

## RESEARCH ARTICLE

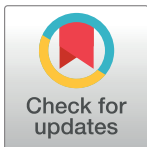
## Monte Carlo samplers for efficient network inference

Zeliha Kilic<sup>1</sup>, Max Schweiger<sup>2,3</sup>, Camille Moyer<sup>2,4</sup>, Steve Pressé<sup>1,2,5\*</sup>

**1** Department of Structural Biology, St. Jude Children's Research Hospital, Memphis, Tennessee, United States of America, **2** Center for Biological Physics, ASU, Tempe, Arizona, United States of America, **3** Department of Physics ASU, Tempe, Arizona, United States of America, **4** School of Mathematics and Statistical Sciences, ASU, Tempe, Arizona, United States of America, **5** School of Molecular Sciences, ASU, Tempe, Arizona, United States of America

☞ These authors contributed equally to this work.

\* [spresse@asu.edu](mailto:spresse@asu.edu)



## OPEN ACCESS

**Citation:** Kilic Z, Schweiger M, Moyer C, Pressé S (2023) Monte Carlo samplers for efficient network inference. *PLoS Comput Biol* 19(7): e1011256. <https://doi.org/10.1371/journal.pcbi.1011256>

**Editor:** James R. Faeder, University of Pittsburgh, UNITED STATES

**Received:** January 4, 2023

**Accepted:** June 9, 2023

**Published:** July 18, 2023

**Copyright:** © 2023 Kilic et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All data used in this manuscript is available at <https://doi.org/10.5281/zenodo.8056901> (DOI:10.5281/zenodo.8056901).

**Funding:** S.P. acknowledges support from NIH NIGMS (R01GM130745), NIH NIGMS (R01GM134426), NIH NIGMS MIRA (R35GM148237). All authors received salaries from NIH during the study. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing interests:** The authors declare no competing interest.

## Abstract

Accessing information on an underlying network driving a biological process often involves interrupting the process and collecting snapshot data. When snapshot data are stochastic, the data's structure necessitates a probabilistic description to infer underlying reaction networks. As an example, we may imagine wanting to learn gene state networks from the type of data collected in single molecule RNA fluorescence *in situ* hybridization (RNA-FISH). In the networks we consider, nodes represent network states, and edges represent biochemical reaction rates linking states. Simultaneously estimating the number of nodes and constituent parameters from snapshot data remains a challenging task in part on account of data uncertainty and timescale separations between kinetic parameters mediating the network. While parametric Bayesian methods learn parameters given a network structure (with known node numbers) with rigorously propagated measurement uncertainty, learning the number of nodes and parameters with potentially large timescale separations remain open questions. Here, we propose a Bayesian nonparametric framework and describe a hybrid Bayesian Markov Chain Monte Carlo (MCMC) sampler directly addressing these challenges. In particular, in our hybrid method, Hamiltonian Monte Carlo (HMC) leverages local posterior geometries in inference to explore the parameter space; Adaptive Metropolis Hastings (AMH) learns correlations between plausible parameter sets to efficiently propose probable models; and Parallel Tempering takes into account multiple models simultaneously with tempered information content to augment sampling efficiency. We apply our method to synthetic data mimicking single molecule RNA-FISH, a popular snapshot method in probing transcriptional networks to illustrate the identified challenges inherent to learning dynamical models from these snapshots and how our method addresses them.

## Author summary

Decoding biochemical reaction networks—the number of species, reactions connecting them and reaction rates—from snapshot data, such as data drawn from single molecule

fluorescence *in situ* hybridization (smFISH) experiments, is of critical interest. Yet, this task's challenges are currently addressed with network specification heuristics since: 1) network size cannot be specified independently of rates; 2) rates may be separated by orders of magnitude, generating stiff ODEs; 3) uncertainty is not propagated into networks and parameters. We present, inspired by recent computational statistics tools, a method to simultaneously deduce reaction networks and associated rates from snapshot data while propagating error over all unknowns. We achieve this by treating network models as random variables and, within the Bayesian nonparametric paradigm, develop a posterior over models themselves. This multidimensional posterior naturally contains multiple hills and valleys, so we propose a combination of samplers allowing for the first simultaneous and self-consistent inference of networks and their associated rates. Our method's ability to treat arbitrary numbers of states contrasts the current state of the art and may modify previous biological conclusions based on network-determination heuristics. We demonstrate on method to synthetic data mimicking smFISH experiments and demonstrate its improvement over naive MCMC schemes.

## 1 Introduction

Systems and Computational Biology model allosteric enzyme control [1], chromatin re-organization [2], transcriptional regulation [3], metabolic interactions [4] and tumor growth [5], among other phenomena, with reaction networks. Ideally we could learn the structure of reaction networks by observing them continuously in time. However, this is often either impossible or introduces additional complications to the required model.

For example, actively monitoring transcription by fluorescence may introduce delays arising from maturation of co-translationally generated protein and label [6]. Also, since single molecule monitoring of transient gene expression dynamics in continuous time requires tracking diffraction-limited actively diffusing RNA, these methods are limited to low-expression genes to avoid crowded tracking [7]. Meanwhile, deducing a reaction network for the expressed genes may only require levels of RNA at various time points [8].

In addition to errors incurred in monitoring dynamics in real-time, in some cases, it may be altogether unnecessary to have time-resolved information [9], for example when probing steady-state reaction network dynamics.

Indeed, sufficient statistics might be accessed by interrupting an ensemble of system instances (*e.g.*, fixing individual cells) and recording instantaneous data from them at successive times. We refer to data collected in this manner as 'snapshot data'. For example, if we wish to infer gene networks, we may randomly select a subpopulation of cells at various time points and subsequently lyse or fix them to enumerate their RNAs [10].

Promising methods using snapshot acquisition exist across the biological sciences, including: pulse-chase [11, 12], sequencing experiments (RNAseq) [13–17], single molecule Fluorescence *In Situ* Hybridization (smFISH) [18, 19] and associated methods of RNA quantification such as expansion microscopy [20]. As such, developing analytical methods to deduce reaction networks suiting snapshot data is of primary concern to Systems and Computational Biology.

While in practice the reaction network, its connectivity between nodes, and its associated rates are encoded in the data, extracting this information has presented unique challenges [21]. Time scales separating rate values may be considerable [9] and reaction networks with rates spanning a broad set of time scales increases the stiffness of the resulting differential

equations [22, 23]. Furthermore, multiple network structures and reaction rate combinations may fit the data almost equally well (put otherwise, different networks describing observations similarly well may have drastically different rates) [24]; additionally, even for a known network, multiple parameter combinations may fit the data equally well (parameters are highly correlated due to additive changes introduced in gene expression) [9]. In fact, owing to these complications, multiple models may be comparably probable *a posteriori*, a characteristic we later refer to as ‘approximate model indeterminacy.’

Within a Bayesian inference setting to estimate reaction networks, these complications introduce hills and valleys in the multidimensional posterior over models naturally leading to the computational demise of naive Markov-Chain Monte Carlo (MCMC) methods [25] that have so far been limited to decoding rates for hand-specified nodes and connectivities between nodes [3, 24, 26, 27]. Within the current MCMC methods, determining the underlying reaction network or calibrating certain rates or time-dependencies may require additional control experiments leading to prohibitive experimental overhead [3] or a prohibitive number of core-hours on computing clusters [28]. Working within a Bayesian nonparametric setting where the number of nodes (not just the reaction rates connecting a known number of nodes) are to be inferred, we exchange these problems with more tractable ones, addressable by modern computational inference techniques. As we will see shortly, we use nonparametrics by employing Beta-Bernoulli process priors [29–33] to learn networks, using a selection of advanced sampling techniques identified below.

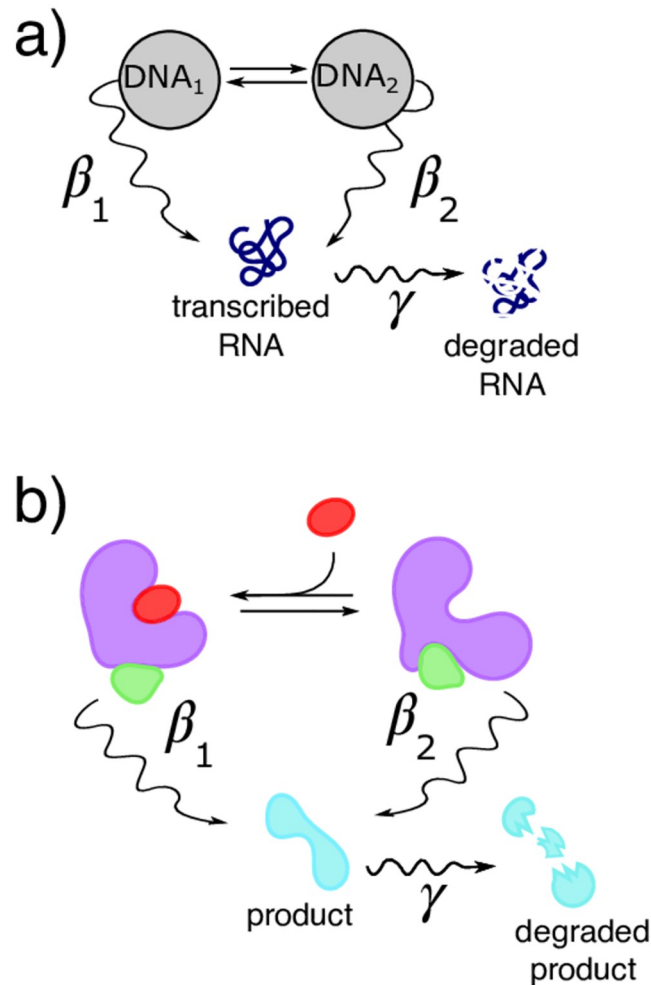
To overcome difficulties presented by rates spanning orders of magnitude and unknown number of nodes and their connectivities, we systematically explore combinations of MCMC samplers. That is, in order to resolve the stiffness of ODEs involved in likelihood calculation, we employ Adaptive Metropolis-Hastings (AMH) to propose new samples based on previous ones; in order to propose probable parameters based on the local geometry of the posterior, we employ Hamiltonian Monte Carlo (HMC), avoiding the overhead incurred by costly computational or experimental rate or model calibration; in order to maximize the explored region of the space of models, we package the above sampling schemes within a Parallel Tempering (PT) MCMC algorithm, allowing our method to characterize the posterior in a reasonable number of samples. Each of these methods and their particular advantages within the Bayesian paradigm are outlined in greater detail in Materials and Methods.

