Inferring kinetic parameters of oscillatory gene regulation from single cell time series data

Joshua Burton[1], Cerys S. Manning[2], Magnus Rattray[1], Nancy Papalopulu[2,†], and Jochen Kursawe[3,⋆]

[1]Division of Informatics, Imaging and Data Sciences, Faculty of Biology Medicine and Health, The University of Manchester, Oxford Road, Manchester, M13 9PT, UK

[2]School of Medical Sciences, Division of Developmental Biology and Medicine, Faculty of Biology Medicine and Health, The University of Manchester, Oxford Road, Manchester M13 9PT, UK

[3]School of Mathematics and Statistics, University of St Andrews, North Haugh, St Andrews, KY16 9SS, UK

[†]nancy.papalopulu@manchester.ac.uk

[⋆]jochen.kursawe@st-andrews.ac.uk

Abstract

Gene expression dynamics, such as stochastic oscillations and aperiodic fluctuations, have been associated with cell fate changes in multiple contexts, including development and cancer. Single cell live imaging of protein expression with endogenous reporters is widely used to observe such gene expression dynamics. However, the experimental investigation of regulatory mechanisms underlying the observed dynamics is challenging, since these mechanisms include complex interactions of multiple processes, including transcription, translation, and protein degradation. Here, we present a Bayesian method to infer kinetic parameters of oscillatory gene expression regulation using an auto-negative feedback motif with delay. Specifically, we use a delay-adapted nonlinear Kalman filter within a Metropolis-adjusted Langevin algorithm to identify posterior probability distributions. Our method can be applied to time series data on gene expression from single cells and is able to infer multiple parameters simultaneously. We apply it to published

data on murine neural progenitor cells and show that it outperforms alternative methods. We further analyse how parameter uncertainty depends on the duration and time resolution of an imaging experiment, to make experimental design recommendations. This work demonstrates the utility of parameter inference on time course data from single cells and enables new studies on cell fate changes and population heterogeneity.

Keywords: Parameter inference, Bayesian methods, Gene expression oscillations, MCMC, Kalman filters, Stem cell differentiation

Lay summary: The cells in an organism can change their state and behaviour by varying the concentration of key molecules, so called transcription factors. For some transcription factors, extensive amounts of time series data reveal oscillations of their concentrations in time. Here, we introduce a statistical method that can analyse such data to infer the rates at which the transcription factor molecules are produced and degraded, and how strongly oscillating transcription factors self-regulate. Our tool may be used to identify differences between individual cells, and is aimed to facilitate studies on the relationship between transcription factor oscillations and cell behaviour.

[This document includes the Supplementary Information, starting on page 42]

1 Introduction

The identification of regulatory mechanisms that control gene expression may have important implications in biological systems. Cell state transitions are a key contributor to many processes in healthy and diseased tissue, and as such they play a major role in development, regeneration, and cancer. There is an increasing amount of literature uncovering the relationship between gene expression dynamics, i.e. dynamic changes in protein copy numbers from a single gene, and cell state transitions[1–7]. For example, Imayoshi et al. (2013)[1] used optogenetics to show that oscillatory expression of the transcription factor ASCL1 promotes cell proliferation of mouse neural progenitor cells, whereas sustained expression promotes differentiation. Manning et al. (2019)[2] linked aperiodic HES5 protein expression dynamics to murine neural progenitors, and declining oscillatory dynamics to differentiating neurons. Further evidence by Soto et. al. and Phillips et. al.[3;8] demonstrates the contribution of gene expression noise to tuning oscillatory dynamics and

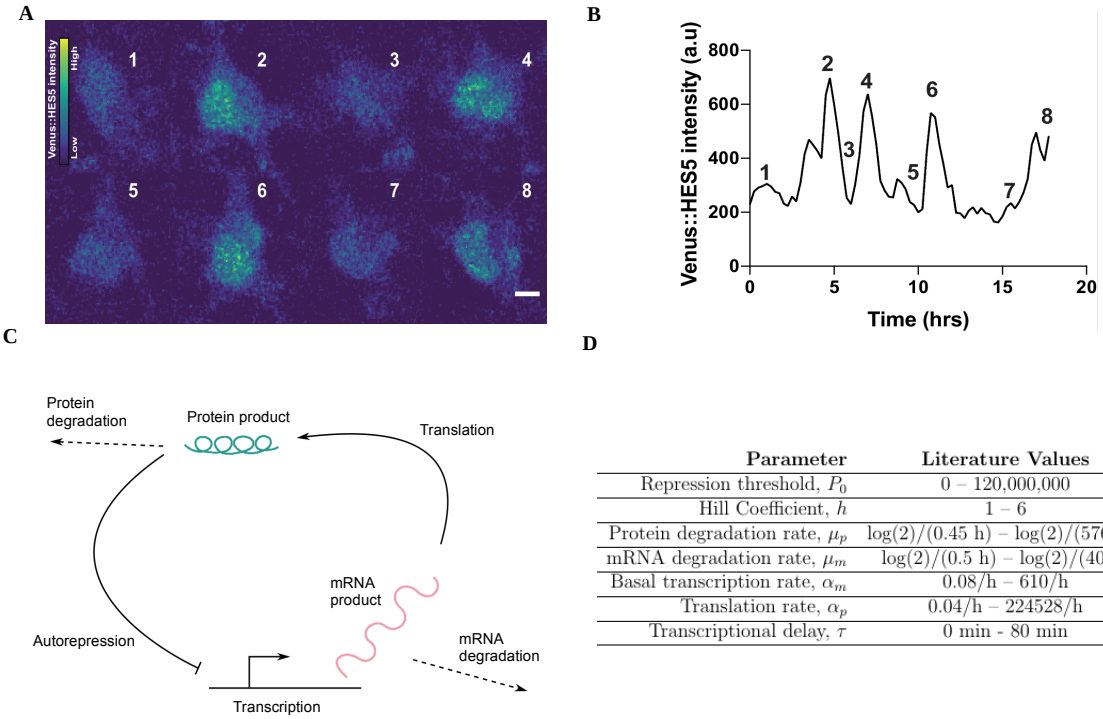
influencing dynamically driven cell state transitions.

