsymbol{\Phi}_i) := |L(D|\boldsymbol{\theta}') - L^{\star}_{\boldsymbol{\Phi}_i}(D|\boldsymbol{\theta}')|/|L(D|\boldsymbol{\theta}')|$
    if $\text{ErrorEst}(\boldsymbol{\theta}', \boldsymbol{\Phi}_i) > \varepsilon_{basis}$ **then**
      With probability $a(i)$, $\boldsymbol{\Phi}_{i+1} = \text{UpdateBases}(\boldsymbol{\Phi}_i, \boldsymbol{\theta}_i)$, otherwise $\boldsymbol{\Phi}_{i+1} := \boldsymbol{\Phi}_i$.
    else
      $\boldsymbol{\Phi}_{i+1} := \boldsymbol{\Phi}_i$.
    end if
  end if
  $\boldsymbol{C}_{i+1} = \text{Cov}(\log_{10}(\boldsymbol{\theta}_0), \ldots, \log_{10}(\boldsymbol{\theta}_{i+1})) + \frac{2.4^2}{d}10^{-6}\boldsymbol{I}_d$; ▷ Update the proposal
end for

Output: Samples $\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots$

Numerical Results

We conduct numerical tests on several stochastic gene expression models to study performance of our proposed Algorithms. The test platform is a desktop computer running Linux Mint and MATLAB 2017a, with 32 GB RAM and Intel Core i7 3.4 GHz quad-core processor.

We compare three sampling algorithms:

1. Adaptive Metropolis-Hastings with full FSP-based likelihood evaluations (AMH-FSP): This version is an adaptation of the Adaptive Metropolis of Haario et al.,[27] which updates the covariance of the Gaussian proposal density at every step. The algorithm always uses the FSP-based likelihood (3) to compute the acceptance probability, and it is solved using the Krylov-based Expokit.[34] This is the reference algorithm by which we assess the accuracy and performance of the other sampling schemes. To the best of our knowledge, such an adaptive Metropolis scheme has not been used elsewhere for gene expression models.

2. Adaptive Delayed Acceptance Metropolis-Hastings with reduced FSP model constructed from Krylov subspace projections (ADAMH-FSP-Krylov): This is Algorithm 3 mentioned above. Similar to AMH-FSP, this algorithm uses a Gaussian proposal with an adaptive covariance matrix. However, it has a first-stage rejection step that employs the reduced model constructed adaptively using Krylov-based projection.

3. Adaptive Metropolis-Hastings with only reduced model-based likelihood evaluations (AMH-ROM): This is similar to AMH-FSP, but we instead use the approximate log-likelihood formula (16). The reduced model is constructed during the run of the ADAMH-FSP-Krylov, and therefore this variant can only be executed after the ADAMH-FSP-Krylov has terminated. We include this variant here in order to study the accuracy and potential speedup when leaving the acceptance/rejection decision fully to the reduced model.

We rely on two metrics for performance evaluation: total CPU time to finish each chain, and the multivariate effective sample size as formulated in Vats et al.[48] Given samples $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n$, the multivariate effective sample size is estimated by

$$\text{mESS} = n \left(\frac{|\Lambda_n|}{|\Sigma_n|} \right)^{1/d},$$

where Λ_n is an estimation of the posterior covariance using the sample covariance, and Σ_n the multivariate batch means estimator. An algorithm, whose posterior distribution matches the full FSP implementation, but with a lower ratio of CPU time per (multivariate) effective sample will be deemed more efficient. We use the MATLAB implementation by Luigi Acerbi[49] for evaluating the effective sample size from the MCMC outputs.

Implementation details

To achieve reproducible results for each example, we reset the random number generator to Mersenne Twister with seed 0 in Matlab using the *rng('default')* command before simulating the single-cell observations with Gillespie's Algorithm[50] and running the ADAMH-Krylov-FSP and AMH-FSP chains. The random seed is then set to the 'default' value again before running the AMH-ROM chain.

Two-state gene expression

We first consider the common model of bursting gene expression[39,51–54] with a gene that can switch between ON and OFF states and an RNA species that is transcribed when the gene is switched on (Table 1). We simulate data at ten equally spaced time points from 0.1 to 1 hour, with 200 independent observations per time point. The gene states are assumed to be unobserved. We generate the reduced bases on subintervals generated by the time points in the set

$$T_{\text{basis}} = \{\Delta t_{\text{data}} j | j = 1, \ldots, 10\} \cup \{\Delta t_{\text{basis}} j | j = 1, \ldots, 100\},$$

Table 1: Two-state gene expression reactions and propensities. We assume that the time unit is hours (hr). Hence, parameters' units are hr^{-1}. ([X] is the number of copies of the species X.)

	reaction	propensity
1.	$G_{OFF} \xrightarrow{k_{on}} G_{ON}$	$\alpha_1 = k_{on}[G_{OFF}]$
2.	$G_{ON} \xrightarrow{k_{off}} G_{OFF}$	$\alpha_2 = k_{off}[G_{ON}]$
3.	$G_{ON} \xrightarrow{k_r} G_{ON} + RNA$	$\alpha_3 = k_r[G_{ON}]$
4.	$RNA \xrightarrow{\gamma} \emptyset$	$\alpha_4 = \gamma[RNA]$

where $\Delta t_{\text{data}} = 0.1\text{hr}$ and $\Delta t_{\text{basis}} = 0.01\text{hr}$. Thus, T_{basis} includes the observation times. We choose the basis update threshold as $\delta = 10^{-4}$. The prior distribution in our test is the log-uniform distribution on a rectangle, whose bounds are given in Table 2. The full FSP state space is chosen as

$$\{\text{OFF}, \text{ON}\} \times \{0, 1, \ldots, 1100\}.$$

We choose a starting point for the sampling algorithms using five iterations of MATLAB's genetic algorithm with a population size of 100, resulting in 600 full FSP evaluations. We then refine the output of the genetic algorithm with a local search using *fmincon* with a maximum of 1000 further evaluations of the full model. This is a negligible cost in comparison to the $10,000$ iterations that we set for the sampling algorithms.

We summarize the performance characteristics of the sampling schemes in Table 3. The ADAMH-FSP-Krylov requires less computational time (Fig. 1) without a significant reduction in the multivariate effective sample size. In terms of computational time, the ADAMH-FSP-Krylov takes less time to generate an independent sample. This is partly explained by observing that the first stage of the scheme filters out many unlikely samples with the efficient approximation, resulting in 78.34% fewer full evaluations in the second stage (cf. Table 3).

We observe from the scatterplot of log-posterior values of the parameters accepted by the ADAMH-FSP-Krylov that the reduced model evaluations are very close to the FSP evaluations, with the majority of the approximate log-posterior values having a relative

error below 10^{-4}, with an average of 1.09×10^{-6} and a median of 8.49×10^{-8} across all 2152 accepted parameter combinations (Fig. 1 C). This accuracy is achieved with a reduced set of no more than 168 basis vectors per time subinterval that was built using solutions from only *four* sampled parameter combinations (Fig. 2). All the basis updates occur during the first tenth portion of the chain, and these updates consume less than one percent of the total chain runtime (Table 4).

From the samples obtained by the ADAMH-Krylov-FSP, we found that full and reduced FSP evaluation take approximately 0.25 and 0.09 seconds on average, allowing for a maximal speedup factor of approximately $100(0.25 - 0.09)/0.25 \approx 65.73\%$ for the current model reduction scheme. Here, the term *reduced model* refers to the final reduced model obtained from the adaptive reduced basis update of the ADAMH-Krylov-FSP. The speedup offered by the ADAMH-Krylov-FSP was found to be $100(2497.70 - 1424.32)/2497.70 \approx 42.97\%$, or approximately two thirds the maximal achievable improvement for the current model reduction scheme. To further investigate the speed and quality of the reduced model learned from the ADAMH-FSP-Krylov run, we performed another run of the adaptive Metropolis-Hastings algorithm with the log-likelihood evaluated solely using the reduced model constructed by the ADAMH-FSP-Krylov. Interestingly, we observe almost identical results using the reduced model alone in comparison to using the full model (Fig. 2 and Table 5), and the 65.03% reduction in computational effort matched very well to the maximal estimated improvement.

Table 2: Two-state gene expression example. Bounds on the support of the prior distribution of the parameters, which we choose to be the uniform distribution. Parameter units are hr^{-1}.