Our combination of sampling techniques provides increased efficiency which turns out to be critical in estimating network node numbers and rates simultaneously and self-consistently, going beyond existing state-of-the-art rate inference techniques [3, 25, 34]. To demonstrate this, we tested our sampling algorithm on a range of biologically realistic networks and rate values and assessed the added efficiency of each novel sampling technique, by replicating our analysis with each method omitted. We demonstrate that only our full method can simultaneously learn reaction rates and network structures at high sampling efficiency for a variety of underlying networks [35].

## 2 Materials and methods

### 2.1 A concrete example of a biochemical network

Here we focus on deducing reaction network models, termed ‘gene networks’, from snapshot RNA count data inspired by smFISH experiments. A gene network is fully specified upon determining the number of nodes (gene states), and the strength of their connections (RNA transcriptional rates in each state, the transition rates between gene states distinguished by differing RNA production rate, and a global RNA degradation rate); see Fig 1A.



**Fig 1. Two examples of reaction networks.** In both networks, straight arrows illustrate possible transitions between network states. Other arrows depict additional biochemical reactions such as production rates (with rate  $\beta$ ) or degradation rates (with rate  $\gamma$ ). For example, **a**) depicts a two state gene network. Grey circles depict RNA production states, differentiated by their different production rates [36]. Alternatively, **b**) depicts enzyme catalysis under allosteric enzyme control [37]. Production states represent the enzyme with or without a bound repressor. Both are outlined in greater detail in section 1.1 in [S1 Text](#).

<https://doi.org/10.1371/journal.pcbi.1011256.g001>

We will elaborate on the precise interpretation of each parameter in [Fig 1](#)'s caption, as well as describe rate interpretation for an alternative system modeled by a precisely the same biochemical reaction network described here in section 1.1 in [S1 Text](#).

### 2.2 Data generation

An  $N$  state gene network is completely specified by its gene states  $\sigma_1, \dots, \sigma_N$ , their various RNA transcription rates  $\vec{\beta} = (\beta_1, \beta_2, \dots, \beta_N)$ , rates of gene state transitions  $\vec{k} = (k_{\sigma_l \rightarrow \sigma_{l'}})$  for  $l = 1, \dots, N, l' = 1, \dots, N$  with  $l \neq l'$  and the overall RNA degradation rate  $\gamma$ . We refer to all dynamical rates compactly as  $\theta = (\vec{k}, \vec{\beta}, \gamma)$ . With these parameters at hand, we simulate Gillespie trajectories with duration  $t_{1,K}$  and record their final state  $m_{t_{1,K}}^i$ . To generate a full set of observations we then repeat this procedure  $J_k$  times for each  $t_k$ . In this manner, we simulate

snapshot data mimicking smFISH experiments [3, 24–26, 38–40] by collecting simulated RNA count data from  $J_k$  cells at all time points  $t_k$ , and a full set of data is  $\mathcal{D} = \bar{m} = \{ \{ m_{t_k}^j \}_{j=1:J_k} \}_{k=1:K}$ .

### 2.3 The inference problem

Now given data  $\mathcal{D} = \bar{m}$ , our goal is to estimate the model (in the smFISH case, determined by the number of gene states  $N$ ) and its parameters  $\theta$ . We accomplish this within a Bayesian non-parametric framework by drawing samples from the posterior,  $P(N, \theta | \bar{m})$ . By Bayes' theorem this posterior is given by

$$P(N, \theta | \bar{m}) \propto P(\bar{m} | N, \theta) P(N, \theta). \tag{1}$$

We therefore begin constructing the posterior by outlining calculation of the likelihood  $P(\bar{m} | N, \theta)$ , followed by outlining our chosen priors  $P(N, \theta)$  and finally describing our sampling method.

#### 2.3.1 Likelihood. The Chemical Master Equation (CME)

$$\frac{d}{dt} \mathbf{P}(t | \theta) = \mathbf{A} \mathbf{P}(t | \theta) \tag{2}$$

is specified for a given model by the structure of the generator matrix  $\mathbf{A}$  (in order maintain our focus on the likelihood's structure, we give the exact form of  $\mathbf{A}$  later in section 1.2 in S1 Text) and an initial probability vector over system states  $\mathbf{P}(t = 0 | \theta)$ . The generator matrix dictates the reaction network and informs the time-varying probability vector  $\mathbf{P}(t | \theta)$ , whose elements are probabilities over each possible state of the network.

Due to the nature of the transcription induction experiments whose output we wish to simulate [3,25,26], we assume that all cells are initially in the same gene state  $\sigma^*$  with zero RNA and subsequently learn  $\sigma^*$ . However, in general,  $P(t = 0 | \theta)$  can be specified as any probability vector, depending upon the initial experimental conditions at hand.

As we are more concerned here with sampling methods than the specific dynamics of any one reaction network, we restrict ourselves to a general discussion of the likelihood, though we enumerate  $\mathbf{A}$ 's elements for each model of interest in section 1.2 in S1 Text.

Given the CME's solution, the likelihood contribution for a set  $\bar{m}_k$  of snapshot data arising at time  $t_k$  is formed from the independent product from cell  $j = 1$  to cell  $J_k$  over cells and then product over times  $t_k$  from  $k = 1$  to  $K$  as follows:

$$P(\bar{m} | N, \theta) = \prod_{k=1}^K \prod_{j=1}^{J_k} \left( \sum_{s_k^j = \sigma_1, \dots, \sigma_N} P_{\theta}^{t_k} (s_k^j, m_k^j) \right). \tag{3}$$

**2.3.2 Priors on network nodes.** With the likelihood at hand, we can construct our posterior probability distribution,  $P(\theta | \bar{m})$ , by first specifying an independent prior on each parameter. Due to the possible scale separation in the rates, we consider identical independent log-normal priors on each element of  $\theta$ . Since, in general, rates are quite distinct from one another, identical independent priors on each rate help ensure that recovered ground-truth rates in simulated data analysis do not result from tuning the prior. We select identical numerical values for the hyper-parameters for all priors (they are shown in table A in S1 Text). The reason the priors themselves have limited impact on our rate posteriors is for the simple reason that data is abundant, *i.e.*, the log-posterior contains one log-prior term and  $\sum_k J_k$  additive log terms in the likelihood.

Alongside the rates, we can learn the number of states by introducing a number ( $L$ ) of hypothetical states  $\sigma_1, \dots, \sigma_L$  to the model, and assigning two parameters to each state. The first is a Bernoulli distributed (binary) parameter  $b_l$  (called a ‘load’) which indicates whether a given state is necessary. When  $b_l = 1$ ,  $\sigma_l$  contributes to the generator matrix as described in section 1.2 in [S1 Text](#). When  $b_l = 0$ , we skip over  $\sigma_l$  when constructing the generator matrix, eliminating its contribution to the likelihood. Second, a real hyper-parameter  $q_l \in (0, 1)$  (called a ‘success probability’) gives the *a priori* probability of sampling  $b_l = 1$ . Taken together, this scheme is called a Beta-Bernoulli process prior [29, 30], and its equations are as follows:

$$q_\ell \sim \text{Beta}\left(\frac{\zeta}{L}, \frac{L-1}{L}\right)$$

$$b_\ell | q_\ell \sim \text{Bernoulli}(q_\ell)$$

for  $l = 1, 2, \dots, L$  with  $\bar{b} = (b_1, b_2, \dots, b_L)$  and  $\bar{q} = (q_1, q_2, \dots, q_L)$ .

Notably, describing each model by its number of states  $N$  as above simply requires assigning  $N = \sum_{\ell=1}^L b_\ell$ .

In this manner, by iteratively sampling  $\bar{b}$ ,  $\bar{q}$  and  $\theta$  using a Gibbs sampling scheme [41], we sample the appropriate number of states attributed to the gene network under observation at each iteration, rigorously overcoming the overfitting problem [42, 43]. Vitaly, from this Gibbs scheme, we sample the number of states and all parameters from the **fully joint posterior over rates and models simultaneously, conditioned on the data**.

**2.3.3 Sampling from conditional posterior distributions.** The Gibbs sampling scheme to sample the posterior on each parameter (*i.e.*, all elements  $\theta, \bar{b}, \bar{q}$ ) divides each MCMC iteration into steps wherein a parameter or set of parameters are drawn from their respective conditional posteriors. Once each parameter ( $\theta, \bar{b}, \bar{q}$ ) has been sampled from its conditional posterior, a single MCMC iteration is complete [42, 43]. We now outline the conditional posteriors required in each step of the Gibbs scheme as follows:

1. **Loads ( $\bar{b}$ )** Each load is sampled directly from its Bernoulli posterior, conditioned upon the most recent estimates of the remaining parameters (including the remaining loads,  $\bar{b} \setminus b_\ell$  when sampling  $b_\ell$ ) and on the data  $\mathcal{D}$

$$P(b_\ell | \bar{b} \setminus b_\ell, \bar{q}, \sigma^*, \theta, \mathcal{D}) \propto P(\mathcal{D} | \bar{q}, \theta, \bar{b}) P(b_\ell | \bar{b} \setminus b_\ell, \bar{q}, \sigma^*) \quad (4)$$

where the normalization constant of the above follows from

$$P(b_\ell = 0 | \bar{b} \setminus b_\ell, \bar{q}, \sigma^*, \theta, \mathcal{D}) + P(b_\ell = 1 | \bar{b} \setminus b_\ell, \bar{q}, \sigma^*, \theta, \mathcal{D}) = 1.$$

2. **Success Probabilities ( $\bar{q}$ )** Success probabilities are sampled using Metropolis-Hastings with Beta distributed proposals [31, 44–47].
3. **Initial Condition Gene State ( $\sigma^*$ )** The initial condition gene state is sampled directly from its categorical posterior.
4. **Rates ( $\theta$ )** Rate quantification is the most difficult Gibbs step (meaning that rates take the largest number of samples to reach convergence to ground truth). As such, rate quantification conducted using Metropolis sampling is expanded upon below.