Experimentally, the dynamics of gene expression can be studied using a variety of approaches. Accurate measurements of protein dynamics are made through live-imaging of transcription factors in single cells, which provides real-time information on gene regulation and identifies cell-to-cell heterogeneity. This can be achieved through fluorescent fusion reporters[9], where endogenously expressed proteins are attached to fluorescent reporter molecules. Fluorescence microscopy can then be used to obtain time series data that quantify protein expression levels over time (Figure 1A and B, supplementary section S.10). It may further be possible to translate the fluorescence intensity into exact protein copy numbers[2;3]. Fluorescent protein reporters are widely used to research the role of transcription factor dynamics in cell differentiation events, and have provided dynamic data on gene expression in various contexts, such as neural differentiation, circadian regulation, and cell cycle regulation[1;2;10–13].

Mechanistically, dynamic gene expression is controlled by multiple processes, including transcriptional pulsing (transcription occurring in pulses or bursts), stochastic fluctuations (due to a limited number of molecules), gene regulatory interactions and translational control. In order to understand how these processes interact to modulate gene expression dynamics it is necessary to use mathematical models.

Within systems biology, mathematical models are often represented as a collection of gene regulatory motifs[14;15]. One very common motif is the delay-mediated, auto-repressive negative feedback loop (Figure 1C), which gives rise to oscillations and other dynamic patterns of gene expression that have been observed in somitogenesis, neurogenesis, and in cancer cell lines[2;3;7;10;16]. In this motif, a protein represses the transcription of its own gene. In combination with delays that are intrinsic to biological systems, this admits a range of dynamic behaviours, most notably oscillations at the mRNA and protein level. Regulation of gene expression through the auto-negative feedback motif contributes to cell state changes in multiple systems, including neural differentiation[2;17;18].

Despite great advances in the collection of dynamic data on gene expression, and the modelling of these data, challenges remain when calibrating models to data. Even simple mathematical models, such as the auto-negative feedback motif (Figure 1C), employ multiple model parameters that



Parameter	Literature Values
Repression threshold, P_0	0 – 120,000,000
Hill Coefficient, h	1 – 6
Protein degradation rate, μ_p	log(2)/(0.45 h) – log(2)/(5763 h)
mRNA degradation rate, μ_m	log(2)/(0.5 h) – log(2)/(40 h)
Basal transcription rate, α_m	0.08/h – 610/h
Translation rate, α_p	0.04/h – 224528/h
Transcriptional delay, τ	0 min - 80 min

Figure 1: **Time series data of protein expression can be modelled with an auto-negative feedback motif.** **A.** Stills from a movie of a single cortical neural progenitor *in vitro* with Venus::HES5 knock-in reporter. Colour bar shows Venus::HES5 intensity. Stills taken at time points 1.75h (1), 4.5h (2), 6h (3), 7h (4), 9.5h (5), 10.75h, (6), 15.5h (7), 17.25h (8). Scale bar 5μm. For details on data collection see supplementary section S.10.1. **B.** Venus::HES5 intensity time series of cell in A. **C.** Graphical representation of the auto-negative feedback motif. **D.** Model parameter values taken from previously published experiments and theoretical considerations.[2;8;19–23].

correspond to biophysical quantities. For example, the auto-negative feedback motif uses rates of transcription, translation, degradation, and other parameters to predict protein and mRNA expression dynamics. Each of these parameters can take a large range of values (Figure 1D). For many application areas, parameter inference, i.e. identifying which parameters correspond to a given experimentally obtained data set, remains an open problem, since it requires the 'inverse' of the model, which typically cannot be computed directly. However, solving this problem bears great potential for the research of gene expression dynamics and its links to cell fate. Identifying

which parameter changes correspond to observed differences in protein expression dynamics may illuminate the molecular pathways that contribute to cell fate control, and identify new sources of heterogeneity within a cell population.

The need for parameter estimation in biological systems has motivated extensive research in recent years, with a variety of approaches being developed for different types of data[24–27]. Techniques using Bayesian inference have emerged as a preferred approach due to their ability to quantify uncertainty in the face of noisy data, which is a common feature of biological experiments[28], by representing parameters with distributions, rather than point estimates[20;29–35]. Placing probability distributions over our parameters, rather than treating them as point estimates, allows us not only to determine the most likely values for each of the parameters, given some data, but also to quantify our uncertainty in them.