Parameter	k_{on}	k_{off}	k_r	γ
Lower bound	1.00e-06	1.00e-06	1.00e-06	1.00e-06
Upper bound	1.00e+01	1.00e+01	1.00e+04	1.00e+01

A gene expression model with spatial components

We consider an extension of the previous model to distinguish between the nucleus and cytoplasmic compartments in the cell, similar to a stochastic model recently considered for

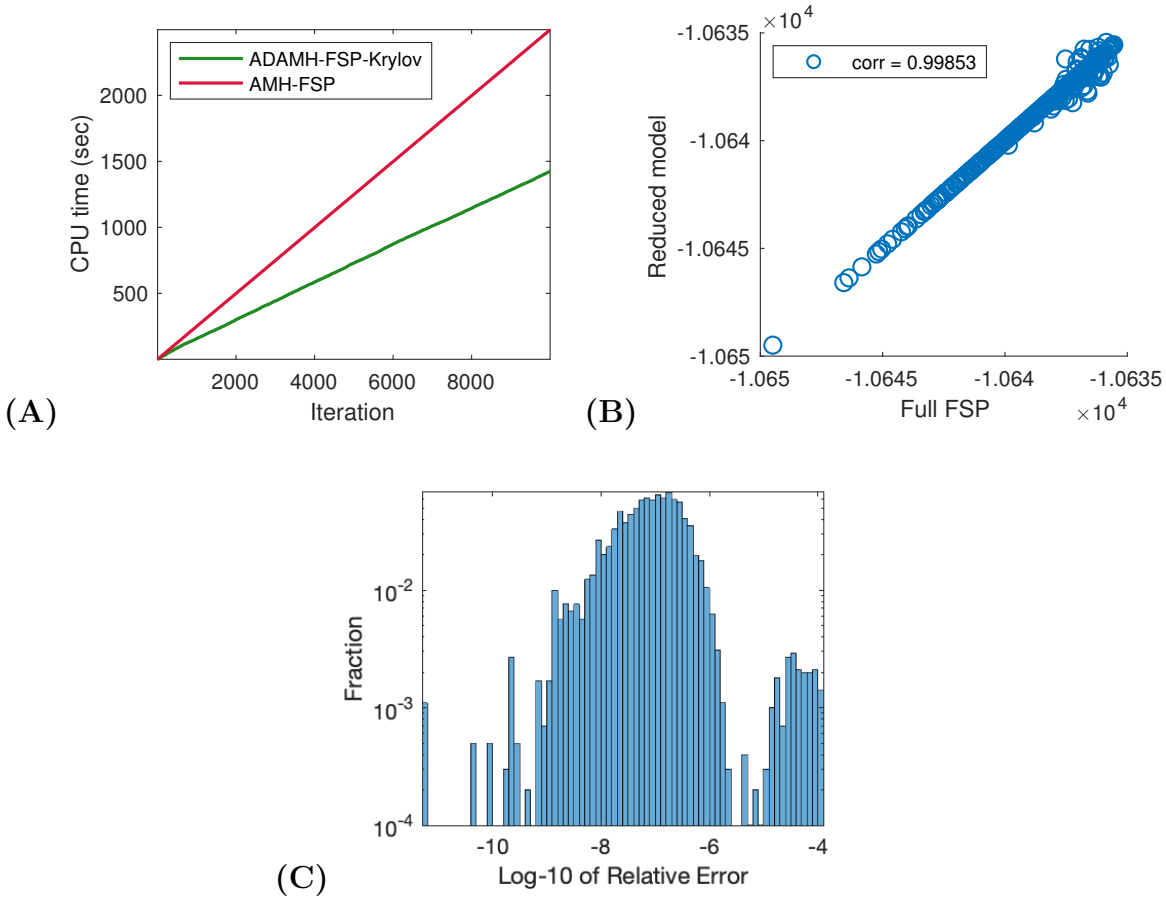

Figure 1: Two-state gene expression example. **(A)** CPU time vs number of iterations for a sample run of the ADAMH-FSP-Krylov and the AMH-FSP. **(B)** Scatterplot of the unnomarlized log-posterior evaluated using the full FSP and the reduced model. Notice that the approximate and true values are almost identical with a correlation coefficient of approximately 0.99853. **(C)** Distribution of the relative error in the approximate log-posterior evaluations at the parameters accepted by the ADAMH chain.

Table 3: Two-state gene expression example. Performance of the Adaptive Delayed Acceptance Metropolis-Hastings with the Krylov-based reduced model (ADAMH-FSP-Krylov), the Adaptive Metropolis-Hastings with full FSP (AMH-FSP), and the Adaptive Metropolis-Hastings with the reduced model constructed by ADAMH-FSP-Krylov (AMH-ROM). The total chain length for each algorithm is 10000.

	mESS	CPU time (sec)	$\frac{\text{CPU time}}{\text{mESS}}$ (sec)	Number of full evaluations	Number of rejections	Number of rejections by full FSP
ADAMH-FSP-Krylov	672.76	1424.32	2.12	2166	7848	14
AMH-FSP	636.76	2497.70	3.92	10000	7859	7859
AMH-ROM	715.11	873.53	1.22	0	7741	0

Table 4: Two-state gene expression example. Breakdown of CPU time spent in the main components of ADAMH-FSP-Krylov.

Component	Time occupied (sec)	Fraction of total time (per cent)
Full FSP Evaluation	545.87	38.33
Reduced Model Evaluation	863.38	60.62
Reduced Model Update	9.55	0.67
Total	1424.32	100.00

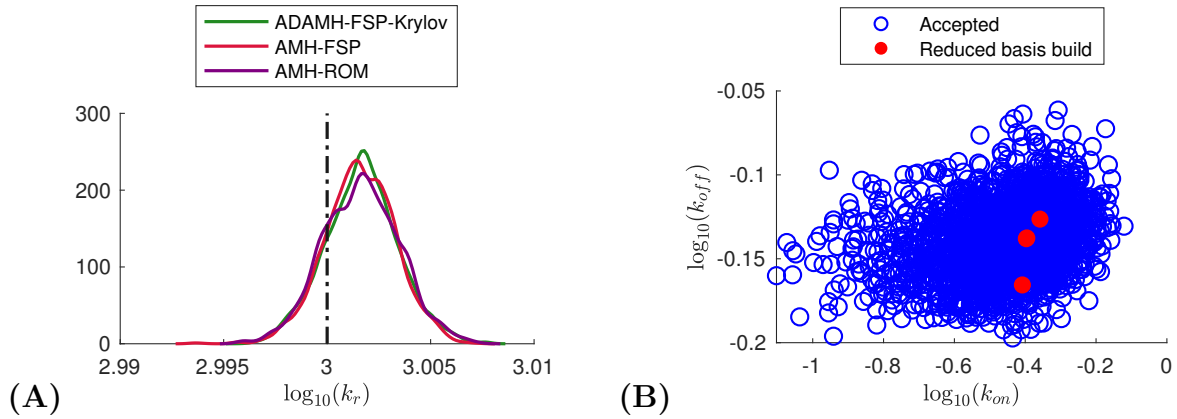

Figure 2: Two-state gene expression example. **(A)** Estimations of the marginal posterior distribution of the parameter k_r using the Adaptive Delayed Acceptance Metropolis-Hastings with Krylov reduced model (ADAMH-FSP-Krylov) and the Adaptive Metropolis-Hastings with full FSP (AMH-FSP). **(B)** 2-D projections of parameter combinations accepted by the ADAMH scheme (blue) and parameter combinations used for reduced model construction (red).

MAPK-activated gene expression dynamics in yeast.[10] The gene can transition between four states $\{0, 1, 2, 3\}$ with transcription activated when the gene state is in states 1 to 3. RNA is transcribed in the nucleus and later transported to the cytoplasm as a first order reaction.

Table 5: Two-state gene expression example. True parameter values and the average values of the parameters visited by the ADAMH-FSP-Krylov and AMH-FSP chains. The "Start" column shows the starting point of both MCMC chains. This starting point is obtained from a numerical optimization procedure that seeks to maximize the full log-likelihood in equation (3). Parameter values are shown in $\log_{10}$ scale. All parameters have unit hr^{-1}.

Parameter	True	Start	ADAMH-FSP-Krylov	AMH-FSP	AMH-ROM
$\log_{10}(k_{\text{on}})$	-3.01e-01	-3.97e-01	-4.49e-01	-4.48e-01	-4.50e-01
$\log_{10}(k_{\text{off}})$	-9.69e-02	-1.38e-01	-1.39e-01	-1.38e-01	-1.40e-01
$\log_{10}(k_r)$	3.00e+00	3.00e+00	3.00e+00	3.00e+00	3.00e+00
$\log_{10}(\gamma)$	0.00e+00	1.03e-03	1.04e-03	9.21e-04	1.01e-03
Log-posterior (un-normalizsed)	-1.0641e+04	-1.0636e+04	-1.0638e+04	-1.0638e+04	-1.0638e+04

These cellular processes and the degradation of RNA in both spatial compartments are modeled by a reaction network with six reactions and three species (Table 6).

We simulated a data set of 200 single-cell measurements at five equally-spaced time points between 1 min and 10 min, that is, $T_{data} = \{2, 4, 6, 8, 10\}$ (min). The time points for generating the basis are $T_{basis} = T_{data} \cup \{j \times 0.2 \text{ min}, j = 1, \ldots, 50\}$. We chose the basis update threshold as $\delta = 10^{-4}$. The prior distribution in our test is the log-uniform distribution on a rectangle, whose bounds are given in Table 7. The full FSP state space is chosen as

$$\{0, 1, 2, 3\} \times \{0, \ldots, 50\} \times \{0, \ldots, 150\},$$

which results in $30,804$ states.

To find the starting point for the chains, we run five generations of MATLAB's genetic algorithm (implemented in the function *ga*) with 600 full FSP evaluations. Then, we run another 500 steps of *fmincon* to refine the output of the *ga* solver. Using the parameter vector output obtained by this combined optimization scheme as the initial sample, we run both the ADAMH-FSP-Krylov and the AMH-FSP for $10,000$ iterations.

The acceleration obtained by using the reduced model is quite evident, with the ADAMH generating an effective sample about twice as fast as the AMH (Table 8). The log-posterior evaluations from the reduced model are accurate (Fig. 3 C and Table 9), with relative

error below the algorithmic tolerance of 10^{-4}, with a mean of 1.11×10^{-5} and a median of 6.98×10^{-6}. This accurate model was built automatically by the ADAMH scheme using just 18 points in the parameter space (Fig. 4), resulting in a set of no more than 438 vectors per time subinterval. All the basis updates occur during the first fifth portion of the chain, and these updates consume about 11.25% of the total runtime (Table 10). The high accuracy of the posterior approximation translates into a very high second-stage acceptance of 96.15% of the proposals promoted by the first-stage reduced-model-based evaluation. Such high acceptance rates in the second stage are crucial to the efficiency for the delayed acceptance scheme, since almost all of the expensive FSP evaluations are accepted.[30]

The close agreement between the first and second stage of the ADAMH algorithm suggests that the reduced model constructed by ADAMH can provide a reliable substitute of the full model. Upon finishing the ADAMH chain, we run another chain with $10,000$ iterations using only the reduced-model-based evaluations, where the reduced model is the final model output from the ADAMH-Krylov-FSP run. We observe that the marginal posterior distributions sampled from this chain are not markedly different from the results of the other two chains (see Fig. (4) for a representative example).