**2.3.4 Parallel tempering.** PT is a scheme wherein multiple MCMC chains draw samples from posterior probability distributions,

$$P(\theta|\mathcal{D})_h = (P(\mathcal{D}|\theta)P(\theta))^{\omega_h} \tag{5}$$

where the temperature parameter for the  $h$ th chain,  $\omega_h = 1/T_h$ , is dictated by its temperature,  $T_h$ , for  $h = 1, 2, \dots, H$  [48, 49]. Temperatures monotonically increase for each chain such that  $T_1 = 1 < T_2 < \dots < T_{h-1} < T_h$ . In this way, the posterior probability distribution associated to the first chain is the target of inference, while the higher temperature chains merely enable movement of this principle chain around the space of models. Chains sampled from posterior probability distributions at higher temperatures explore the model space more easily due to the effective ‘flattening’ arising from higher temperatures.

We initialize each chain at random from its prior distribution. Then, each chain conducts a preset number of sampling iterations from the respective tempered posterior probability distributions [50, 51] as shown in 5. Then the final parameter set for each chain, following each round of sampling iterations, is probabilistically ‘swapped’ with another temperature using an MH variant with deterministic proposals. Concretely, if we propose a swap after a fixed number of iterations between the parameter sets from chain  $a$ ,  $\theta_a$ , and from chain  $b$ ,  $\theta_b$ , with  $\omega_a > \omega_b$ , the proposal distribution is

$$Q_{\theta_a^{old}, \theta_b^{old}}(\theta_a^{prop}, \theta_b^{prop}) = \delta_{\theta_a^{old}}(\theta_b^{prop}) \delta_{\theta_b^{old}}(\theta_a^{prop}), \tag{6}$$

where  $\delta_{\theta_i}(\theta_j)$  is the Kronecker delta function equal to 1 if all corresponding elements of  $\theta_i$  and  $\theta_j$  match and 0 otherwise, and the probability of accepting the swap is

$$R_{\theta_a^{old}, \theta_b^{old}}(\theta_a^{prop}, \theta_b^{prop}) = \min\left(1, e^{(\omega_a - \omega_b)(P(\theta_a|\mathcal{D}) - P(\theta_b|\mathcal{D}))}\right). \tag{7}$$

The pairing of chains is done at random for each round maximizing the mixing amongst chains. Additionally, we optimize the swapping of chains by employing an adaptive parallel tempering algorithm for which we tune the temperature of each chain, except for the lowest temperature, according to the scheme from [52].

**2.3.5 Rate inference.** Now that we have discussed the core Gibbs sampling scheme and the overall PT, we discuss how to evolve rates for chains at each temperature.

To obtain Metropolis-Hastings proposals for rates, our method supplements a traditional Adaptive Metropolis-Hastings (AMH) sampling scheme with the intermittent use of Hamiltonian Monte Carlo (HMC) sampling [53, 54]. Our AMH algorithm follows the adaptive scheme described in [55]. Briefly, each parameter set  $\theta$  is updated subject to a Normal proposal distribution with an adaptable covariance matrix,  $\Sigma$ :

$$Q_{\theta^{old}}(\theta^{prop}) = \mathbf{Normal}(\theta_i^{prop}, \theta_i^{old}, \Sigma) \tag{8}$$

where  $\theta_i^{old}$  is the set of parameters associated to the previous iteration, and  $\theta^{prop}$  is the proposed set of parameters. Once a proposal has been made, the new set is either accepted with probability

$$R_{\theta^{old}}(\theta^{prop}) = \min\left(1, \frac{P(\mathcal{D}|\theta^{prop}) P(\theta^{prop}) Q_{\theta^{prop}}(\theta^{old})}{P(\mathcal{D}|\theta^{old}) P(\theta^{old}) Q_{\theta^{old}}(\theta^{prop})}\right), \tag{9}$$

or rejected with the complementary probability  $(1 - R_{\theta^{old}}(\theta^{prop}))$ , and the algorithm repeats in an iterative fashion. The adaptation portion of the algorithm uses the movement of each parameter up to the current iteration to tune the covariance matrix,  $\Sigma$ . Concretely, an element

of the covariance matrix,  $\Sigma_{ij}$ , is specified as the covariance between parameters  $i$  and parameter  $j$  resulting from the previous sample sequences of both parameters. Then, in order to guarantee that the resulting matrix is a valid parameter for the multivariate normal proposal distribution, we add a small fixed multiple ( $\epsilon = 3.7508 \times 10^{-17}$ ) of the identity, to guarantee that the proposal covariance can be inverted when sampling [55, 56].

While this adaptive scheme incorporates correlations between parameters into the proposal covariance matrix, it fails to explicitly leverage the shape of the posterior probability distribution to inform proposals. Therefore, the HMC proposal method is introduced [57], to which AMH can be seen as a less effective, albeit less expensive analogue.

HMC generates proposals by reversibly evolving parameters according to pseudo-Hamiltonian dynamics [57]. We start with our target distribution

$$\mathbb{P}(\theta|\mathbf{data}) = \int d\mathbf{p} \mathbb{P}(\theta, \mathbf{p}|\mathbf{data}) \quad (10)$$

where the auxiliary variables  $p$  are introduced allowing us to write

$$\mathbb{P}(\theta, \mathbf{p}|\mathbf{data}) = \exp(-H(\theta, \mathbf{p})) \quad (11)$$

thereby interpreting the log-posterior as the Hamiltonian

$$H(\theta, \mathbf{p}) = -\log(\mathbb{P}(\theta, \mathbf{p}|\mathbf{data})) \quad (12)$$

where

$$H(\theta, \mathbf{p}) = L(\theta) + V(\theta) + T(\mathbf{p}). \quad (13)$$

Here  $\mathbf{p} = (p_1, p_2, \dots, p_r)$  are the momenta, each corresponding to a parameter of interest, up to  $r$  parameters. In order to initialize the dynamics, we choose the value of these momenta at random from a standard normal distribution at the start of each HMC iteration. The pieces of the Hamiltonian function include the negative logarithm of the prior,  $L(\theta) = -\log(P(\theta))$ , the negative logarithm of the likelihood,  $V(\theta) = -\log(P(\mathcal{D}|\theta))$ , and the kinetic energy given by,

$$T(\mathbf{p}) = \frac{\mathbf{p}W^{-1}\mathbf{p}}{2}, \quad (14)$$

where  $W$  is a square mass matrix of size  $r \times r$ .

To integrate the resulting Hamiltonian dynamics for analytical likelihoods we may use Stormer-Verlet (*i.e.*, plain leapfrog) [58, 59] which is the symplectic integrator of choice. However, to treat our non-analytic likelihoods, we resort to operator splitting, Strang-splitting [60].

In Strang-splitting, the Hamiltonian function is split into two components

$$H(\theta, \mathbf{p}) = H^1(\theta, \mathbf{p}) + H^2(\theta, \mathbf{p}), \quad (15)$$

where

$$H^1(\theta, \mathbf{p}) = V(\theta), \quad (16)$$

and

$$H^2(\theta, \mathbf{p}) = L(\theta) + T(\mathbf{p}). \quad (17)$$

With these two components, we apply the following three steps for each proposal within the HMC algorithm: 1) we simulate dynamics (integrate the Hamilton-Jacobi equations) within the parameter space for half a time-step according to  $H^1(\theta, \mathbf{p})$ ; 2) we advance a whole time-step according to  $H^2(\theta, \mathbf{p})$ ; and 3) we repeat the first step. By repeating these steps for a

predetermined amount of time, we obtain a set of proposed parameter values,  $\theta^{prop}$ , and corresponding auxiliary variables,  $\mathbf{p}^{prop}$ , accepted or rejected according to a Metropolis-Hastings acceptance ratio, when compared to the set of parameters from the first step of the HMC,  $\theta^{old}$  and variables  $\mathbf{p}^{old}$ . The acceptance probability reads

$$R_{\theta^{old}}(\theta^{prop}) = \min\left(1, \frac{e^{-H(\theta^{prop}, \mathbf{p}^{prop})}}{e^{-H(\theta^{old}, \mathbf{p}^{old})}}\right). \quad (18)$$

Employing a combination of all three sampling methods discussed above, AMH, HMC, and PT, within our algorithm allows us to maximize the effectiveness of our sampling, enabling model inference on high dimensional, complicated biological systems from snapshot data without imposing *a priori* assumptions on the underlying system's dynamics.

### 3 Results

Here for one and two state networks we benchmark alternative versions of our method: 1) the “full method”—our method using advanced proposal schemes (AMH and HMC) within the PT framework; 2) the “fixed proposals”—a version which generates proposals from a fixed proposal distribution (with no AMH and HMC) within the PT framework; 3) the “without PT”—a version containing advanced proposal schemes (AMH and HMC), sampling only a single Markov chain.

As we will show, even the full method may be inefficient for three state networks and beyond. Thus for the three state network, we contrast: 1) the hybrid “full method”, as before; and 2) the “full method +  $\gamma$ ”—the full method, assisted by hand-specification of the product degradation rate,  $\gamma$ .

We demonstrate the sampling efficiency of each method by considering alternative methods along three dimensions: 1) accuracy of learned posteriors (that is, how close the learned posterior's mode is to the ground truth) following a fixed number of samples; 2) duration of burn-in (the number of iterations to convergence); and 3) number of independent samples drawn as determined by the autocorrelation between successive samples. Here, we highlight selected results for the one, then two, and finally three state network from the analysis of synthetic snapshot data (for concreteness of the example data, chosen to replicate smFISH), and full results (plots containing information on all learned rates) are contained in section 1.3 in [S1 Text](#).