To achieve parameter estimation with uncertainty quantification, Bayesian inference aims to identify the posterior distribution of the model under consideration, denoted $\pi(\boldsymbol{\theta} \mid \mathbf{y})$, where $\boldsymbol{\theta}$ and $\mathbf{y}$ are the model parameters and observed data respectively. The posterior distribution describes the probability of the model parameters given observed data, and can be calculated using Bayes' rule

$$\begin{aligned} \pi(\boldsymbol{\theta} \mid \mathbf{y}) &= \frac{\pi(\mathbf{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\pi(\mathbf{y})} \\ &\propto \pi(\boldsymbol{\theta})\pi(\mathbf{y} \mid \boldsymbol{\theta}). \end{aligned} \tag{1}$$

Here, $\pi(\mathbf{y} \mid \boldsymbol{\theta})$ is referred to as the likelihood, and is a measure of the fit of a statistical model to the observed data, given specific values of the model parameters. The prior probability, $\pi(\boldsymbol{\theta})$, is a distribution which outlines one's beliefs in the parameters $\boldsymbol{\theta}$ before any new data is taken into account. These prior distributions can be informed using published data (Figure 1D), as well as physical constraints (e.g. rate constants must be positive). To visualise the posterior distribution and use it in further analysis it is common to work with computationally generated samples from this distribution. Posterior probabilities may be difficult to compute directly, hindering the efficient generation of these samples[36;37].

Specifically, it may not be possible to calculate posterior probabilities if the likelihood of the model is not available. In these cases, Approximate Bayesian Computation (ABC) can be used. However, ABC reduces the data to a small number of summary statistics, which inevitably decreases the accuracy of inference[38]. If an expression for the likelihood is available and can be calculated at given parameter points, the calculation of the marginal likelihood $\pi(\mathbf{y})$ often poses a further challenge in Bayesian inference, since it may require the numerical integration of the likelihood and prior probability. To overcome this challenge, sampling from the exact posterior distribution can be achieved using Markov chain Monte Carlo (MCMC) techniques, such as the Metropolis-Hastings random walk[39].

MCMC methods can produce samples from the posterior distribution $\pi(\boldsymbol{\theta} \mid \mathbf{y})$ even if the integration factor $\pi(\mathbf{y})$ is unknown. In many scenarios, the reconstruction of a posterior distribution using MCMC sampling can be slow, in particular if the parameter space is high-dimensional, if the calculations of the likelihood are computationally expensive, or if parameters are highly correlated within the posterior distribution[40]. In these scenarios, more efficient Hamiltonian Monte Carlo (HMC) or Metropolis-adjusted Langevin algorithm (MALA) methods are preferable[41–43]. HMC and MALA algorithms additionally require the gradient of the posterior probability with respect to the model parameters and can result in orders-of-magnitude faster convergence of the sampled distribution to the posterior distribution, especially for high-dimensional distributions or when parameter correlations are present[42;44].

For time series data specifically, a common approach to calculating the likelihood is the Kalman filter. The Kalman filter is an algorithm which calculates the likelihood of the data at each time point, given a mathematical model of stochastic dynamics, and an observation noise model. It can generally be applied to Markov processes, where dynamic changes over time only depend on the current state of the system, and not past states. The Kalman filter is a powerful method to calculate posterior probabilities if delays are not present in the model[45], and can be extended to estimate the gradient of the likelihood function, making gradient-based sampling of the posterior distribution possible[46].

A number of recent methods focus specifically on time series of gene expression[2;26;47–51]. For

the study of oscillatory gene expression, a wide array of studies discuss time series data of protein concentrations, such as in Figure 1A,B, as well as the description of these data through the auto-negative feedback motif (Figure 1C). Despite this, a reliable Bayesian inference method for this popular combination of data and model is still missing. Since the model includes delays, the widely-used Kalman filter approaches are not applicable. Recently, Calderazzo et al. (2018) have addressed this problem by identifying a method to introduce delays into the Kalman filter[52], indicating that accurate Bayesian inference for the auto-negative feedback motif on time series data of gene expression may be possible. However, this approach lacks the ability to calculate gradients of the posterior probability distribution, thus preventing the use of efficient gradient-based sampling methods. Furthermore, while Calderazzo et al. (2018) applied their method to a motif containing negative feedback, this method has not yet been applied to the widely used motif in Figure 1C, which includes mRNA in addition to protein.

Here, we present a Bayesian inference pipeline that can be used as a non-invasive method to measure kinetic parameters of gene expression emerging from the auto-negative feedback motif using protein expression time course data. We extend the Kalman filtering method presented by Calderazzo et al. (2018) by introducing a recursive implementation to calculate the gradient of the likelihood. This enables us to embed the non-linear delay-adapted Kalman filter into a state-of-the-art MALA sampling algorithm. This extension enhances the robustness of the inference, making it more suitable for use in typical experimental settings.

Our method is able to capture multiple kinetic parameters of gene expression simultaneously using time course data from single cells, and outperforms previous approaches. We demonstrate the accuracy of our method on *in silico* data, provide an example on how the method can be applied to experimental data, and show how the method can be used to obtain experimental design recommendations. This work is paving the way for the use of Bayesian inference methods for the investigation of gene expression dynamics and their links to cell fate.

2 Methods

In this section we give an overview of the key components of our method. First, we introduce the mathematical model for the auto-negative feedback motif. Then, we discuss how we use a delay-adapted non-linear Kalman filter to approximate the likelihood function. Lastly, we provide details on data processing. Descriptions of our method that require longer derivations, as well as further details on data collection, are provided in the Supplementary Information. This includes our implementation of two MCMC methods, Metropolis-Hastings random walk (MH) and MALA, as well as our proposed algorithm to compute the gradient of the likelihood function, which is a major technical advancement in this paper. The availability of this gradient enables the use of a wider range of MCMC samplers, such as MALA, which we use throughout the paper.