From the posterior samples of the ADAMH chain, we estimate that an average full FSP evaluation would take 1.31 seconds, while an average reduced model evaluation takes 0.30 seconds, leading to an average speedup (in terms of total CPU time) of approximately 77.35%. The comparative runtimes shown in Table 8 confirms this estimate, with the AMH-ROM taking about 76.58% less time than the AMH-FSP chain. The speed up of the ADAMH-Krylov-FSP was comparable at approximately 45.91%.

Genetic toggle switch

The final model we consider in our numerical tests is the nonlinear genetic toggle switch[55] with the propensity functions listed in Table 11. We use the same parameters as those in Fox and Munsky.[56] Using the stochastic simulations and the 'true' parameters as given in

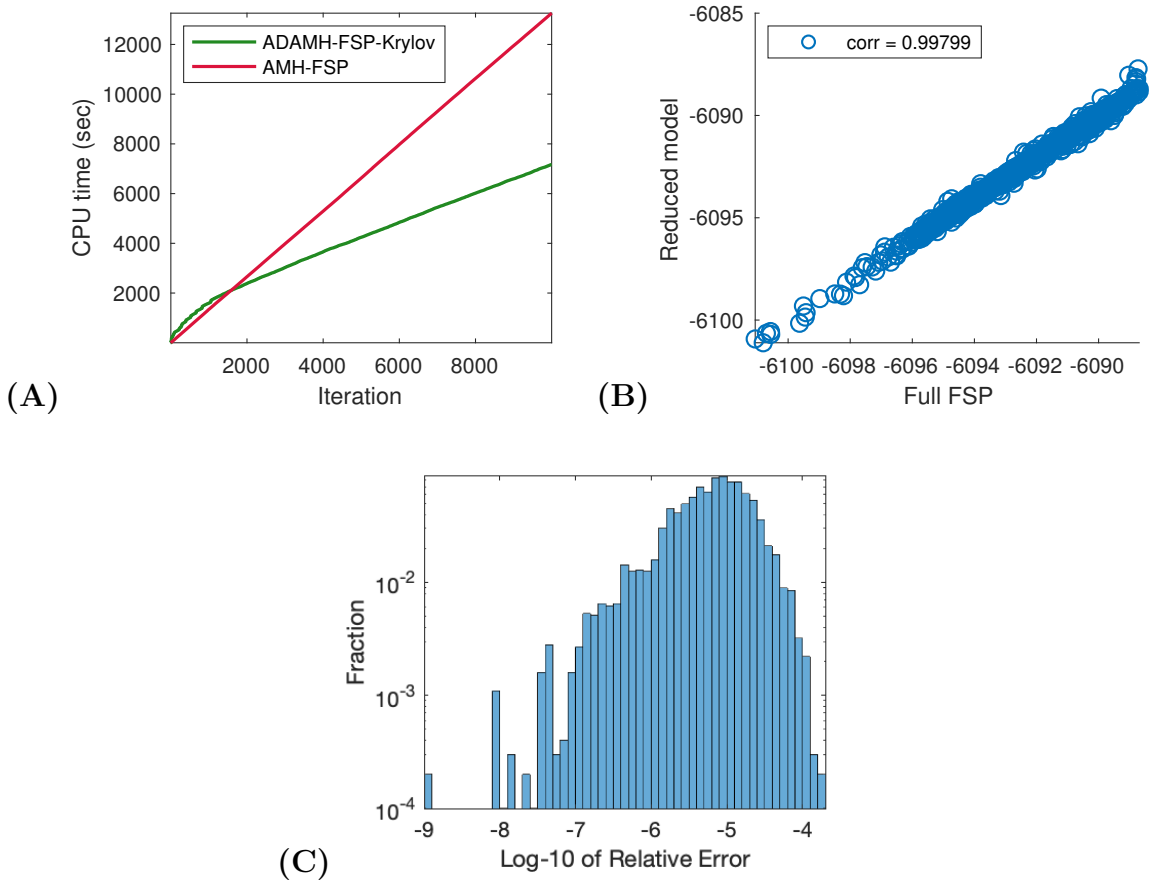

Figure 3: Spatial gene expression. **(A)** CPU time vs number of iterations for a sample run of the ADAMH-FSP-Krylov and the AMH-FSP. **(B)** Scatterplot of the unnomarlized log-posterior evaluated using the full FSP and the reduced model. Notice that the approximate and true values are almost identical with a correlation coefficient of approximately 0.9979. **(C)** Distribution of the relative error in the approximate log-posterior evaluations at the parameters accepted by the ADAMH chain.

Table 6: Spatial gene expression reactions and propensities. The gene is considered as one species with 4 different states G_i, $i = 0, \ldots, 3$. We assume that the time unit is seconds (sec). Hence, the parameters' units are sec^{-1}

	reaction	propensity
1.	$G_i \xrightarrow{k^+_{gene}} G_{i+1}$	$\alpha_1 = k^+_{gene}[i \leq 2]$
2.	$G_i \xrightarrow{k^-_{gene}} G_{i-1}$	$\alpha_2 = k^-_{gene}[i \geq 1]$
3.	$G_i \xrightarrow{k_r} G_i + RNA_{nuc}$	$\alpha_3 = k_r[i \geq 1]$
4.	$RNA_{nuc} \xrightarrow{\gamma_{nuc}} \emptyset$	$\alpha_4 = \gamma_{nuc}[RNA_{nuc}]$
5.	$RNA_{nuc} \xrightarrow{k_{trans}} RNA_{cyt}$	$\alpha_5 = k_{trans}[RNA_{nuc}]$
6.	$RNA_{cyt} \xrightarrow{\gamma_{cyt}} \emptyset$	$\alpha_6 = \gamma_{cyt}[RNA_{cyt}]$

Table 7: Spatial gene expression. Bounds on the support of the prior distribution of the parameters, which we choose to be the log-uniform distribution. All parameters have the same unit sec^{-1}.

Parameter	k^+_{gene}	k^-_{gene}	k_r	γ_{nuc}	k_{trans}	γ_{cyt}
Lower bound	1.00e-06	1.00e-06	1.00e-06	1.00e-08	1.00e-06	1.00e-06
Upper bound	1.00e+00	1.00e+00	1.00e+01	1.00e+00	1.00e+00	1.00e+00

Table 12, we generate data at $2, 6$ and 8 hours, each with 500 single-cell samples. To build the reduced bases for the FSP reduction, we use the union of ten equally-spaced points between zero and 8 hrs and the time points of observations. The prior distribution in our test was chosen as the log-uniform distribution on a rectangle, whose bounds are given in Table 13. The full FSP size is set as the rectangle $\{0, \ldots, 100\} \times \{0, \ldots, 100\}$, corresponding to 10,201 states.

To find the starting point for the chains, we run five generations of MATLAB's genetic algorithm with 600 full FSP evaluations. Then, we run another 1000 iterations of *fmincon* to refine the output of the *ga* solver. Using the parameter vector output by this combined optimization scheme as initial sample, we run both the ADAMH-FSP-Krylov and the AMH-FSP for $100,000$ iterations.

The efficiency of the ADAMH-Krylov-FSP is confirmed in Table 14, where the delayed acceptance scheme is 37.16% faster than the AMH-FSP algorithm, compared to a maximum potential savings of 59.82% when exclusively using the reduced FSP model.

Table 8: Spatial gene expression. Performance of the Adaptive Delayed Acceptance Metropolis-Hastings with Krylov-based reduced model (ADAMH-FSP-Krylov) vs the Adaptive Metropolis-Hastings with full FSP (AMH-FSP). The total chain length for each algorithm is $100,000$. The ADAMH-FSP-Krylov scheme uses markedly fewer full evaluations than the AMH-FSP scheme.

	mESS	CPU time (sec)	$\frac{\text{CPU time}}{\text{mESS}}$ (sec)	Number of full evaluations	Number of rejections	Number of rejections by full FSP
ADAMH-FSP-Krylov	447.99	7171.93	16.01	2546	7552	98
AMH-FSP	401.54	13258.43	33.02	10000	7586	7586
AMH-ROM	329.14	3105.75	9.44	0	7601	0

Table 9: Spatial gene expression. True parameter values and the average values of the parameters visited by the ADAMH-FSP-Krylov and AMH-FSP chains. The "Start" column shows the starting point of both MCMC chains. This starting point is obtained from a numerical optimization procedure that seeks to maximize the full log-likelihood in equation (3). Parameter values are shown in $\log_{10}$ scale. All parameters have unit sec^{-1}.