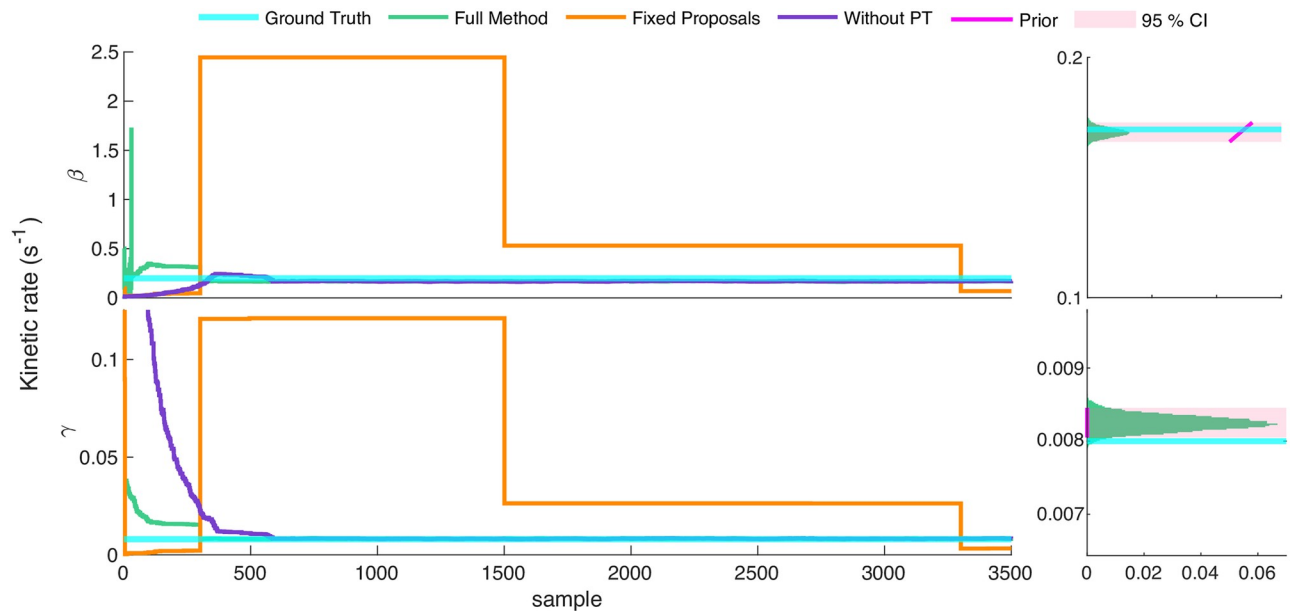
We will find across all examples that the full method outperforms each of the opposing methods. Additionally, we observe that the stiffness of the CME renders a fixed proposal scheme infeasible even for the simplest case. Our advanced proposal schemes overcome the challenge encountered with fixed proposals by more frequently proposing parameters maintaining the stability of the eventual CME's solution. Though it doesn't prove vital in all cases, we repeatedly find that, since PT improves our method's mixing by reducing the separation between the isolated modes of our posterior [53], it leads to faster convergence.

In all of our computational experiments, we will also find that AMH and HMC improve sampling efficiency for the reasons discussed in methods.

#### 3.1 One state network

For ease of interpretation alone, here we show results of the parametric method (*i.e.*, fixing, as opposed to learning, the number of states). [Fig 2](#) depicts traces of rates across MCMC samples converging in this case.

[Fig 2](#) clearly shows the necessity of informed proposals for parameter inference on the one state network. Initializing from an uninformative prior, the full method converges to the



**Fig 2. Inference strategies for the one state network.** Here we show traces of MCMC samples drawn from the target distribution depicting the convergence of three methods outlined above. Clearly, the fixed proposal scheme (orange trace) demonstrates poor mixing for both parameters ( $\beta$ , production, and  $\gamma$ , degradation, rates). That is, it fails to propose samples which approach ground truth with as many or fewer samples as other methods. The performance difference between the full method (green trace) and the method without parallel tempering (purple trace) is less apparent, but still present for this example. To the right of each rate's MCMC trace, we plot a histogram of the posterior for the corresponding parameter, complete with the ground truth (cyan line), prior density (magenta curve) and 95% confidence interval (magenta shaded region).

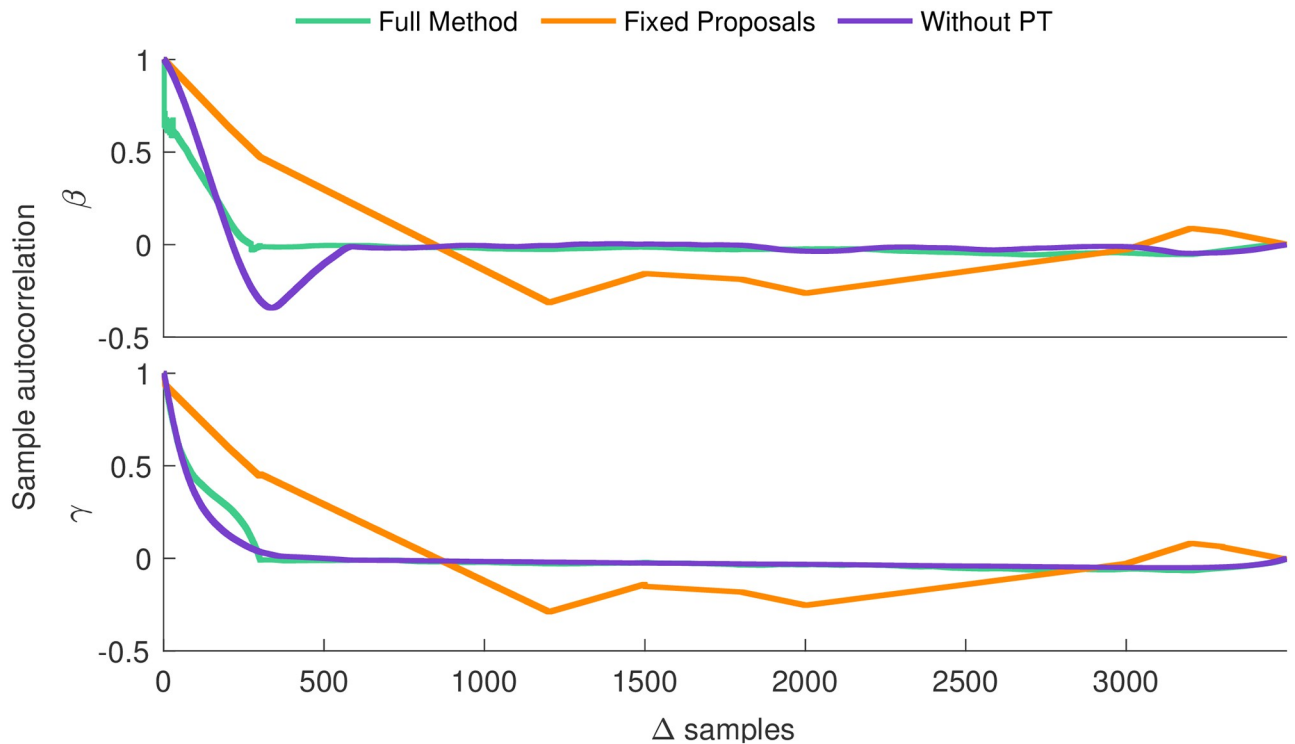
<https://doi.org/10.1371/journal.pcbi.1011256.g002>

ground truth first (within 500 iterations), while the method neglecting parallel tempering requires approximately 200 more iterations for convergence. Meanwhile, the method relying on a fixed distribution to generate proposals does not visit the ground truth rates within the number of samples allotted. Therefore though adaptive proposals are necessary for the one state network, PT is not entirely necessary to reach convergence in a reasonable number of samples, even if it does improve the speed of convergence.

The number of independent samples in an MCMC chain determines the fidelity of a Bayesian MCMC algorithm's empirical characterization of its target distribution. Therefore lower autocorrelation (more **independence**) between samples allows characterization of the posterior in fewer MCMC samples. As indicated by Fig 3, the full method demonstrates a near immediate reduction in the sample autocorrelation for all parameters. Meanwhile, the method with uninformed proposals frequently rejects proposed models, and as a result of the negligible movement of the MCMC chain, fails to clearly indicate that successive samples are independent even over the entire duration of inference. Thus, the advanced proposal mechanisms improve efficiency by increasing the proportion of accepted samples.

Fig 4 shows that as each method approaches the ground truth network, its log-posterior increases. Predictably since it finds the ground truth network rates fastest, the full method's log-posterior reaches its maximum first, and remains on its eventual 'plateau' for the highest number of MCMC iterations. In exact correspondence to Figs 2 and 3, the next fastest to converge is the method without PT, with the method with fixed proposals following.

Notably, a large decrease in the log-posterior occurs for the method with fixed proposals. This behavior is the result of repeatedly proposing improbable networks, a direct consequence of an MCMC method exploring the target distribution slowly.



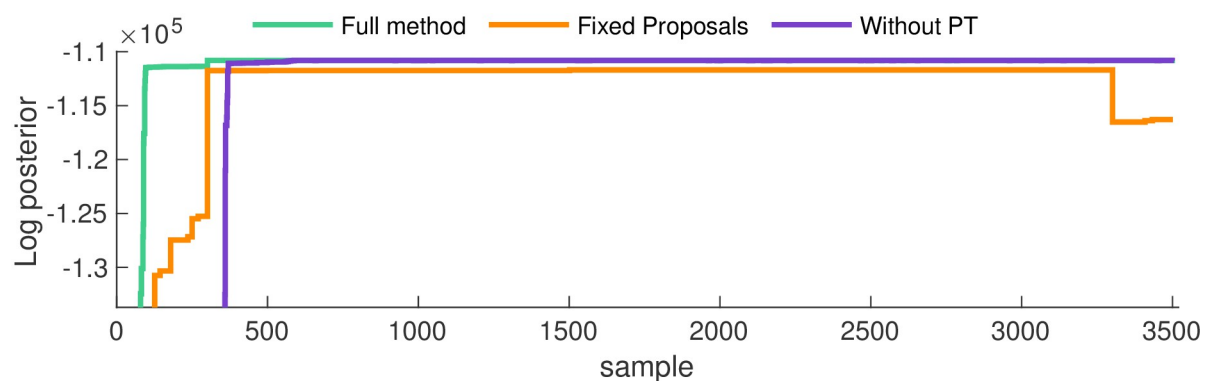
**Fig 3. Sample autocorrelation analysis for the one state network.** An auto-correlation that converges to zero fastest is ideal as it maximizes the number of independent samples. We see that the full method converges quickest.