2.1 The negative feedback chemical reaction network

Here we consider a widely used model of gene expression, that incorporates knowledge of the auto-repressive negative feedback loop (Figure 1C). Our model describes both protein and mRNA expression dynamics over time at the level of a single cell, accounting for transcription and translation, as well as degradation. We include a delay in the model, representing the time taken from the initiation of transcription until the production of a transcript and its removal from the nucleus. We further account for the effect of transcriptional auto-repression, where a high abundance of the target protein inhibits transcription of the mRNA [22;23;53].

Let $p(t)$ and $m(t)$ define the number of protein and mRNA molecules, respectively, at time t for a gene of interest. Gene expression is often subject to stochastic effects due to finite molecule numbers. To reflect this we model the system with delayed Chemical Langevin Equations [54–56],

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \alpha_m f(p(t-\tau)) - \mu_m m + \sqrt{\alpha_m f(p(t-\tau)) + \mu_m m}\,\xi_m, \tag{2}$$

$$\frac{\mathrm{d}p}{\mathrm{d}t} = \alpha_p m - \mu_p p + \sqrt{\alpha_p m + \mu_p p}\,\xi_p, \tag{3}$$

where ξ_m, ξ_p denote Gaussian white noise, i.e.

$$\langle \xi_m(t_1)\xi_m(t_2)\rangle = \delta(t_1 - t_2),$$
$$\langle \xi_p(t_1)\xi_p(t_2)\rangle = \delta(t_1 - t_2),$$
$$\langle \xi_m(t_1)\xi_p(t_2)\rangle = 0,$$

where $\delta(\cdot)$ is the Dirac-delta function.

The parameters μ_m, μ_p, α_m, and α_p describe the rate of mRNA degradation, protein degradation, basal transcription rate in the absence of protein, and translation rate respectively. The transcriptional delay is given by τ, and auto-repression is taken into account via the use of a Hill function

$$f(p(t-\tau)) = \frac{1}{1 + [p(t-\tau)/P_0]^h}, \tag{4}$$

reducing the rate of transcription for increasing amounts of protein p at time $t - \tau$[57]. Here, τ, the time delay, is the duration of the transcription process. The Hill function (eq. (4)) is close to one when the protein at time $t - \tau$ is much less than the repression threshold P_0 and close to zero when the the protein at time $t - \tau$ is much more than the repression threshold. The steepness of the transition from one to zero can be regulated by the Hill coefficient h. The Hill coefficient reflects the extent of cooperativity between ligand binding sites for the gene of interest[58].

From equations (2) and (3) we can see that the instantaneous rate of transcription is determined by $\alpha_m f(p(t-\tau))$. This allows us to define an approximation for the average rate of transcription as

$$\alpha_T = \alpha_m f(\hat{p}) \tag{5}$$

where $\hat{p}$ is the average expression of protein. This average expression of protein may be obtained from simulated or experimental data.

We simulate the stochastic differential equations (2) and (3) using the Euler-Maruyama method with a time step $\Delta t = 1$ min, which is chosen sufficiently small to ensure numerical accuracy of the scheme.

A deterministic version of the model in equations (2) and (3) was first developed by Monk (2003) in order to describe gene expression oscillations of Hes1, p53 and NF-κB, and various versions of the model have since been widely studied[2;8;21–23;55;56]. In particular, when molecular copy numbers of mRNA and protein are low, we expect the rate processes of transcription, translation, and degradation to stochastically vary with time. This effect is accounted for by the noise terms in the Chemical Langevin Equations (2) and (3)[54–56].

2.2 The likelihood function can be evaluated through Kalman filtering

The Kalman filter is an algorithm which calculates the likelihood function for linear stochastic differential equations describing time-series data[59]. The Kalman filter evaluates the likelihood of each time-point recording consecutively. The full likelihood is then the product of these individual likelihoods, exploiting the Markov property of the underlying stochastic process. The Kalman filter can be extended to non-linear dynamical systems by using piecewise-linear Gaussian approximations[60].

Here, we implement a Kalman filter, extended to account for non-linearity and delay, in order to evaluate the likelihood that our observed data results from the model in eqs. (2) and (3) at a given parameter combination. This likelihood can then be used to infer model parameters for a given experimentally observed time series recording of gene expression. The resulting posterior distribution may then represent testable predictions on the biophysical characteristics of the gene of interest, such as transcription, translation, and degradation.

Our Kalman filter implementation uses a finer discretisation on the time axis than that given by the observation interval. Specifically, we introduce z hidden states between consecutive observations. Introducing such hidden states is common when applying Kalman filters to non-linear stochastic differential equations. It increases the accuracy of a piece-wise linear Gaussian approximation. In the following, the time variable t will assume integer values numbering all discretisation time points, i.e. $t = 0, 1, ..., nz$, where n is the total number of observations.

It is possible to show that the likelihood of a set of observations given specific model parameters can be expressed as[52]

$$\pi(\mathbf{y} \mid \boldsymbol{\theta}) = \prod_{i=0}^{n-1} \phi(y_{i\cdot z}; F\rho_{i\cdot z}, FP_{i\cdot z}F^T + \Sigma_\epsilon), \tag{6}$$

where the subscript $i \cdot z$ denotes multiplication of i and z and

$$\phi(x; \mu, \Sigma) = \frac{1}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right) \tag{7}$$

is the multivariate Normal distribution. The true, unobserved state of the system at time t is given by $\boldsymbol{X}(t) = x_t = [m(t), p(t)]^{\mathrm{T}}$, and the relationship between x_t and the observed data y_t is given by $y_t = Fx_t + \epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \Sigma_\epsilon)$ and F is a 1×2 matrix. Thus, F and ϵ represent our measurement model. Throughout, we use $F = [0, 1]$, since we aim to apply our method to data on protein expression dynamics, where measurements of mRNA levels are not available. The value Σ_ϵ is called the measurement variance, and describes the observation noise introduced through the experimental measurement process. The variables ρ and P represent the *state space mean* and *state space variance* respectively. We define $y_{0:t} = [y_0, y_z, y_{2z}, \ldots, y_t]^T$, and write $\rho_t = \mathbb{E}[\mathbf{X}(t) \mid y_{0:t-1}]$ and $P_t = \mathrm{Cov}(\mathbf{X}(t), \mathbf{X}(t) \mid y_{0:t-1})$.