Parameter	True	Start	ADAMH-FSP-Krylov	AMH-FSP	AMH-ROM
$\log_{10}(k^{+}_{\text{gene}})$	-2.52e+00	-2.55e+00	-2.55e+00	-2.55e+00	-2.56e+00
$\log_{10}(k^{-}_{\text{gene}})$	-2.22e+00	-2.21e+00	-2.21e+00	-2.22e+00	-2.22e+00
$\log_{10}(k_r)$	-5.23e-01	-4.18e-01	-4.18e-01	-4.20e-01	-4.25e-01
$\log_{10}(\gamma_{\text{nuc}})$	-2.52e+00	-1.88e+00	-1.89e+00	-1.91e+00	-1.93e+00
$\log_{10}(k_{\text{trans}})$	-1.52e+00	-1.54e+00	-1.54e+00	-1.54e+00	-1.54e+00
$\log_{10}(\gamma_{\text{cyt}})$	-2.52e+00	-2.56e+00	-2.56e+00	-2.56e+00	-2.56e+00
Log-posterior (un-normalizsed)	-6.0994e+03	-6.0887e+03	-6.0915e+03	-6.0916e+03	-6.0917e+03

Similar to the last two examples, we observe a close agreement between the first and second stage of the ADAMH run, where 98.36% of the proposals promoted by the reduced-model-based evaluations are accepted by the full-FSP-based evaluation. This high second-stage acceptance rate is explained by the quality of the reduced model in approximating the log-posterior values (Fig. 5 C). We also ran another chain using the reduced model outputted by the ADAMH, which yields similar results to the reference chain (Fig. 6) but with reduced computational time (Table 14). The accurate reduced model consists of no more than 634 basis vectors per time subinterval, with all the basis updates occurring during the first tenth portion of the chain.

From the samples obtained by the ADAMH, we found that Expokit takes 0.42 sec to

Table 10: Spatial gene expression example. Breakdown of CPU time spent in the main components of ADAMH-FSP-Krylov.

Component	Time occupied (sec)	Fraction of total time (per cent)
Full FSP Evaluation	3335.65	46.51
Reduced Model Evaluation	3019.64	42.10
Reduced Model Update	807.13	11.25
Total	7171.93	100.00

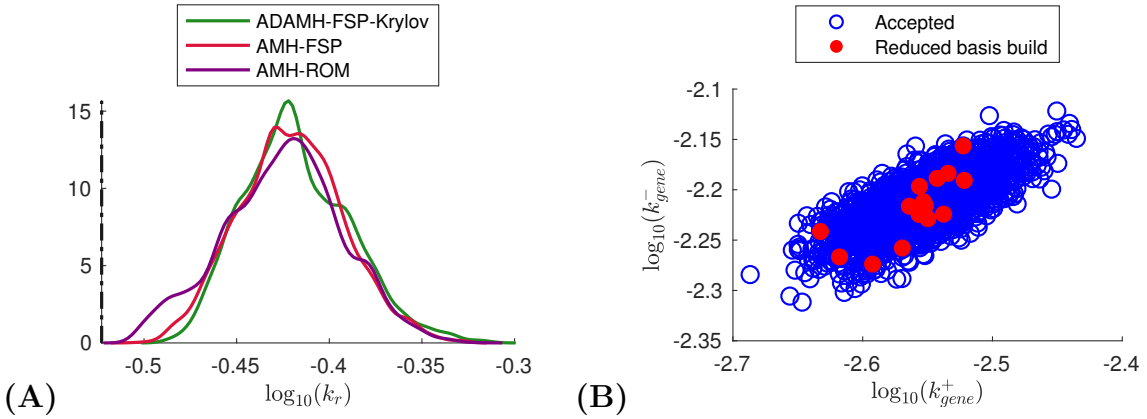

Figure 4: Spatial gene expression. **(A)** Estimations of the marginal posterior distribution of the parameter k_r using the Adaptive Delayed Acceptance Metropolis-Hastings with the Krylov reduced model (ADAMH-FSP-Krylov), the Adaptive Metropolis-Hastings with full FSP (AMH-FSP), and the approximate Adaptive Metropolis-Hastings using the reduced order model learned from the ADAMH-FSP-Krylov run (AMH-ROM). The dashed vertical line marks the true parameter value. **(B)** 2-D projections of parameter combinations accepted by the ADAMH scheme (blue) and parameter combinations used for reduced model construction (red). The truncated appearance of the samples is the consequence of the upper bound on the support of the prior (see Table 7).

Table 11: Reaction channels of the genetic toggle switch example. We assume that the time unit is seconds (sec). Hence, the unit for the parameters $k_{0X}, k_{1X}, \gamma_X, k_{0Y}, k_{1Y}, \gamma_Y$ are sec^{-1}. The other parameters (dimensionless) are fixed at $a_{yx} = 2.6 \times 10^{-3}, a_{xy} = 6.1 \times 10^{-3}, n_{yx} = 3, n_{xy} = 2.1$

	reaction	propensity
1.	$X \longrightarrow \emptyset$	$\gamma_X[X]$
2.	$\emptyset \longrightarrow X$	$k_{0X} + \frac{k_{1X}}{1+a_{yx}([Y]^{n_{yx}})}$
3.	$Y \longrightarrow \emptyset$	$\gamma_Y[Y]$
4.	$\emptyset \longrightarrow Y$	$k_{0Y} + \frac{k_{1Y}}{1+a_{xy}([X]^{n_{xy}})}$

solve the full FSP model and 0.17 sec to solve the reduced model.

Table 12: Genetic toggle switch. True parameter values and the average values of the parameters visited by the ADAMH-FSP-Krylov and AMH-FSP chains. The "Start" column shows the starting point of both MCMC chains. This starting point is obtained from a numerical optimization procedure that seeks to maximize the full log-likelihood (3). Parameter values are shown in $\log_{10}$ scale. All parameters have unit sec^{-1}.

Parameter	True	Start	ADAMH-FSP-Krylov	AMH-FSP	AMH-ROM
$\log_{10}(k_{0X})$	-2.66e+00	-2.65e+00	-2.65e+00	-2.65e+00	-2.65e+00
$\log_{10}(k_{1X})$	-1.77e+00	-1.75e+00	-1.75e+00	-1.75e+00	-1.75e+00
$\log_{10}(\gamma_X)$	-3.42e+00	-3.40e+00	-3.40e+00	-3.40e+00	-3.40e+00
$\log_{10}(k_{0Y})$	-4.17e+00	-4.69e+00	-5.05e+00	-5.07e+00	-5.04e+00
$\log_{10}(k_{1Y})$	-1.80e+00	-1.77e+00	-1.77e+00	-1.77e+00	-1.77e+00
$\log_{10}(\gamma_Y)$	-3.42e+00	-3.39e+00	-3.39e+00	-3.39e+00	-3.39e+00
Log-posterior (un-normalizsed)	-9.2973e+03	-9.2919e+03	-9.2945e+03	-9.2945e+03	-9.2946e+03

Table 13: Genetic toggle switch. Bounds on the support of the log-uniform prior. Parameters have the same unit sec^{-1}.

Parameter	k_{0X}	k_{1X}	γ_X	k_{0Y}	k_{1Y}	γ_Y
Lower bound	1.00e-06	1.00e-06	1.00e-06	1.00e-06	1.00e-06	1.00e-06
Upper bound	1.00e-01	1.00e-01	1.00e-01	1.00e-01	1.00e-01	1.00e-01

Discussion and concluding remarks

There is a clear need for efficient computational algorithms for the uncertainty analysis of gene expression models. In this work, we proposed and investigated new approaches for Bayesian parameter inference of stochastic gene expression parameters from single-cell data that employ adaptive tuning of proposal distributions in addition to delayed acceptance MCMC and reduced-order modeling. Numerical tests confirm that the reduced model can be used to significantly speed up the sampling process without incurring much loss in accuracy.

A surprising observation from our numerical results is that once trained, the reduced model constructed by the ADAMH-FSP-Krylov closely matches the original FSP sampling results. This suggests that the ADAMH-FSP-Krylov algorithm could be used as a data-