<https://doi.org/10.1371/journal.pcbi.1011256.g003>

### 3.2 Two state network

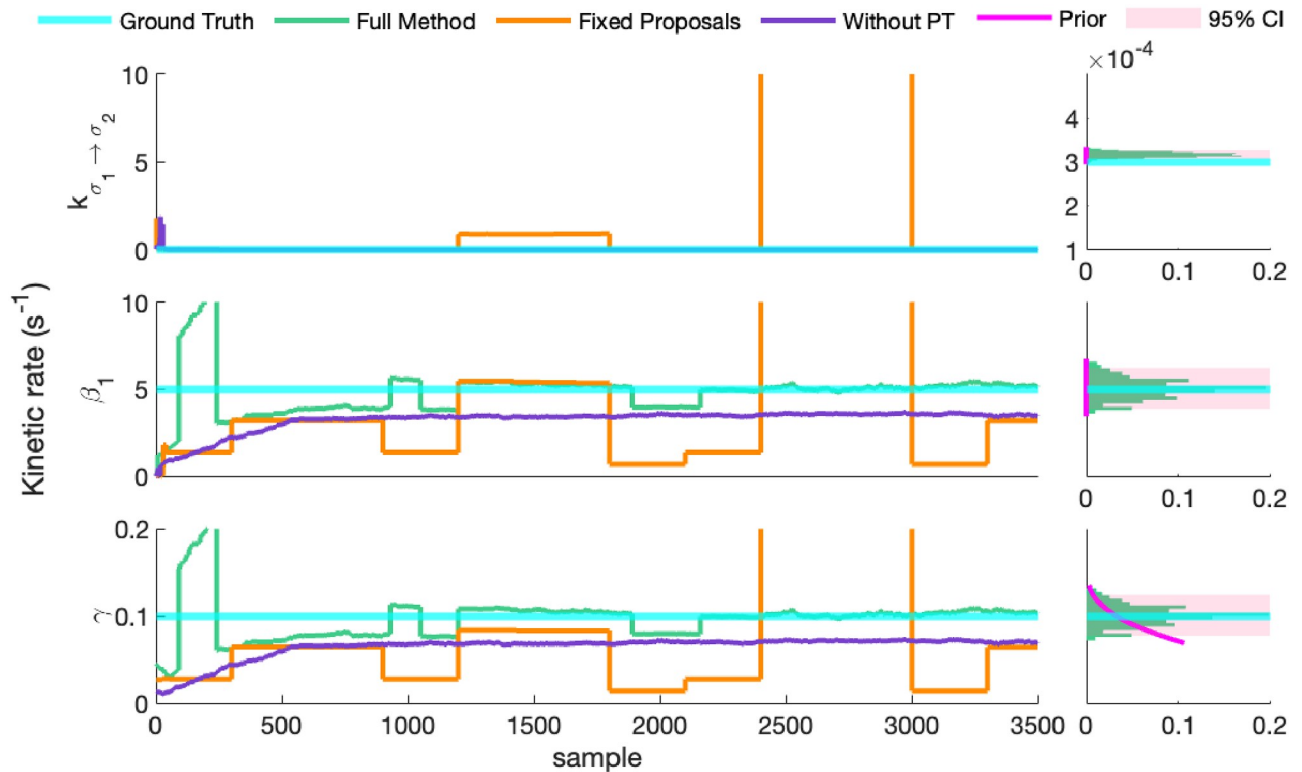
The remainder of analysis is concerned with the nonparametric inference method, learning the number of network nodes as well as rates.

Once again, Fig 5 demonstrates the necessity of the full method to accurately infer the two state network model in a reasonable number of samples. By the end of Fig 5's 3500 samples, the full method has converged to the ground truth and is exploring the neighborhood of the



**Fig 4. Convergence in the log-posterior for the one state network.** Here we compare convergence of the log-posterior for the three methods outlined above. Echoing Fig 2, the full method demonstrates the fastest convergence, followed by the method without PT, with fixed proposals lagging behind.

<https://doi.org/10.1371/journal.pcbi.1011256.g004>



**Fig 5. Inference strategies for the two state network.** Here we compare our full method (green trace) to our method with PT removed (purple trace), and our method with adaptive proposals removed (orange trace). We show only representative rates though the rest are shown alongside our posterior over the number of states in Fig A in [SI Text](#). To demonstrate that the method reliably converges to the ground truth, the second column depicts marginal rate histograms (green bars) taken from the displayed MCMC chain as well as 5 additional MCMC chains, initialized from the prior. Additionally, to illustrate the method's confidence, we depict (magenta shaded area) the region in which 95% of MCMC samples fall, and to demonstrate the distinctness of the posterior from the prior, we depict the rate prior density (magenta curve) within the region covered by the learned posterior.

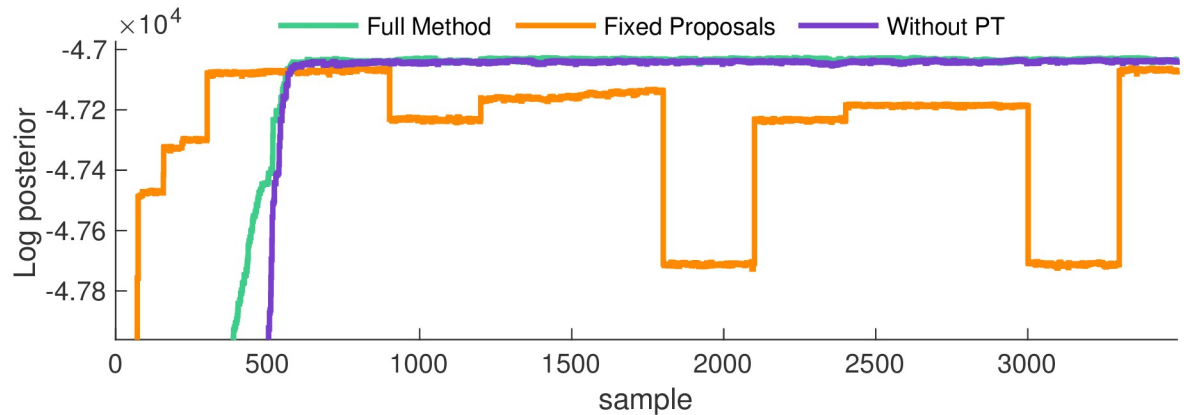
<https://doi.org/10.1371/journal.pcbi.1011256.g005>

ground truth, while the alternative methods lag behind. As we have seen earlier in our discussion of section 3.1, owing to its ability to swap chains (thereby effectively “jumping” around the space of models, potentially between isolated modes of the target distribution), the method which uses PT without advanced proposal mechanisms explores a broader model diversity than that without PT. In fact, the utility of the PT scheme is on full display in [Fig 5](#), as the method without PT demonstrates trapping in a local posterior maximum that, even with AMH and HMC, requires PT to escape.

Analysis of the log-posterior clearly shows the local trapping of the method without PT on one of the target's modes. Its log-posterior, the purple trace in [Fig 6](#), flattens around iteration number 600 (notably, almost the exact same iteration as the full method), but it remains at values bested only slightly by the full method for the same reasons identified in section 3.1. This indicates that even though the method without PT “converges”, the quality of convergence is inferior to the full method, as shown in [Fig 5](#).

### 3.3 Three state network

For the three state network, the dimensionality of the model space causes two problems: 1) the number of plausible candidate models is high, increasing the possibility for local trapping of the sampler; 2) owing to its high dimensionality, the model space is inherently difficult to



**Fig 6. Convergence in the log-posterior for the two state network.** Here we compare increases in the log-posterior for the three methods discussed so far. Echoing Fig 2, the full method demonstrates the fastest convergence, followed by the method without PT, with fixed proposals lagging behind. Examining the purple trace, the method without PT apparently converges but, due to local maximization, never reaches the ground truth parameters. We show autocorrelations for only the rates highlighted here, but the rest are shown in Fig B in S1 Text.