The Kalman filter calculates ρ_t, and P_t in eq. (6) using an iterative process with two main steps. At iteration k, the first k observations have been used to infer a probability distribution over the true state of the system $\mathbf{X}(t)$ for all discretisation time points up to $t = kz$. This probability distribution is characterised by it's mean $\rho^*_{kz} = \mathbb{E}[\mathbf{X}(t) \mid y_{0:kz}]$ and covariance $P^*_{kz} = \mathrm{Cov}(\mathbf{X}(t), \mathbf{X}(t) \mid y_{0:kz})$.

In the Kalman filter *prediction step* we then use the model to calculate the predicted probability distribution for protein and mRNA copy numbers at the next observation time point, $\mathbf{X}((k+1)z)$. We use this prediction to evaluate the likelihood of the observed data at the $k+1$ observation time point. Before the prediction for the next observation is made, the Kalman filter *update step* is applied, in which the probability distribution of the state space up to observation $k+1$ is updated to take the measurement at $t = (k+1)z$ into account.

For our update step we derive an expression for the mean and variance of the state space distribution $\pi(x_{t-\tau:t} \mid y_{0:t})$, denoted $\rho^*_{t-\tau:t}$ and $P^*_{t-\tau:t}$ respectively. That is, the likelihood of our state space estimates from the past time $t - \tau$ to the current time, t, given all of our current observations. This is necessary in order to accurately predict the state space distribution at the next observation time point, $\pi(x_{t+\Delta t} \mid y_{0:t})$, as past states can affect future states due to the presence of delays. We provide detailed derivations of our Kalman filter prediction and update steps in the

Supplementary Information Section S.1.

2.3 Implementation of MCMC sampling algorithms

The aim of our inference algorithm is to generate independent samples from the posterior distribution, $\pi(\boldsymbol{\theta} \mid \mathbf{y})$. In this paper, we compare results from two different sampling methods, MH and MALA. The MH algorithm and MALA are two of the most widely used MCMC methods for drawing random samples from a probability distribution. For completeness, we provide their algorithms in the Supplementary Information Sections S.2 and S.3.

Drawing proposals using MALA requires the calculation of the gradient of the log-posterior $U(\boldsymbol{\theta})$, which we outline in Section S.4. This is achieved by iteratively computing the derivatives of state space mean, ρ_t, and state space variance, P_t, with respect to each parameter, as detailed in Section S.5.

2.4 Trends in the data are identified by Gaussian processes

Before applying our inference method we detrend protein expression time series using Gaussian process regression, in order to identify and exclude data that show significant long-term trends [61;62] (see Section 3.3 for further motivation). Specifically, we make use of a *scaled* squared exponential Gaussian process combined with white noise, whose kernel is given by

$$k(t, t') = \gamma \exp\left(\frac{-\|x(t) - x(t')\|^2}{2l^2}\right) + \eta\delta(t - t'), \tag{8}$$

where $\|x(t) - x(t')\|$ is the Euclidean distance between $x(t)$ and $x(t')$, l is the lengthscale, and $\gamma, \eta \in (0, \infty)$. In the Gaussian process regression the hyperparameters γ, l, and η are found using constrained optimisation.

The initial value of the lengthscale is 1000 minutes, and is bounded uniformly in the range (1000 min, 2000 min). The lower bound of this range, 1000 minutes, was chosen to ensure that detrending does not perturb ultradian dynamics in the data. The upper bound, 2000 minutes, was chosen sufficiently large to ensure that detrending is not affected by it. The initial value of the parameter γ is the variance of the data, σ^2_{data}, and is restricted by a uniform prior to $(0.1\sigma^2_{\text{data}}, 2\sigma^2_{\text{data}})$. The

parameter η has initial value 100, and is restricted by a uniform prior to $(10^{-5}, \sigma^2_{\text{data}})$. Here, $x(t)$ and $x(t')$ represent our protein expression time course data at time t and t' respectively. We identified data without a significant long term trend manually by visual inspection (see Section 3.3, Figure 4) and removed any residual trend before applying our inference method.

3 Results

Single cells in a seemingly homogeneous population can change cell fate based on gene expression dynamics. The control of gene expression dynamics can be understood with the help of mathematical models, and by fitting these models to experimentally measured data. Here, we analyse our new method for parameter inference on single-cell time series data of gene expression using the widely used auto-negative feedback motif. We first validate our method by showing the performance of our algorithm on *in silico* data sets. We then demonstrate the utility of our method by applying it to experimentally measured data and, finally, use our method to analyse how parameter uncertainty may depend on properties of the data, as well as the experimental design.