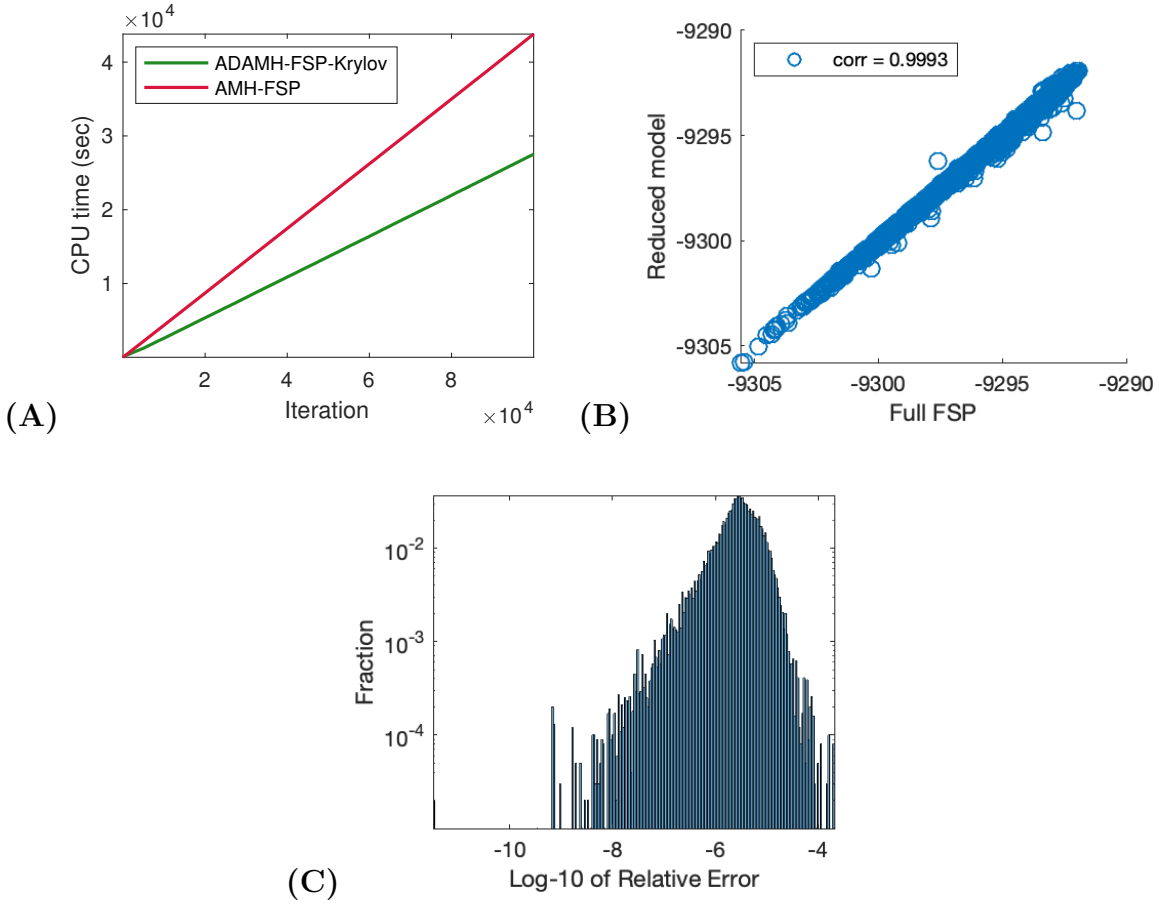

Figure 5: Genetic toggle switch example. **(A)** CPU time vs number of iterations for a sample run of the ADAMH-FSP-Krylov and the AMH-FSP. **(B)** Scatterplot of the unnomarlized log-posterior evaluated using the full FSP and the reduced model. Notice that the approximate and true values are almost identical with a correlation coefficient of approximately 0.9993. **(C)** Distribution of the relative error in the approximate log-posterior evaluations at the parameters accepted by the ADAMH chain.

Table 14: Genetic toggle switch example. Performance of the Adaptive Delayed Acceptance Metropolis-Hastings with Krylov-based reduced model (ADAMH-FSP-Krylov) vs the Adaptive Metropolis-Hastings with full FSP (AMH-FSP). The total chain length for each algorithm was $100,000$. The ADAMH-FSP-Krylov scheme uses markedly fewer full evaluations than the AMH-FSP scheme, and 98.36% of the parameters promoted by the first-stage are accepted in the second stage.

	mESS	CPU time (sec)	$\frac{\text{CPU time}}{\text{mESS}}$ (sec)	Number of full evaluations	Number of rejections	Number of rejections by full FSP
ADAMH-FSP-Krylov	4522.66	27524.78	6.09	23893	76498	391
AMH-FSP	4520.54	43799.22	9.69	100000	76587	76587
AMH-ROM	4666.93	17299.84	3.71	0	76579	0

Table 15: Genetic toggle switch example. Breakdown of CPU time spent in the main components of ADAMH-FSP-Krylov.

Component	Time occupied (sec)	Fraction of total time (per cent)
Full FSP Evaluation	10142.56	36.85
Reduced Model Evaluation	17040.89	61.91
Reduced Model Update	66.54	0.24
Total	27524.78	100.00

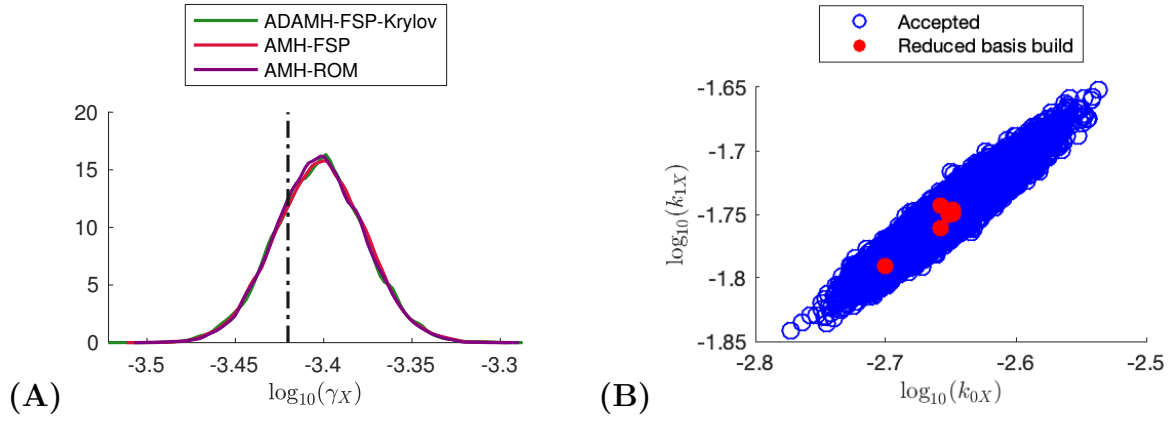

Figure 6: Genetic toggle switch example. **(A)** Estimations of the marginal posterior distribution of the parameter γ_X using the Adaptive Delayed Acceptance Metropolis-Hastings with Krylov reduced model (ADAMH-FSP-Krylov), the Adaptive Metropolis-Hastings with full FSP (AMH-FSP), and the approximate Adaptive Metropolis-Hastings using the reduced order model learned from the ADAMH-FSP-Krylov run (AMH-ROM). The dashed vertical line marks the true parameter value. **(B)** 2-D projections of parameter combinations accepted by the ADAMH scheme (blue) and parameter combinations used for reduced model construction (red).

driven method to learn reduced representations of the full FSP-based model, which could then be successfully substituted for the full FSP model in subsequent Bayesian updates. In other words, it could be equally accurate but more efficient to cease full FSP evaluations in the ADAMH scheme once confident about the accuracy of the reduced model. In our numerical tests, the ADAMH updates completed first 10-20% of the MCMC chain, at which point the remaining chain could have been sampled using only the reduced model. Perhaps other approaches to substitute function approximations into the expensive likelihood evaluations[57,58] could provide additional insights to the reduced order modeling approximations we have used.

While we have achieved a significant reduction in computational time with our implementation of the Krylov subspace projection, other model reduction algorithms may yet improve this performance.[59] For example, the reduced models discovered here achieved levels of accuracy (i.e., relative errors of 10^{-8} or less) that are much higher than one would expect to be necessary to compare models in light of far less accurate data. In light of this finding and the fact that parameter discrimination can be achieved at different levels of accuracy for different combinations of models and data,[60] we suspect that it could be advantageous to build less accurate models that can be evaluated in less time.

Our present work assumes the full FSP-based solution can be computed for use to learn the reduced model bases and to evaluate the second stage likelihood in the ADAMH-FSP-Krylov algorithm. For many problems, the required FSP state space can be so large that it would be impossible even to keep the full model in computer memory. Representing the FSP model in a low-rank tensor format[20] is a promising approach that we plan to investigate in order to overcome this limitation. Our current work has focused on using reduced models for uncertainty quantification, but the equally important task of finding optimal parameter fits should also benefit from reduced order modeling. For example, techniques from other engineering fields, such as trust-region methods,[61] may provide valuable improvements to infer stochastic models from gene expression data. In time, a wealth of algorithms and insight

remains to be gained by adapting computational technology from the broader computational science and engineering communities to analyze stochastic gene expression.

Acknowledgments

Research reported in this publication was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award numbers R25GM105608 and R35GM124747. The work reported here was partially supported by a National Science Foundation grant (DGE-1450032). Any opinions, findings, conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of the National Science Foundation. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Appendix: Mathematical proofs

Preliminaries on adaptive MCMC algorithms

We will derive ergodicity results in the following sections based on Theorem 1 in the paper of Roberts and Rosenthal,[47] and we will use some proof techniques of Theorem 1 from Cui et al.[31] for part of our analysis. All random variables we will discuss below will be of the form $X : \Omega \to \mathcal{X}$ where $\mathcal{X}$ is a metric space with the associated Borel σ-algebra $B(\mathcal{X})$.

Let $\mathcal{X}$ be the parameter space, assumed to have a metric space topology, and $\pi : B(\mathcal{X}) \to [0, 1]$ the target distribution to be sampled from by an adaptive MCMC algorithm. We will assume that π has a density $f : \mathcal{X} \to [0, \infty)$. Let K_γ denote a transition kernel that depends on an adaptation index $\gamma \in \mathcal{Y}$, and assume that each K_γ has π as an invariant distribution. We assume that for each fixed γ, an MCMC algorithm with K_γ as the Markov transition kernel will eventually converge to π, that is

$$\lim_{n\to\infty} \|K^n(x, .) - \pi\|_{TV} = 0$$

where $\|\mu - \nu\|_{TV} = \sup_{B\in B(\mathcal{X})} |\mu(B) - \nu(B)|$ is the total variation distance between two probability measures on $\mathcal{X}$.

Let X_n be the random variable representing the state of the adaptive MCMC at iteration n, and let Γ_n be the random variable representing the choice of kernel for updating from X_n to X_{n+1}. The state of the algorithm is then modeled by the discrete-time stochastic process $\{(X_n, \Gamma_n)\}$, whose transition between steps is determined by the underlying rules of the algorithm. Finally, let $\mathcal{G}_n = \sigma(\{X_0, \ldots, X_n, \Gamma_0, \ldots, \Gamma_n\})$ denote the filtration generated by $\{(X_n, \Gamma_n)\}$. Thus, each Γ_{n+1} is a $\mathcal{G}_{n+1}$-measurable random variable.

Roberts and Rosenthal proved the following important result, which gives sufficient conditions for ergodicity of an adaptive MCMC.

Theorem 0.1 (Theorem 1 in Roberts and Rosenthal[47])**.** *Consider an adaptive MCMC*

algorithm with state space $\mathcal{X}$ and adaptation index $\mathcal{Y}$, with transition kernels K_γ, $\gamma \in \mathcal{Y}$. The algorithm is ergodic if the following conditions hold

(i) (Simultaneous uniform ergodicity) For every $\varepsilon > 0$, there exists $N = N(\varepsilon)$ such that

$$\|K_\gamma^n(x, .) - \pi\|_{TV} < \varepsilon$$

for every $x \in \mathcal{X}$, $\gamma \in \mathcal{Y}$, and $n > N$.