<https://doi.org/10.1371/journal.pcbi.1011256.g006>

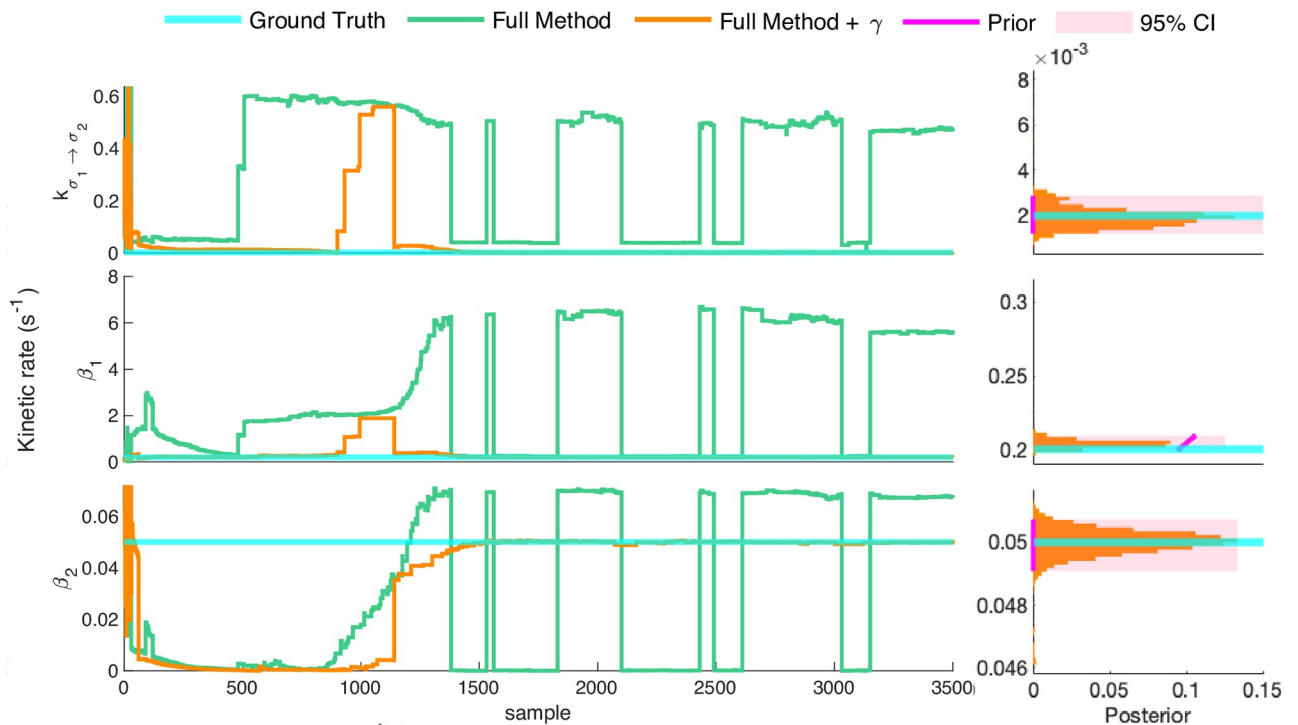
explore. The influence of these issues is clear in Fig 7, where the full method fails to reach the ground truth rates after 3500 samples, since the snapshot data's scant dynamical information results in a tendency toward trapping in local posterior maxima. Given enough time, the data is often sufficient to determine the relative strength of multiple local maxima using the full method and recover the ground truth for all parameters simultaneously (see supplemental figure 1 of [8]). We improve computational efficiency, we may reduce model dimensionality by fixing one or more parameters. Owing to its "global" role in the reaction network the degradation rate ( $\gamma$ ) is a natural choice. In part, this is because production rate ( $\beta$ ) can be reduced to compensate for a lower assumed degradation rate giving rise to approximate model indeterminacy.

Figs 7 and E in S1 Text demonstrate how specifying  $\gamma$  results in timely convergence of our nonparametric model inference method. This is further supported by Fig D in S1 Text which shows reduced autocorrelations, and thus more efficient sampling, under specification of  $\gamma$ .

## 4 Discussion

We have systematically characterized the performance of various different methods to learn the number of nodes and connectivities of reaction networks. For the purpose of analysis, we've used synthetic data inspired by smFISH, an essential tool in the study of gene transcription dynamics [61, 62]. Previously, attempts to quantify transcription dynamics have been hindered by two issues: 1) prohibitively costly likelihood calculations, and high dimensional parameter spaces; as well as 2) highly correlated and scale-separated parameters. Filling this gap, our method can learn models quickly and precisely. Owing to the issues we've identified, previous methods have resorted to assuming *a priori* that certain reaction pathways do not occur (e.g., assuming that gene states are linearly or cyclically connected) [26, 36].

We improve upon the limitations of previous rate specification tools [3, 25, 25, 28, 34, 63] by considering advantages and disadvantages of combinations of HMC, PT, and AMH to a custom Gibbs sampling scheme. We utilize the resulting increased efficiency to sample complex posteriors inherent when simultaneously sampling networks and parameters within a Bayesian nonparametric paradigm.



**Fig 7. Nonparametric inference strategies for the three state network.** Here we show traces depicting the convergence of two methods outlined above. Clearly, the full method (green trace) is insufficient, transitioning between local maxima rather than converging to the ground truth. By contrast, addition of a pre-calibrated degradation rate permits accurate inference in a timely manner. As before, we depict the 95% confidence interval (magenta shaded), and we depict the rate prior density (magenta curve) within the region covered by the learned posterior. What is more, we show results for all remaining rates alongside our posterior over the number of states in Fig C in [S1 Text](#).

<https://doi.org/10.1371/journal.pcbi.1011256.g007>

As it stands, our method represents a distinct advancement in MCMC to quantify reaction networks. Notably, this advancement comes from combining previously-developed samplers albeit on a novel nonparametric framework requiring simultaneous inference of discrete and continuous parameters, rather than necessarily providing a distinct innovation in any one sampling technique. While we have explored three samplers, a number of other samplers exist (such as affine invariant ensemble samplers [64, 65], and pre-conditioned Monte Carlo [66]), though these may break down for high-dimensional [67] or non-convex [68] posteriors, respectively.

When HMC, AMH and PT are used in concert, these tools consistently provide a significant reduction in the number of iterations to reach convergence, compared to the same method with any of the three tools removed. Additionally analysis of sample auto-correlation (Fig 3, and Figs B and D in [S1 Text](#)) demonstrates our full method's lower sample auto-correlation as compared to its alternatives. This result indicates that, in addition to converging in fewer samples, the number of independent samples per MCMC sample is greatest for the full method.

By further investigating limitations of advanced samplers, our tools allow us to set limits on the complexity of reaction networks that we can realistically learn from typical finite snapshot data set sizes before encountering severe albeit approximate model indeterminacy. For example in the three state gene network case, simply introducing additional data on RNA degradation into a joint likelihood with our RNA snapshot data partially resolves indeterminacy. Critically however, our method allows us to increase the amount of data considered to discover more complex networks while avoiding overfitting.

Although we developed our method with non-equilibrium data in mind, we can readily model measurements at equilibrium whose likelihood follows from a reaction network's steady-state CME solution. In this case, the finite state projection (FSP) we use to compute likelihoods [69] may require, as an approximation, the introduction of an absorbing state [70]. Barring this modification to the generator matrix, sampling methods compared here treat equilibrium and non-equilibrium systems in precisely the same manner and their performance is not affected by the interpretation of the states at hand.

To our already general method, some additional generalizations are possible at little to no additional cost. For instance, we make the simplifying assumption here that data arrives uncorrupted by measurement noise. Adding any arbitrary emission distribution to our model merely entails the addition of a distribution over different underlying RNA counts and marginalization over these counts [8]. Along these same lines, the initial condition used to solve the CME can be modified to suit any experimentally relevant initial condition.

Moving forward, provided appropriate data, our method lends itself to simple adaptation to infer alternative reaction networks, not necessarily composed of gene states with associated RNA production, by simply modifying the structure of the generator matrix (see section 1.2 in [S1 Text](#)).

To help further improve computational efficiency, we may use genetic algorithms [25] to initiate our MCMC chain closer to the global posterior maximum than what is currently done by initializing samples from the prior. In this manner, burn-in time might be reduced, though local-maximum trapping remains unavoidable. We exclude this technique here for two reasons: 1) the genetic algorithm employed in [25] requires pre-specification of the number of gene states, a limitation that our method overcomes; 2) analyzing the characteristics and duration of burn-in are of primary interest here.

While introducing additional cost, we can extend our framework to consider more complex reaction networks. First, we may relax our assumption that only a single transcribing copy of each gene exists in every cell, and instead we may learn a distribution over the number of genes in each cell in the case genes appear on plasmids. Additionally, time-varying rates of production, state transitions, or degradation, [3, 26, 71] are readily added to the method, at the modest cost of introducing time-dependency to the generator matrix. The generator matrix may be further expanded to consider RNA transport across spatial compartments such as the nucleus or cytoplasm [3]. Finally, as the number of quantified genes increases using multiplexed smFISH methods [72], with some modification, our approach allows for models of co-varying gene expression networks [62, 72, 73].

Each generalization mentioned above introduces complexity to the likelihood's computation arising from the increase in the state number and complexity of the connectivity map. Obtaining computationally efficient CME solutions in deriving the likelihood, the key computational bottleneck, is therefore critical in implementing these generalizations. Currently, the time cost of the likelihood computation scales roughly linear with  $\mathbf{A}$ 's dimension, with evidence that alternative methods to integrate the CME—*i.e.*, FSP based Krylov subspace methods [74, 75]—may be faster than the CME solution method used here. How the computational cost of computing  $\mathbf{A}$ 's CME solution scales with its sparsity is more complex. In the case of densely connected reaction networks, the recently proposed Quantized Tensor Train method [76] may be more efficient than the FSP-based Krylov subspace approach which uses incremental time stepping rather than jumping immediately to the times desired for analysis. Alternatively, there have been promising attempts to solve ODEs using neural networks [77] or to parallelize matrix exponentiation using GPU hardware [78]. In addition to facilitating the difficulties arising from dense CME generator matrix exponentiation, neural network approaches may further enable parameter inference models when non-Markovian dynamics are present

[79]. These tools may eventually become important as we move to monitoring dynamics in real time and deal with, for instance, finite time fluorescent protein maturation introducing delays between gene expression and fluorescence reporter detections [80].

## Supporting information

**S1 Text.** Supporting information file containing Supporting Figures (Fig A—E) and Supporting Table (Table A). Fig A. Two state nonparametric network inference strategies. Fig B. Two state nonparametric network inference autocorrelations. Fig C. Three state nonparametric network inference strategies. Fig D. Sample autocorrelation analysis of three state network. Fig E. Convergence in the log-posterior for the three state network. Table A. Table of symbol names and (where applicable) their numerical values.

(PDF)

## Acknowledgments

We thank Prof. Ioannis Sgouralis, Prof. Douglas Shepherd and Dr. Zachary Fox for interesting discussions and insights.