3.1 Sampled posterior distributions agree with analytical derivations for one-dimensional parameter inference.

We first test our inference method on *in silico* data from the forward model of the auto-negative feedback motif (Figure 1C). This is done using Chemical Langevin Equations, as detailed in Section 2.1. Specifically, we emulate an in silico imaging experiment by selecting simulated data in sparse intervals of Δt_{obs} mins and mimic measurement noise by adding random perturbations to each observation time point (Figure 2A). These perturbations are drawn from a Gaussian distribution with variance Σ_ϵ. Testing the method on *in silico* data first is beneficial, since ground truth parameter values are known *a priori* for the generated *in silico* data sets, and can be compared to the obtained posterior distributions.

We start by applying our inference method to simple test cases, where the true values of all but one parameter are known, and only the remaining, unknown, parameter value is inferred (Figure 2). This allows us to compare our sampled posterior distributions to the exact likelihood, which can

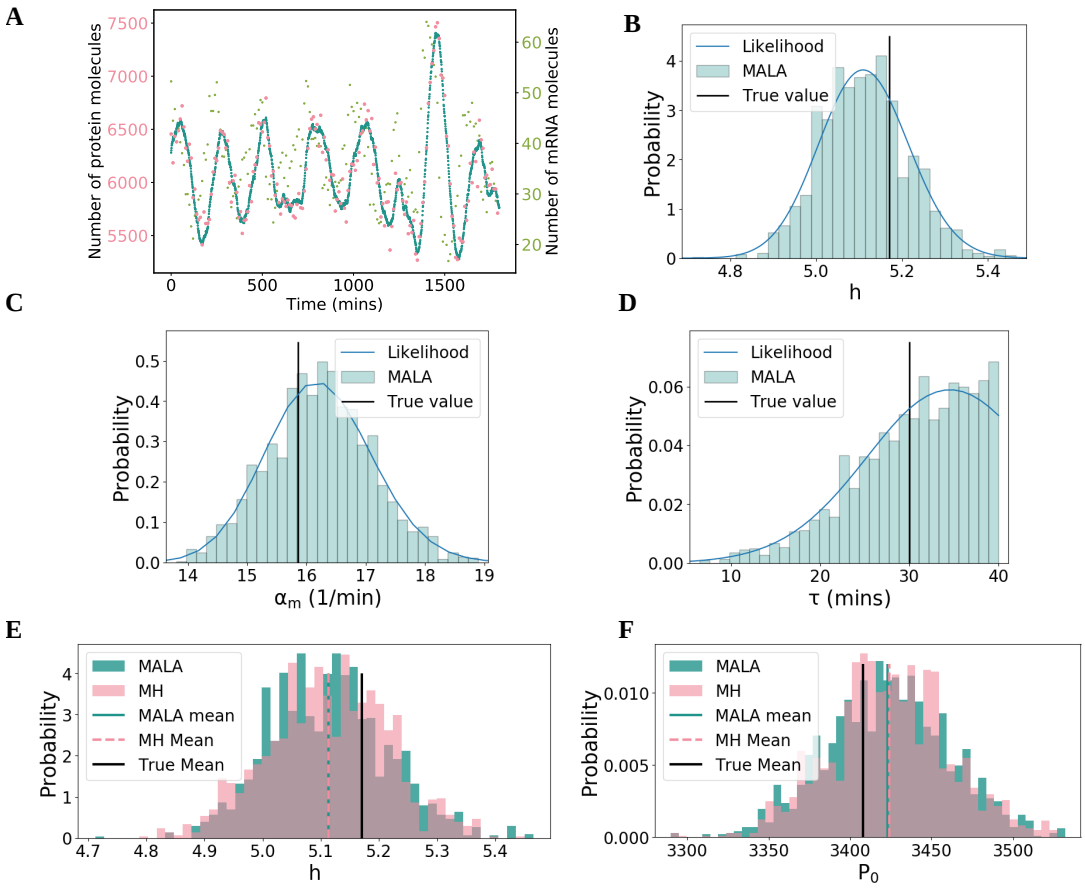


Figure 2: **Our algorithm accurately samples posterior distributions. A.** Simulated experimental data. Protein copy numbers are simulated using the Chemical Langevin Equation (see Section 2, blue dots). Experimental observations are emulated every five minutes by adding Gaussian noise to the protein copy number (pink). The parameter values used were $P_0 = 3407.99, h = 5.17, \mu_m = \log(2)/30, \mu_p = \log(2)/90, \alpha_m = 15.86, \alpha_p = 1.27, \tau = 30, \Sigma_\epsilon = 10000$, and the simulated mRNA copy numbers are also included (green dots). **B, C, D.** Posterior distributions for one-dimensional inference. For individual model parameters, posterior distributions were inferred while keeping all other parameters fixed, respectively. Shown above are the inferred marginal posteriors for the Hill coefficient (B), transcription rate (C), and transcriptional delay (D) respectively as histograms, using MALA as the underlying sampling algorithm (see Section S.2) for 2500 iterations. The blue lines are the exact likelihood calculations. The sampled and exact distributions coincide. **E, F.** Histograms for both MALA and MH on the 1-dimensional problem for the Hill coefficient (E) and repression threshold (F).

be calculated in these one-dimensional examples using equation eq. (6). If our inference method is accurate, the sampled posterior distribution should closely match the exact likelihood if the Markov chain has converged (see Section S.7). We find that this is indeed the case for example *in silico* data sets (Hill coefficient, transcription rate and transcriptional delay in Figure 2B–D, repression threshold and translation rate in Figure S1). Additionally, ground truth parameter values lie well within the support of the posterior distribution (Figure 2B–D, supplementary Figure S1, vertical black lines).