(ii) (Diminishing adaptation) $\lim_{n\to\infty} D_n = 0$ in probability where

$$D_n = \sup_{x\in\mathcal{X}} \|K_{\Gamma_n}(x, .) - K_{\Gamma_{n+1}}(x, .)\|_{TV}$$

is a $\mathcal{G}_{n+1}$-measurable random variable.

We immediately get a useful corollary.

Corollary 0.2. *Consider an adaptive MCMC with state space $\mathcal{X}$ and transition kernels K_γ, $\gamma \in \mathcal{Y}$ that are ergodic w.r.t π. Assume that the following conditions are satisfied:*

(i) The algorithm satisfies the diminishing adaptiation condition.

(ii) $\mathcal{X}$ is a compact metric space.

(iii) $\mathcal{Y} = \cup_{j=1}^m \mathcal{Y}_j$ where each $\mathcal{Y}_j$ is a compact metric space.

(iv) For each $n = 1, 2, \ldots$, and on each set $\mathcal{X} \times \mathcal{Y}_j$ with the product metric space topology, the mapping

$$(x, \gamma) \mapsto S(x, \gamma; n, j) = \|K_\gamma^n(x, .) - \pi(.)\|_{TV}$$

is continuous.

Then, the adaptive MCMC algorithm is ergodic.

Proof. Our proof is a modification of the proof of Corollary 3 in.[47] Fix a number $\varepsilon > 0$ and an index $j \in \{1, \ldots, m\}$. Let W_n^j be the set of all $(x, \gamma) \in \mathcal{X} \times \mathcal{Y}_j$ such that

$$S(x, \gamma; n, j) < \varepsilon.$$

Since each kernel is ergodic, for every $(x, \gamma) \in \mathcal{X} \times \mathcal{Y}_j$ there exists some n such that $(x, \gamma) \in W_n^j$, and that $S(x, \gamma; n', j) < \varepsilon$ for all $n' > n$. We thus have

$$\mathcal{X} \times \mathcal{Y}_j = \cup_{n=1}^{\infty} {W_n}^j$$

Due to continuity, each ${W_n}^j$ is an open set. By compactness, there exists a finite subcover $\{{W_{n_i}}^j\}_{i=1}^{r_j}$ for $\mathcal{X} \times \mathcal{Y}_j$. Choose $N_j(\varepsilon)$ to be the maximum of all $n_1, \ldots, n_{r_j}$. Then, choose $N(\varepsilon) = N_1(\varepsilon) + \ldots + N_m(\varepsilon)$, we have $\|K_\gamma^n(x, .) - \pi\|_{TV} < \varepsilon$ for all $n > N(\varepsilon)$ and $(x, \gamma) \in \mathcal{X} \times \mathcal{Y}$. Thus, simultaneous uniform ergodicity is satisfied. Combining with diminishing adaptation, the preceding theorem shows that the algorithm is ergodic. □

Convergence of adaptive DAMH with diminishing model adaptations

In this section, we analyze the convergence of an adaptive variant of the DAMH. As seen in the pseudocode of Algorithm 4, this variant modifies the approximation and the proposal density at every step, using the samples accepted so far on the chain. The update of the approximate model occurs randomly, with the upate probability at step n pre-specified as $a(n)$.

Proposition 0.3. *Consider an adaptive delayed acceptance Metropolis-Hastings algorithm with the target distribution supported on a state space $\mathcal{X}$, proposal adaptation space $\mathcal{Y}$, approximation space $\mathcal{Z}$. Let f be the density of the target distribution π with respect to a finite reference measure λ, that is, $\pi(dx) = f(x)\lambda(dx)$. Let $f^\star_{x,\varphi}$ be the family of approximations*

Algorithm 4 Adaptive Delayed Acceptance MH with probabilistic approximation adaptation.

Input:

Target density $f(.)$;
Proposal densities $q_\gamma(.,.)$;
Posterior density approximations $f^\star_{x,\varphi}(.)$;
Adaptation probability $a(n), \quad , n = 1, 2, \ldots$;
Chain length N.

Assume that $x_n = x$, $\gamma_n = \gamma$ at iteration n. The next sample is determined by the following steps.

1. Propose a candidate x' from the proposal density $q_\gamma(x, .)$.

2. Compute the first-step acceptance probability

$$\alpha_{\gamma,\varphi}(x, x') = \min\left\{1, \frac{q_\gamma(x', x) f^\star_{x,\varphi}(x')}{q_\gamma(x, x') f^\star_{x,\varphi}(x)}\right\}$$

3. With probability $\alpha_{\gamma,\varphi}$, set $y = x'$. Otherwise, set $y = x$. The actual proposal distribution is

$$Q^\star_{x,\gamma,\varphi}(x, dz) = q_\gamma(x, z)\alpha_{\gamma,\varphi}(x, z)\lambda(dz) + \delta_x(dz)(1 - r_{\gamma,\varphi}(x)),$$

where

$$r_{\gamma,\varphi}(x) = \int_{\mathcal{X}} q_\gamma(x, z')\alpha_{\gamma,\varphi}(x, z')\lambda(dz')$$

is the overall probability that a proposal is accepted in the first step.

4. Set $x_{n+1} = y$ with probability

$$\beta_{\gamma,\varphi}(x, x') = \min\left\{1, \frac{q_\gamma(x', x) f^\star_x(x') f(x')}{q_\gamma(x, x') f^\star_x(x) f(x)}\right\}.$$

Otherwise, set $x_{n+1} = x_n$.

5. With probability $a(n)$, update the approximation $f^\star_{x,\varphi}$.

6. Update the first-step proposal $q_\gamma(x, .)$.

Output:

Samples $x_1, \ldots, x_N$.

to f. Let q_γ be the first-step proposal densities. The algorithm is ergodic under the following conditions:

(i) $\mathcal{X}, \mathcal{Y}$ *are compact metric spaces, and* $\mathcal{Z} = \cup_{j=1}^{m} \mathcal{Z}_j$ *where each* $\mathcal{Z}_j$ *is a compact metric space.*

(ii) For each fixed γ, φ*, the transition kernel* $K_{\gamma,\varphi}$ *is ergodic.*

(iii) $\lambda\{x\} = 0$ *for all* $x \in \mathcal{X}$*.*

(iv) The mapping $(x, y, \gamma) \mapsto q_\gamma(x, y)$ *is continuous and uniformly bounded on* $\mathcal{X} \times \mathcal{X} \times \mathcal{Y}$ *which is a compact metric space equipped with the product space metric.*

(v) For each $y \in \mathcal{Y}$*, the mapping* $(x, \varphi) \mapsto f^{\star}_{x,\varphi}(y)$ *is continuous on each* $\mathcal{X} \times \mathcal{Z}_j$*.*

(vi) Diminishing adaptation: The chain (Γ_n, Φ_n) *satisfies*

$$\lim_{n\to\infty} \sup_{x\in\mathcal{X}} \|K_{\Gamma_{n+1},\Phi_{n+1}}(x, .) - K_{\Gamma_n,\Phi_n}(x, .)\|_{TV} = 0$$

in probability.

Proof. The ADAMH could be viewed as an adaptive MCMC algorithm with state space $\mathcal{X}$ and adaptation space $\mathcal{Y} \times \mathcal{Z}$. In order to apply corollary 0.2, we will prove that for any fixed $n = 1, 2, \ldots$, and fixed $j = 1, \ldots, m$, the mapping

$$(x, \gamma, \varphi) \mapsto \|K^n_{\gamma,\varphi}(x, .) - \pi\|_{TV}$$

is continuous on $\mathcal{X} \times \mathcal{Y} \times \mathcal{Z}_j$. In order to do so, we proceed as in the proof of theorem 1 in.[31] Fix $(x, \gamma, \varphi) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Z}_j$, the transition kernel for the DAMH associated with (x, γ, φ) is

$$K_{\gamma,\varphi}(x, dz) = q_\gamma(x, z)\alpha_{\gamma,\varphi}(x, z)\beta_{\gamma,\varphi}(x, z)\lambda(dz) + \delta_x(dz)(1 - \rho_{\gamma,\varphi}(x)),$$

where $\alpha_{\gamma,\varphi}(x,z) = \min\left\{1, \frac{q_\gamma(z,x)f^\star_{x,\varphi}(z)}{q_\gamma(x,z)f^\star_{x,\varphi}(x)}\right\}$ is the first step acceptance probability, $\beta_{\gamma,\varphi}(x,z) = \min\left\{1, \frac{q_\gamma(z,x)f^\star_x(z)f(z)}{q_\gamma(x,z)f^\star_x(x)f(x)}\right\}$ is the second step acceptance probability, and

$$\rho_{\gamma,\varphi}(x) = \int_{\mathcal{X}} q_\gamma(x,z)\alpha_{\gamma,\varphi}(x,z)\beta_{\gamma,\varphi}(x,z)\lambda(dz)$$

is the overall probability for a proposal to be accepted.

Fix the value of z, then due to conditions (iv) and (v), $g(x,z,\gamma,\varphi) = q_\gamma(x,z)\alpha_{\gamma,\varphi}(x,z)\beta_{\gamma,\varphi}(x,z)$ is jointly continuous in $(x,\gamma,\varphi) \in \mathcal{X}\times\mathcal{Y}\times\mathcal{Z}_j$. Furthermore, condition (iv) implies that the functions $z \mapsto g(x,z,\gamma,\varphi)$ is uniformly bounded for $(x,\gamma,\varphi) \in \mathcal{X}\times\mathcal{Y}\times\mathcal{Z}_j$. By the bounded convergence theorem, $\rho_{\gamma,\varphi}(x)$ is jointly continuous in the three variables x,γ,φ.