## Author Contributions

**Conceptualization:** Zeliha Kilic, Max Schweiger, Camille Moyer, Steve Pressé.

**Data curation:** Max Schweiger, Steve Pressé.

**Formal analysis:** Zeliha Kilic, Max Schweiger, Steve Pressé.

**Funding acquisition:** Steve Pressé.

**Investigation:** Zeliha Kilic, Max Schweiger, Camille Moyer, Steve Pressé.

**Methodology:** Zeliha Kilic, Max Schweiger, Camille Moyer, Steve Pressé.

**Project administration:** Steve Pressé.

**Resources:** Steve Pressé.

**Software:** Zeliha Kilic, Max Schweiger, Camille Moyer.

**Supervision:** Zeliha Kilic, Steve Pressé.

**Validation:** Max Schweiger.

**Visualization:** Max Schweiger.

**Writing – original draft:** Zeliha Kilic, Max Schweiger, Camille Moyer.

**Writing – review & editing:** Zeliha Kilic, Max Schweiger, Steve Pressé.

## References

1. Hung KYS, Klumpe S, Eisele MR, Elsasser S, Tian G, Sun S, et al. Allosteric control of Ubp6 and the proteasome via a bidirectional switch. *Nature communications*. 2022; 13(1):1–13. <https://doi.org/10.1038/s41467-022-28186-y>
2. Fletcher A, Zhao R, Enciso G. Non-cooperative mechanism for bounded and ultrasensitive chromatin remodeling. *Journal of Theoretical Biology*. 2022; 534:110946. <https://doi.org/10.1016/j.jtbi.2021.110946> PMID: 34717936
3. Munsky B, Li G, Fox ZR, Shepherd DP, Neuert G. Distribution shapes govern the discovery of predictive models for gene regulation. *Proceedings of the National Academy of Sciences*. 2018; 115:7533–7538. <https://doi.org/10.1073/pnas.1804060115> PMID: 29959206

4. Shen X, Wang R, Xiong X, Yin Y, Cai Y, Ma Z, et al. Metabolic reaction network-based recursive metabolite annotation for untargeted metabolomics. *Nature communications*. 2019; 10(1):1–14. <https://doi.org/10.1038/s41467-019-09550-x> PMID: 30944337
5. Gatto F, Ferreira R, Nielsen J. Pan-cancer analysis of the metabolic reaction network. *Metabolic engineering*. 2020; 57:51–62. <https://doi.org/10.1016/j.ymben.2019.09.006> PMID: 31526853
6. Liu B, Mavrova SN, van den Berg J, Kristensen SK, Mantovanelli L, Veenhoff LM, et al. Influence of fluorescent protein maturation on FRET measurements in living cells. *ACS sensors*. 2018; 3(9):1735–1742. <https://doi.org/10.1021/acssensors.8b00473> PMID: 30168711
7. Morisaki T, Lyon K, DeLuca KF, DeLuca JG, English BP, Zhang Z, et al. Real-time quantification of single RNA translation dynamics in living cells. *Science*. 2016; 352(6292):1425–1429. <https://doi.org/10.1126/science.aaf0899> PMID: 27313040
8. Kilic Z, Schweiger M, Moyer C, Shepherd D, Pressé S. Gene expression model inference from snapshot RNA data using Bayesian non-parametrics. *Nature Computational Science*. 2023; p. 1–10.
9. Fritsche-Guenther R, Witzel F, Sieber A, Herr R, Schmidt N, Braun S, et al. Strong negative feedback from Erk to Raf confers robustness to MAPK signalling. *Molecular systems biology*. 2011; 7(1):489. <https://doi.org/10.1038/msb.2011.27> PMID: 21613978
10. Femino AM, Fay FS, Fogarty K, Singer RH. Visualization of single RNA transcripts in situ. *Science*. 1998; 280(5363):585–590. <https://doi.org/10.1126/science.280.5363.585> PMID: 9554849
11. Marzi MJ, Ghini F, Cerruti B, De Pretis S, Bonetti P, Giacomelli C, et al. Degradation dynamics of microRNAs revealed by a novel pulse-chase approach. *Genome research*. 2016; 26(4):554–565. <https://doi.org/10.1101/gr.198788.115> PMID: 26821571
12. Tak T, Wijten P, Heeres M, Pickkers P, Scholten A, Heck AJ, et al. Human CD62Ldim neutrophils identified as a separate subset by proteome profiling and in vivo pulse-chase labeling. *Blood, The Journal of the American Society of Hematology*. 2017; 129(26):3476–3485. PMID: 28515092
13. Wang Z, Gerstein M, Snyder M. RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*. 2009; 10:57–63. <https://doi.org/10.1038/nrg2484> PMID: 19015660
14. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-Seq data with DESeq2. *Genome biology*. 2014; 15:550. <https://doi.org/10.1186/s13059-014-0550-8> PMID: 25516281
15. Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nature biotechnology*. 2011; 29:644–652. <https://doi.org/10.1038/nbt.1883> PMID: 21572440
16. Ziegenhain C, Vieth B, Parekh S, Hellmann I, Enard W. Quantitative single-cell transcriptomics. *Briefings in functional genomics*. 2018; 17:220–232. <https://doi.org/10.1093/bfpg/ely009> PMID: 29579145
17. Gaidatzis D, Burger L, Florescu M, Stadler MB. Analysis of intronic and exonic reads in RNA-Seq data characterizes transcriptional and post-transcriptional regulation. *Nature biotechnology*. 2015; 33:722–729. <https://doi.org/10.1038/nbt.3269> PMID: 26098447
18. Rahman S, Zenklusen D. Single-molecule resolution fluorescent in situ hybridization (smFISH) in the yeast *S. cerevisiae*. In: *Imaging Gene Expression*. Springer; 2013. p. 33–46.
19. Shaffer SM, Wu MT, Levesque MJ, Raj A. Turbo FISH: a method for rapid single molecule RNA FISH. *PLoS one*. 2013; 8(9):e75120. <https://doi.org/10.1371/journal.pone.0075120> PMID: 24066168
20. Asano SM, Gao R, Wassie AT, Tillberg PW, Chen F, Boyden ES. Expansion microscopy: protocols for imaging proteins and RNA in cells and tissues. *Current protocols in cell biology*. 2018; 80(1):e56. <https://doi.org/10.1002/cpcb.56> PMID: 30070431
21. Kramer A, Calderhead B, Radde N. Hamiltonian Monte Carlo methods for efficient parameter estimation in steady state dynamical systems. *BMC Bioinformatics*. 2014; 15(1):253. <https://doi.org/10.1186/1471-2105-15-253> PMID: 25066046
22. Hellander A, Lötstedt P. Hybrid method for the chemical master equation. *Journal of Computational Physics*. 2007; 227(1):100–122. <https://doi.org/10.1016/j.jcp.2007.07.020>
23. Peleš S, Munsky B, Khammash M. Reduction and solution of the chemical master equation using time scale separation and finite state projection. *The Journal of chemical physics*. 2006; 125(20):204104. <https://doi.org/10.1063/1.2397685> PMID: 17144687
24. Neuert G, Munsky B, Tan RZ, Teytelman L, Khammash M, Van Oudenaarden A. Systematic identification of signal-activated stochastic gene regulation. *Science*. 2013; 339:584–587. <https://doi.org/10.1126/science.1231456> PMID: 23372015
25. Vo HD, Fox Z, Baetica A, Munsky B. Bayesian estimation for stochastic gene expression using multifidelity models. *The Journal of Physical Chemistry B*. 2019; 123:2217–2234. <https://doi.org/10.1021/acs.jpcb.8b10946> PMID: 30777763