Our proposed inference method uses the MALA sampler, which relies on calculating likelihood gradients (see Section S.4). The comparison with exact calculations in Figure 2B–D and supplementary Figure S1 validates our implementation of MALA, and the associated computations of the likelihood gradient. In order to further test our implementation of MALA, and the associated computations of the likelihood gradients, we compare our results to posterior distributions sampled using the MH algorithm, which does not require gradient calculations. Despite an expected slower convergence of the MH algorithm, this comparison is feasible for one-dimensional posterior distributions, which typically can be well approximated with a few thousand samples. The sampled means have a relative difference below 0.03%, and the standard deviations fall within 4% of each other (Table 1, supplementary Table S3). This comparison reveals that posterior distributions from both samplers agree well with each other (Figure 2E–F, Figure S2), and further validates the implementation of the individual likelihood gradients.

Parameter	**True Value**	$\boldsymbol{\mu}$ **(MALA)**	$\boldsymbol{\mu}$ **(MH)**	$\boldsymbol{\sigma}$ **(MALA)**	$\boldsymbol{\sigma}$ **(MH)**
Repression threshold, P_0	3408	3422	3423	37.51	36.65
Hill Coefficient, h	5.17	5.113	5.112	0.100	0.104

Table 1: The true values for the parameters which were used to generate the data in Figure 3A, alongside the means, $\boldsymbol{\mu}$, and standard deviations, $\boldsymbol{\sigma}$ of the corresponding one-dimensional posterior distributions, from both the MALA and MH algorithms (Figure 2E,F).

3.2 Our method allows for simultaneous inference of multiple model parameters

Having validated the method on one-dimensional posterior distributions, we further test the performance of the method by simultaneously inferring multiple model parameters from a single *in*

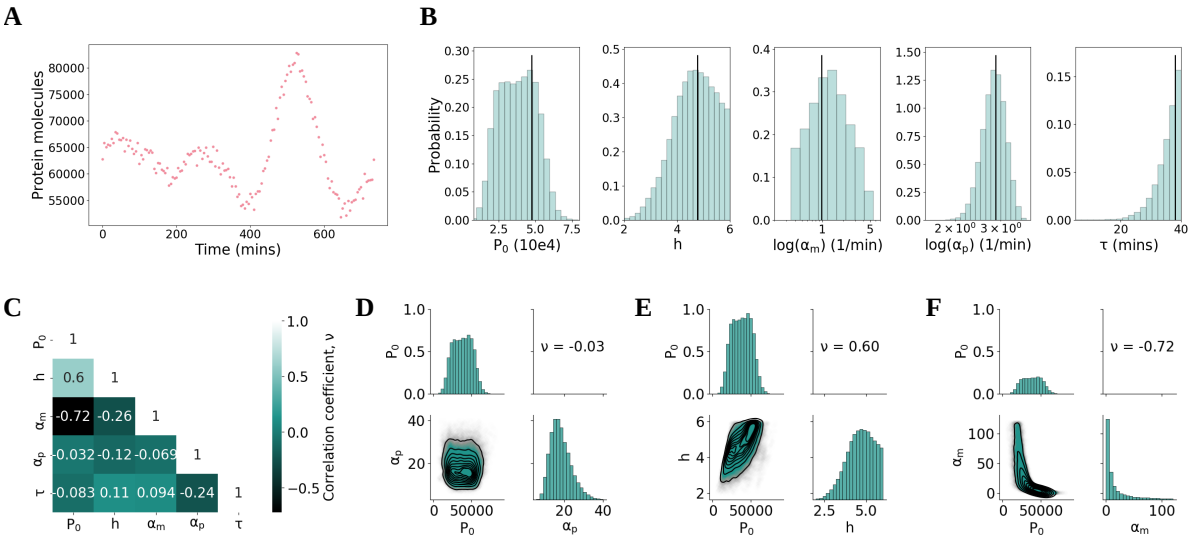


Figure 3: **MCMC sampling enables simultaneous inference of multiple parameters A.** An *in silico* data set was generated using parameter values $P_0 = 47515, h = 4.77, \mu_m = \log(2)/30, \mu_p = \log(2)/90, \alpha_m = 2.65, \alpha_p = 17.61, \tau = 38.0, \Sigma_\epsilon = 10^6$. **B.** Our method is applied to the data set in (A) to sample the joint posterior distribution over five parameters. The marginal posteriors for each parameter are shown. All marginal posterior means are within half a standard deviation of the true value. **C.** A (symmetric) correlation matrix which shows the correlation coefficient, ν, between samples for each pair of parameters. Diagonal entries are, by definition, perfectly correlated ($\nu = 1$), and off diagonal entries take values in the range $[-1, 1]$. **D,E,F.** Joint posterior distributions showing the relationship between different pairs of parameters. D shows the repression threshold, P_0 is not correlated with the protein translation rate, α_p. E shows a correlation of $\nu = 0.6$ between the repression threshold and the Hill coefficient, h. In other words, for this data set, a higher Hill coefficient indicates a higher repression threshold (and vice-versa). Finally, F shows the strong negative correlation between P_0 and α_m.

silico data set and comparing the resulting posterior distribution to the ground truth parameter combination (Figure 3A,B). Since we cannot measure convergence of the sampled posterior through comparison to the true posterior distribution in the multi-dimensional case, we rely on typical MCMC convergence diagnostics (Section S.7).