By induction, we can show that the n-step transition kernel has the form

$$K^n_{\gamma,\varphi}(x,dz) = g_n(x,z,\gamma,\varphi)\lambda(dz) + \delta_x(dz)\left(1-\rho_{\gamma,\varphi}(x)\right)^n$$

where g_n is an appropriate function that is jointly continuous in x,γ and φ.

From condition (iii), δ_x and π are orthogonal measures. Therefore,

$$\|K^n_{\gamma,\varphi}(x,.) - \pi\|_{TV} = (1-\rho_{\gamma,\varphi}(x))^n + \frac{1}{2}\int_{\mathcal{X}} (g_n(x,z,\gamma,\varphi) - f(z))\lambda(dz).$$

The integral on the right hand side is jointly continuous in x,γ,φ due to the bounded convergence theorem. This shows that $\|K^n_{\gamma,\varphi}(x,.) - \pi\|_{TV}$ is continuous in the variable $(x,\gamma,\varphi) \in \mathcal{X}\times\mathcal{Y}\times\mathcal{Z}_j$. From this, conditions (i), (vi) and corollary 0.2 combined show that the algorithm is ergodic. □

Proposition 0.4. *Assume the ADAMH with probabilistic model adaptation satisfies conditions (i)-(v) in proposition* 0.3. *Assume further that the proposal is symmetric, that the approximate posterior adaptation probability* $a(n) \to 0$ *as* $n \to \infty$, *and that* $d_{\mathcal{Y}}(\Gamma_{n+1},\Gamma_n) \to 0$ *in probability (here* $d_{\mathcal{Y}}$ *denote the metric on* $\mathcal{Y}$*). Then, the algorithm satisfies diminishing adaptation.*

Proof. All conditions for ergodicity in proposition 0.3 are satisfied, except for the diminishing adaptation that we will verify. Fix a value of n. Consider a fixed set of values $(\gamma_j, \varphi_j)_{j=1}^n$ of adaptivity parameters of the ADAMH chain up to iteration n.

Fix an event $A \in B(\mathcal{X})$ and $x \in \mathcal{X}$. We have

$$|K_{\gamma_{n+1},\varphi_{n+1}}(x,A) - K_{\gamma_n,\varphi_n}(x,A)| \leq \underbrace{\left|K_{\gamma_{n+1},\varphi_{n+1}}(x,A) - K_{\gamma_n,\varphi_{n+1}}(x,A)\right|}_{D_1} + \underbrace{\left|K_{\gamma_n,\varphi_{n+1}}(x,A) - K_{\gamma_n,\varphi_n}(x,A)\right|}_{D_2}$$

We bound each term separately. First of all, we have $D_2 = 0$ if $\varphi_n = \varphi_{n+1}$ and $D_2 \leq K_{\gamma_n,\varphi_{n+1}}(x,A) + K_{\gamma_n,\varphi_n}(x,A) \leq 2$ if $\varphi_n \neq \varphi_{n+1}$, with the latter event taking place with probability less than $a(n)$.

Due to the symmetry of the proposal, the first and second step acceptance probabilities do not depend on the choice of γ. This and the uniform continuity of $q_\gamma(x,y)$ gives us

$$\begin{aligned} D_1 &\leq 2\int_{\mathcal{X}} |(q_{\gamma_{n+1}}(x,y) - q_{\gamma_n}(x,y))\alpha_{\varphi_n}(x,y)\beta_{\varphi_n}(x,y)|\lambda(dy) \\ &\leq 2\int_{\mathcal{X}} |q_{\gamma_{n+1}}(x,y) - q_{\gamma_n}(x,y)|\lambda(dy) \\ &\leq C \cdot d_{\mathcal{Y}}(\gamma_n, \gamma_{n+1}) \end{aligned}$$

where $C > 0$ is independent of x, y, γ and φ.

Combining the bounds on D_1 and D_2 we get

$$|K_{\gamma_{n+1},\varphi_{n+1}}(x,A) - K_{\gamma_n,\varphi_n}(x,A)| \leq C \cdot d_{\mathcal{Y}}(\gamma_n, \gamma_{n+1}) + 2\chi([\varphi_n = \varphi_{n+1}])$$

where $\chi(A) = 1$ if A is true and 0 otherwise. Taking the supremum over all x and A we get

$$D_n = \sup_{x\in\mathcal{X}} \|K_{\gamma_{n+1},\varphi_{n+1}}(x,.) - K_{\gamma_n,\varphi_n}(x,.)\|_{TV} \leq C \cdot d_{\mathcal{Y}}(\gamma_n, \gamma_{n+1}) + 2\chi([\varphi_n \neq \varphi_{n+1}])$$

Fix a scalar $\varepsilon > 0$. The set of runs where $D_n < \varepsilon$ include sample chains where both events $\varphi_n = \varphi_{n+1}$ and $C.d_{\mathcal{Y}}(\gamma_n, \gamma_{n+1}) < \varepsilon$ hold. Therefore, the event $[D_n \geq \varepsilon]$ is a subset of the

event $[C \cdot d_{\mathcal{Y}}(\Gamma_n, \Gamma_{n+1}) \geq \varepsilon] \cup [\Phi_n \neq \Phi_{n+1}]$. We therefore have

$$\begin{aligned}\mathbb{P}[D_n \geq \varepsilon] &\leq \mathbb{P}[C \cdot d_{\mathcal{Y}}(\Gamma_n, \Gamma_{n+1}) \geq \varepsilon] + \mathbb{P}[\Phi_n \neq \Phi_{n+1}] \\ &\leq \mathbb{P}[d_{\mathcal{Y}}(\Gamma_n, \Gamma_{n+1}) \geq \varepsilon/C] + a(n).\end{aligned}$$

The last right hand side of the inequality above converges to 0 as $n \to \infty$. Therefore, D_n converges to 0 in probability. The diminishing adaptation condition is satisfied and the algorithm is ergodic. □

Regularity of the ROM-based likelihood approximation

Let $\mathcal{S}_j$ be the set of all $n \times j$ matrices $\boldsymbol{Q}$ such that $\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}_{j\times j}$. It is known that $\mathcal{S}_j$ with the metric defined by the induced matrix 2-norm

$$\|\boldsymbol{Q}\| = \max_{\boldsymbol{x}\neq\boldsymbol{0}} \frac{\|\boldsymbol{Q}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2}$$

is a compact metric space (indeed, it is the inverse image of $\boldsymbol{I}_{j\times j}$ via the continuous mapping $\boldsymbol{A} \mapsto \boldsymbol{A}^T\boldsymbol{A}$). Let $m_{\max}$ be the maximum dimension allowed in the reduced basis and let $\boldsymbol{\Phi}$ be a particular basis set constructed during a run of the ADAMH chain, then there exists a tuple $(j_1, \ldots, j_{n_B})$ with $1 \leq j_k \leq m_{\max}$ such that

$$\boldsymbol{\Phi} \in \mathcal{S}_{j_1} \times \cdots \times \mathcal{S}_{j_{n_B}} := \boldsymbol{S_j}$$

Thus, the set of all possible choices of reduced basis set $\boldsymbol{\Phi}$ is the finite union of all $\boldsymbol{S_j}$ with $\boldsymbol{j}$ bounded elementwise by $m_{\max}$. Note that each $\boldsymbol{S_j}$ is a compact metric space with the product space topology. Thus, we can apply the theory developed in the previous section to show that the ADAMH-FSP-Krylov is ergodic. The following propositions concern the continuity in the change of the reduced-order approximations with respect to the change in basis.

Proposition 0.5. *Fix a space* $\boldsymbol{S_j}$ *as above, and let* $\boldsymbol{\Phi}$ *and* $\boldsymbol{\Psi}$ *be elements of this space. For for every fixed* $\boldsymbol{\theta} \in \Theta$ *we have*

$$L^{\star}_{\boldsymbol{\Psi}}(\boldsymbol{\theta}) \to L^{\star}_{\boldsymbol{\Phi}}(\boldsymbol{\theta})$$

as $\boldsymbol{\Psi} \to \boldsymbol{\Phi}$ *in* $\boldsymbol{S_j}$*, where* $L^{\star}_{\boldsymbol{\Phi}}$ *is the approximation to the FSP log-likelihood as defined in eq.* (16)*.*

Proof. From eq. (7), it is clear that the mapping $\boldsymbol{\Phi} \mapsto p_{\boldsymbol{\Phi}}(t_k)$ is continuous on $\boldsymbol{S_j}$ for all time points t_k. The mapping $\boldsymbol{\Phi} \mapsto L^{\star}_{\boldsymbol{\Phi}}(\boldsymbol{\theta})$ is a composition of continuous mappings $\boldsymbol{\Phi} \mapsto p_{\boldsymbol{\Phi}}(t_k)$ and $\boldsymbol{p} \mapsto \sum_{j=1}^{n_i} \log(\varepsilon \vee \boldsymbol{p}_j)$ and is therefore continuous. □

Ergodicity of the ADAMH-FSP-Krylov algorithm

Proposition 0.6. *The ADAMH-FSP-Krylov algorithm is ergodic.*

Proof. We apply proposition 0.3 with $\mathcal{X} = \boldsymbol{\Theta}$. The proposal densities of the first step are Gaussian with γ being the modified empirical covariance matrix as in the adaptive Metropolis Algorithm.[27] Similar to the proof of Theorem 1 in Haario et al.,[27] we can take $\mathcal{Y}$ to be a closed, bounded subset of the set of positive definite matrices. The reduced model space is $\mathcal{Z} = \cup_{\boldsymbol{j}} \boldsymbol{S_j}$ the finite union of the compact spaces $\boldsymbol{S_j}$ with $\boldsymbol{j} \leq m_{\max}$ pointwise. These spaces satisfy condition (i), and the proposal density satisfies condition (iv).