26. Wang M, Zhang J, Xu H, Golding I. Measuring transcription at a single gene copy reveals hidden drivers of bacterial individuality. *Nature microbiology*. 2019; 4:2118–2127. <https://doi.org/10.1038/s41564-019-0553-z> PMID: 31527794
27. Munsky B, Fox Z, Neuert G. Integrating single-molecule experiments and discrete stochastic models to understand heterogeneous gene transcription dynamics. *Methods*. 2015; 85:12–21. <https://doi.org/10.1016/j.ymeth.2015.06.009> PMID: 26079925
28. Neuert G, Munsky B, Tan RZ, Teytelman L, Khammash M, van Oudenaarden A. Systematic Identification of Signal-Activated Stochastic Gene Regulation. *Science*. 2013; 339(6119):584–587. <https://doi.org/10.1126/science.1231456> PMID: 23372015
29. Cheng Y, Li D, Jiang W. The Exact Inference of Beta Process and Beta Bernoulli Process From Finite Observations. *Computer Modeling in Engineering & Sciences*. 2019; 121:49–82. <https://doi.org/10.32604/cmescs.2019.07657>
30. Thibaux R, Jordan MI. Hierarchical Beta processes and the Indian buffet process. In: *Artificial Intelligence and Statistics*; 2007. p. 564–571.
31. Sgouralis I, Bryan JS, Presse S. Enumerating High Numbers of Fluorophores from Photobleaching Experiments: a Bayesian Nonparametrics Approach. *bioRxiv*. 2020;.
32. Tavakoli M, Jazani S, Sgouralis I, Shafraz OM, Sivasankar S, Donaphon B, et al. Pitching single-focus confocal data analysis one photon at a time with Bayesian nonparametrics. *Physical Review X*. 2020; 10(1):011021. <https://doi.org/10.1103/physrevx.10.011021> PMID: 34540355
33. Jazani S, Sgouralis I, Shafraz OM, Levitus M, Sivasankar S, Pressé S. An alternative framework for fluorescence correlation spectroscopy. *Nature communications*. 2019; 10(1):1–10. <https://doi.org/10.1038/s41467-019-11574-2> PMID: 31413259
34. Lin YT, Buchler NE. Exact and efficient hybrid Monte Carlo algorithm for accelerated Bayesian inference of gene expression models from snapshots of single-cell transcripts. *The Journal of chemical physics*. 2019; 151:024106. <https://doi.org/10.1063/1.5110503> PMID: 31301707
35. Wolff U, Collaboration A, et al. Monte Carlo errors with less errors. *Computer Physics Communications*. 2004; 156(2):143–153. [https://doi.org/10.1016/S0010-4655\(03\)00467-3](https://doi.org/10.1016/S0010-4655(03)00467-3)
36. Li G, Neuert G. Multiplex RNA single molecule FISH of inducible mRNAs in single yeast cells. *Scientific data*. 2019; 6:1–9. <https://doi.org/10.1038/s41597-019-0106-6> PMID: 31209217
37. Modi T, Ozkan SB, Pressé S. Information Propagation in Time through Allosteric Signaling. *Biophysical Journal*. 2021; 120(3):300a. <https://doi.org/10.1016/j.bpj.2020.11.1917>
38. Schuh L, Saint-Antoine M, Sanford EM, Emert BL, Singh A, Marr C, et al. Gene Networks with Transcriptional Bursting Recapitulate Rare Transient Coordinated High Expression States in Cancer. *Cell Systems*. 2020; 10:363–378.e12. <https://doi.org/10.1016/j.cels.2020.03.004> PMID: 32325034
39. Golding I, Paulsson J, Zawilski SM, Cox EC. Real-time kinetics of gene activity in individual bacteria. *Cell*. 2005; 123:1025–1036. <https://doi.org/10.1016/j.cell.2005.09.031> PMID: 16360033
40. So Lh, Ghosh A, Zong C, Sepúlveda LA, Segev R, Golding I. General properties of transcriptional time series in *Escherichia coli*. *Nature genetics*. 2011; 43:554–560. <https://doi.org/10.1038/ng.821> PMID: 21532574
41. Gelman A, Carlin JB, Stern HS, Rubin DB. *Bayesian data analysis*. Chapman and Hall/CRC; 1995.
42. Hastings WK. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*. 1970; 57(1):97–109. <https://doi.org/10.1093/biomet/57.1.97>
43. Smith AFM, Roberts GO. Bayesian Computation Via the Gibbs Sampler and Related Markov Chain Monte Carlo Methods. *J Roy Stat Soc B*. 1993; 55(1):3–23.
44. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. Equation of state calculations by fast computing machines. *The journal of chemical physics*. 1953; 21(6):1087–1092. <https://doi.org/10.1063/1.1699114>
45. Sgouralis I, Pressé S. An introduction to infinite HMMs for single-molecule data analysis. *Biophysical journal*. 2017; 112(10):2021–2029. <https://doi.org/10.1016/j.bpj.2017.04.027> PMID: 28538142
46. Sgouralis I, Pressé S. Icon: an adaptation of infinite hmms for time traces with drift. *Biophysical journal*. 2017; 112:2117–2126. <https://doi.org/10.1016/j.bpj.2017.04.009> PMID: 28538149
47. Sgouralis I, Madaan S, Djutanta F, Kha R, Hariadi RF, Pressé S. A Bayesian nonparametric approach to single molecule forster resonance energy transfer. *The Journal of Physical Chemistry B*. 2018; 123(3):675–688. <https://doi.org/10.1021/acs.jpcc.8b09752>
48. Berg BA. Introduction to Markov chain Monte Carlo simulations and their statistical analysis. *Markov Chain Monte Carlo Lect Notes Ser Inst Math Sci Natl Univ Singap*. 2005; 7:1–52. [https://doi.org/10.1142/9789812700919\\_0001](https://doi.org/10.1142/9789812700919_0001)

49. Gupta S, Lee RE, Faeder JR. Parallel Tempering with Lasso for model reduction in systems biology. *PLoS computational biology*. 2020; 16(3):e1007669. <https://doi.org/10.1371/journal.pcbi.1007669> PMID: 32150537
50. Earl DJ, Deem MW. Parallel tempering: Theory, applications, and new perspectives. *Physical Chemistry Chemical Physics*. 2005; 7(23):3910–3916. <https://doi.org/10.1039/b509983h> PMID: 19810318
51. Fukunishi H, Watanabe O, Takada S. On the Hamiltonian replica exchange method for efficient sampling of biomolecular systems: Application to protein structure prediction. *The Journal of chemical physics*. 2002; 116(20):9058–9067. <https://doi.org/10.1063/1.1472510>
52. Gupta S, Hainsworth L, Hogg J, Lee R, Faeder J. Evaluation of parallel tempering to accelerate Bayesian parameter estimation in systems biology. In: 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP). IEEE; 2018. p. 690–697.
53. Neal RM. MCMC using Hamiltonian dynamics; 2012.
54. Neal RM, et al. MCMC using Hamiltonian dynamics. *Handbook of markov chain monte carlo*. 2011; 2(11):2.
55. Haario H, Saksman E, Tamminen J, et al. An adaptive Metropolis algorithm. *Bernoulli*. 2001; 7:223–242. <https://doi.org/10.2307/3318737>
56. Andrieu C, Thoms J. A tutorial on adaptive MCMC. *Statistics and computing*. 2008; 18(4):343–373. <https://doi.org/10.1007/s11222-008-9110-y>
57. Betancourt M. A Conceptual Introduction to Hamiltonian Monte Carlo; 2017.
58. Dimova S, Bazlyankov T. Numerical methods for Hamiltonian systems: Implementation and comparison. In: AIP Conference Proceedings. vol. 1684. AIP Publishing LLC; 2015. p. 090002.
59. Verlet L. Computer “experiments” on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Physical review*. 1967; 159(1):98. <https://doi.org/10.1103/PhysRev.159.98>
60. Strang G. On the construction and comparison of difference schemes. *SIAM Journal on Numerical Analysis*. 1968; 5(3):506–517. <https://doi.org/10.1137/0705041>
61. Raj A, Van Den Bogaard P, Rifkin SA, Van Oudenaarden A, Tyagi S. Imaging individual mRNA molecules using multiple singly labeled probes. *Nature methods*. 2008; 5:877–879. <https://doi.org/10.1038/nmeth.1253> PMID: 18806792
62. Wheat JC, Sella Y, Willcockson M, Skoultchi AI, Bergman A, Singer RH, et al. Single-molecule imaging of transcription dynamics in somatic stem cells. *Nature*. 2020; 583(7816):431–436. <https://doi.org/10.1038/s41586-020-2432-4> PMID: 32581360
63. Ballnus B, Schaper S, Theis FJ, Hasenauer J. Bayesian parameter estimation for biochemical reaction networks using region-based adaptive parallel tempering. *Bioinformatics*. 2018; 34:i494–i501. <https://doi.org/10.1093/bioinformatics/bty229> PMID: 29949983
64. Foreman-Mackey D, Hogg DW, Lang D, Goodman J. emcee: the MCMC hammer. *Publications of the Astronomical Society of the Pacific*. 2013; 125(925):306. <https://doi.org/10.1086/670067>
65. Goodman J, Weare J. Ensemble samplers with affine invariance. *Communications in applied mathematics and computational science*. 2010; 5(1):65–80. <https://doi.org/10.2140/camcos.2010.5.65>
66. Karamanis M, Beutler F, Peacock JA, Nabergoj D, Seljak U. Accelerating astronomical and cosmological inference with preconditioned Monte Carlo. *Monthly Notices of the Royal Astronomical Society*. 2022; 516(2):1644–1653. <https://doi.org/10.1093/mnras/stac2272>
67. Huijser D, Goodman J, Brewer BJ. Properties of the affine-invariant ensemble sampler’s ‘stretch move’ in high dimensions. *Australian & New Zealand Journal of Statistics*. 2022; 64(1):1–26. <https://doi.org/10.1111/anzs.12358>
68. Efendiev Y, Hou T, Luo W. Preconditioning Markov chain Monte Carlo simulations using coarse-scale models. *SIAM Journal on Scientific Computing*. 2006; 28(2):776–803. <https://doi.org/10.1137/050628568>
69. Munsky B, Khammash M. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of chemical physics*. 2006; 124(4):044104. <https://doi.org/10.1063/1.2145882> PMID: 16460146
70. Gupta A, Khammash M. Finding the steady-state solution of the chemical master equation. In: 2017 IEEE Conference on Control Technology and Applications (CCTA); 2017. p. 953–954.
71. Weber L, Raymond W, Munsky B. Identification of gene regulation models from single-cell data. *Physical biology*. 2018; 15:055001. <https://doi.org/10.1088/1478-3975/aabc31> PMID: 29624181
72. Chen KH, Boettiger AN, Moffitt JR, Wang S, Zhuang X. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science*. 2015; 348(6233):aaa6090. <https://doi.org/10.1126/science.aaa6090> PMID: 25858977

73. Lubeck E, Coskun AF, Zhiyentayev T, Ahmad M, Cai L. Single-cell in situ RNA profiling by sequential hybridization. *Nature methods*. 2014; 11(4):360. <https://doi.org/10.1038/nmeth.2892> PMID: 24681720
74. Vo H, Sidje RB. Improved Krylov-FSP method for solving the chemical master equation. *Lect Notes Eng Comput Sci*. 2016; 2226.
75. Vo HD, Munsy BE. A parallel implementation of the Finite State Projection algorithm for the solution of the Chemical Master Equation. *bioRxiv*. 2020;.
76. Kazeev V, Khammash M, Nip M, Schwab C. Direct solution of the chemical master equation using quantized tensor trains. *PLoS computational biology*. 2014; 10(3):e1003359. <https://doi.org/10.1371/journal.pcbi.1003359> PMID: 24626049
77. Dufera TT. Deep neural network for system of ordinary differential equations: Vectorized algorithm and simulation. *Machine Learning with Applications*. 2021; p. 100058. <https://doi.org/10.1016/j.mlwa.2021.100058>
78. Defez E, Ibáñez J, Alonso-Jordá P, Alonso JM, Peinado J. On Bernoulli matrix polynomials and matrix exponential approximation. *Journal of Computational and Applied Mathematics*. 2022; 404:113207. <https://doi.org/10.1016/j.cam.2020.113207>
79. Jiang Q, Fu X, Yan S, Li R, Du W, Cao Z, et al. Neural network aided approximation and parameter inference of non-Markovian models of gene expression. *Nature communications*. 2021; 12(1):1–12. <https://doi.org/10.1038/s41467-021-22919-1> PMID: 33976195
80. Dong GQ, McMillen DR. Effects of protein maturation on the noise in gene expression. *Physical Review E*. 2008; 77(2):021908. <https://doi.org/10.1103/PhysRevE.77.021908> PMID: 18352052