The posterior density is

$$f(\boldsymbol{\theta}) = \pi_0(\boldsymbol{\theta}) \exp(-L(D|\boldsymbol{\theta})),$$

and the approximate posterior densities are

$$f^{\star}_{\boldsymbol{\Phi}}(\boldsymbol{\theta}) = \pi_0(\boldsymbol{\theta}) \exp(-L^{\star}_{\boldsymbol{\Phi}}(D|\boldsymbol{\theta})),$$

where these are the densities of the true and approximate posterior distributions with respect to the Lebesgue measure. From Theorem 1 in Christen and Fox,[28] condition (ii) is satisfied.

Condition (v) is then satisfied using proposition 0.5.

Since the empirical covariances are computed from values in a bounded set, the modification to the empirical covariance matrix γ at step n is $O(1/n)$, so changes in Γ_n converge to 0 (see Haario et al.[27]). Thus, the conditions in proposition 0.4 are satisfied. The algorithm therefore satisfies all sufficient conditions for ergodicity outlined in proposition 0.3. □

References

(1) McAdams, H. H.; Arkin, A. *Proc. Natl. Acad. Sci. U. S. A.* **1997**, *94*, 814–819.

(2) Elowitz, M. B.; Levine, A. J.; Siggia, E. D.; Swain, P. S. *Science* **2002**, *297*, 1183–1186.

(3) Kaern, M.; Elston, T. C.; Blake, W. J.; Collins, J. J. *Nature Rev. Genet.* **2005**, *6*, 451–464.

(4) Gillespie, D. T. *Physica A* **1992**, *188*, 404–425.

(5) Neuert, G.; Munsky, B.; Tan, R. Z.; Teytelman, L.; Khammash, M.; Oudenaarden, A. V. *Science* **2013**, *339*, 584–587.

(6) Shepherd, D. P.; Li, N.; Micheva-Viteva, S. N.; Munsky, B.; Hong-Geller, E.; Werner, J. H. *Anal. Chem.* **2013**, *85*, 4938–4943.

(7) Munsky, B.; Fox, Z.; Neuert, G. *Methods* **2015**, *85*, 12–21.

(8) Gómez-Schiavon, M.; Chen, L.; West, A. E.; Buchler, N. E. *Genome Biol.* **2017**, *18*, 164.

(9) Munsky, B.; Khammash, M. *J. Chem. Phys.* **2006**, *124*, 044104.

(10) Munsky, B.; Li, G.; Fox, Z. R.; Shepherd, D. P.; Neuert, G. *PNAS* **2018**,

(11) Peherstorfer, B.; Willcox, K.; Gunzburger, M. *SIAM Review* **2018**, *60*, 550–591.

(12) Asher, M. J.; Croke, B. F. W.; Jakeman, A. J.; Peeters, L. J. M. *Water Resour. Res.* **2015**, *51*, 5957–5973.

(13) Razavi, S.; Tolson, B. A.; Burn, D. H. *Water Resour. Res.* **2012**, *48*.

(14) Pinnau, R. *Model Order Reduction: Theory, Research Aspects and Applications*; Springer Berlin Heidelberg, 2008; Vol. 13; pp 95–109.

(15) Benner, P.; Gugercin, S.; Willcox, K. *SIAM R* **2015**, *57*, 483–531.

(16) Peleš, S.; Munsky, B.; Khammash, M. *J. Chem. Phys.* **2006**, *125*, 1–13.

(17) Munsky, B.; Khammash, M. *IEEE Trans. Aut. Contrl.* **2008**, *53*, 201–214.

(18) Tapia, J. J.; Faeder, J. R.; Munsky, B. *2012 IEEE 51st IEEE Conf. Decis. Ctrl. (CDC)* **2012**, *836*, 5361–5366.

(19) Vo, H. D.; Sidje, R. B. Solving the chemical master equation with aggregation and Krylov approximations. Proceedings of IEEE 55th Conference on Decision and Control. 2016; pp 7093–7098.

(20) Kazeev, V.; Khammash, M.; Nip, M.; Schwab, C. *PLoS Comput. Biol.* **2014**, *10*.

(21) Dolgov, S.; Khoromskij, B. N. *Numer. Linear Algebra Appl.* **2013**, *22*, 197–219.

(22) Vo, H. D.; Sidje, R. B. *J. Chem. Phys.* **2017**, *147*.

(23) Dayar, T.; Orhan, M. C. *Numer. Linear Algebra Appl.* **2018**, *25*, e2158, e2158 nla.2158.

(24) Waldherr, S.; Haasdonk, B. *BMC systems biology* **2012**, *6*, 81.

(25) Liao, S.; Vejchodský, T.; Erban, R. *J. R. Soc. Interface* **2015**, *12*, 20150233.

(26) Oseledets, I. V. *SIAM J. Sci. Comput.* **2011**, *33*, 2295–2317.

(27) Haario, H.; Saksman, E.; Tamminen, J. *Bernoulli* **2001**, *7*, 223.

(28) Christen, J. A.; Fox, C. *J. Comput. Graph. Stat.* **2005**, *14*, 795–810.

(29) Efendiev, Y.; Hou, T.; Luo, W. *SIAM J. Sci. Comput.* **2006**, *28*, 776–803.

(30) Cui, T.; Fox, C.; O'Sullivan, M. J. *Water Resources Research* **2011**, *47*.

(31) Cui, T.; Fox, C.; O'Sullivan, M. *Adaptive Error Modelling MCMC sampling for Large Scale Inverse Problems*; 2012.

(32) Golightly, A.; Henderson, D. A.; Sherlock, C. *Statistics and Computing* **2015**, *25*, 1039–1055.

(33) Saad, Y. *SIAM J. Numer. Anal.* **1992**, *29*, 209–228.

(34) Sidje, R. B. *ACM Trans. Math. Softw.* **1998**, *24*, 130–156.

(35) Burrage, K.; Hegland, M.; MacNamara, S.; Sidje, R. B. In 150^{th} *Markov Anniversary Meeting, Charleston, SC, USA*; Langville, A., Stewart, W., Eds.; Boson Books, 2006; pp 21–38.

(36) Sidje, R. B.; Vo, H. D. *Math. Biosci.* **2015**, *269*, 10–16.

(37) Gauckler, L.; Yserentant, H. *ESAIM. Math. Model.* **2014**, *48*, 1757–1775.

(38) Femino, A. M.; Fay, F. S.; Fogarty, K.; Singer, R. H. *Science* **1998**, *280*, 585–590.

(39) Raj, A.; van Oudenaarden, A. *Cell* **2008**, *135*, 216–226.

(40) Chaturantabut, S.; Sorensen, D. C. *SIAM J. Sci. Comput.* **2010**, *32*, 2737–2764.

(41) Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E. *J. Chem. Phys.* **1953**, *21*, 1087–1092.

(42) Hastings, W. K. *Biometrika* **1970**, *57*, 97–109.

(43) Roberts, G. O.; Rosenthal, J. S. *Probability Surveys* **2004**, *1*, 20–71.

(44) Roberts, G. O.; Gelman, A.; Gilks, W. R. *Annal. Appl. Prob.* **1997**, *7*, 110–120.

(45) Cui, T.; Marzouk, Y. M.; Willcox, K. E. *Int. J. Numer. Meth. Engnr.* **2015**, *102*, 966–990.

(46) Golub, G.; Van Loan, C. *Matrix Computations*, 4th ed.; John Hopkins University Press, 2012.

(47) Roberts, G.; Rosenthal, J. S. *J. Appl. Probability* **2007**, *44*, 458–475.

(48) Vats, D.; Flegal, J. M.; Jones, G. L. *Multivariate Output Analysis for Markov Chain Monte Carlo*; 2017.

(49) Acerbi, L. multiESS. `https://https://github.com/lacerbi/multiESS`, 2018.

(50) Gillespie, D. T. *J. Phys. Chem.* **1977**, *81*, 2340–2361.

(51) Munsky, B.; Neuert, G.; van Oudenaarden, A. *Science* **2012**, *336*, 183–187.

(52) Peccoud, J.; Ycart, B. *Theoretical Pop. Biol.* **1995**, *48*, 222 – 234.

(53) Golding, I.; Paulsson, J.; Zawilski, S. M.; Cox, E. C. *Cell* **2005**, *123*, 1025–1036.

(54) Iyer-Biswas, S.; Hayot, F.; Jayaprakash, C. *Phys. Rev. E* **2009**, *79*, 031911.

(55) Gardner, T.; Cantor, C.; Collins, J. *Nature* **2000**, *403*, 339–342.

(56) Fox, Z. R.; Munsky, B. *bioRxiv* **2018**,

(57) Conrad, P. R.; Marzouk, Y. M.; Pillai, N. S.; Smith, A. *J. Amer. Stat. Assoc.* **2016**, *111*, 1591–1607.

(58) Conrad, P.; Davis, A.; Marzouk, Y.; Pillai, N.; Smith, A. *SIAM/ASA J. Uncertainty Quantification* **2018**, *6*, 339–373.

(59) Benner, P.; Cohen, A.; Ohlberger, M.; Wilcox, K. e. *Model reduction and approximation: Theory and algorithms*; SIAM, 2017.

(60) Fox, Z.; Neuert, G.; Munsky, B. *J. Chem. Phys.* **2016**, *145*.

(61) Qian, E.; Grepl, M.; Veroy, K.; Willcox, K. *SIAM J. Sci. Comput.* **2017**,