We choose a data set that shares characteristics with typically collected time course data from single cells. Specifically, our *in silico* data set is of similar length and observation intervals as previously analysed by Manning et al. (2019). In this paper, the degradation rates of protein and mRNA have been measured, so we assume these measurements as known values, leaving five

unknown parameter values to infer. The prior distributions were uniform, defined by the range of values given in Table S1, and log-uniform for α_m and α_p (see Section S.6 for details).

We find that the marginal posterior means, i.e. values of largest probability, all lie within maximally half a standard deviation of the ground truth values (Table 2). This indicates that a high degree of accuracy in the inference can be achieved with the amount of data typically gathered from a single cell.

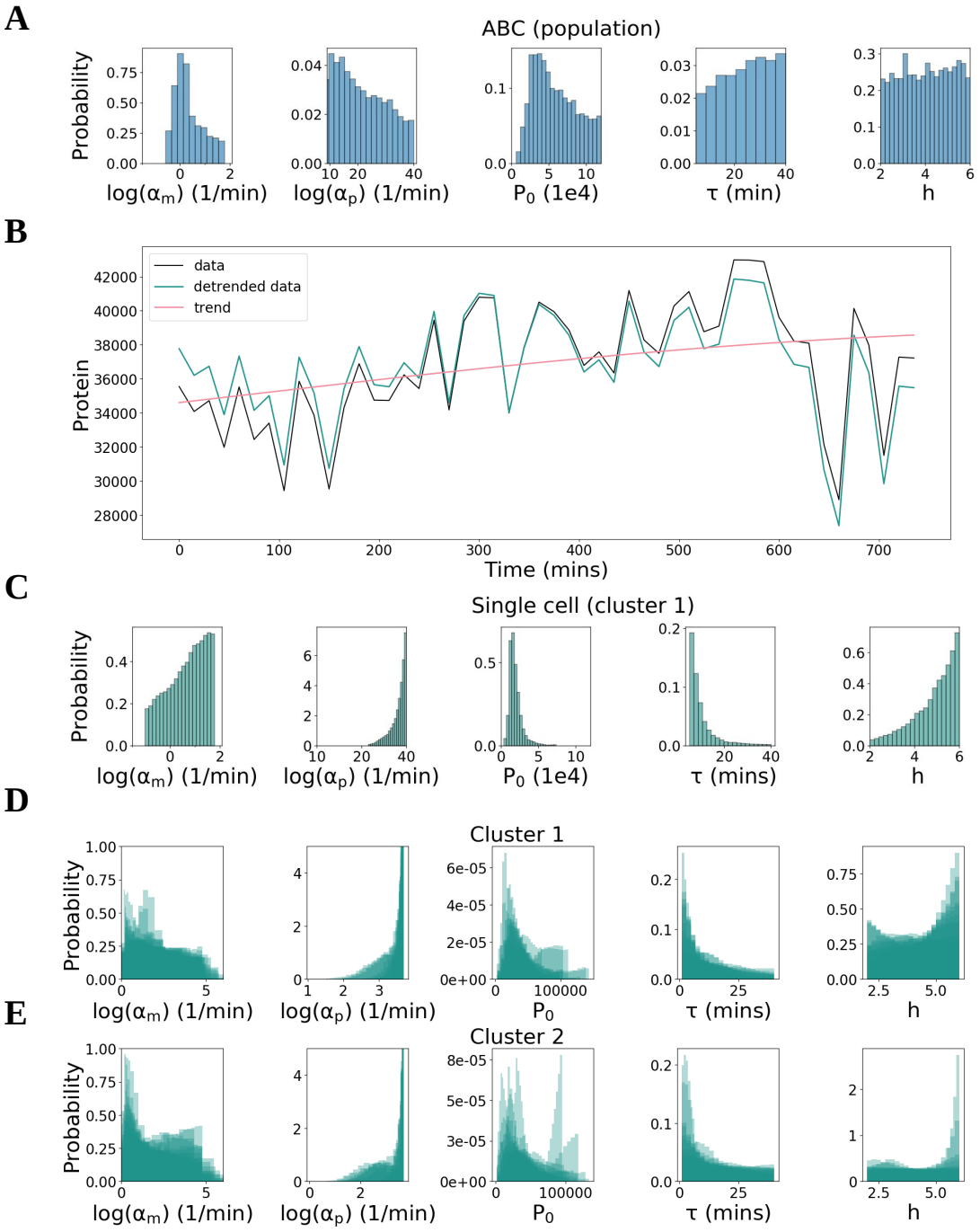
Parameter	**True Value**	**Mode**	**SD**
Repression threshold, P_0	47515	44915	12434
Hill Coefficient, h	4.77	4.41	0.80
Log basal transcription rate, $\log(\alpha_m)$	0.975	1.45	1.24
Log translation rate, $\log(\alpha_p)$	2.869	2.808	0.288
Transcriptional delay, τ	38.0	39.87	4.39

Table 2: The true values for the parameters which were used to generate the data in Figure 2A, alongside the means, $\boldsymbol{\mu}$, and standard deviations, $\boldsymbol{\sigma}$, using MALA.

Simultaneous inference of multiple parameters further allows for the investigation of pairwise parameter correlations, using correlation coefficient ν (Figure 3C). Pairwise correlations provide crucial information on how posterior distributions can be constrained further. Specifically, the strong correlation between the repression threshold, P_0, and the logarithm of the basal transcription rate, $\log(\alpha_m)$ (Figure 3E), highlights that the data in Figure 3A is consistent with either high repression thresholds and low transcription, or vice versa. Such strong pairwise correlations (Figure 3E,F) imply that gaining new information on one of the two parameters would constrain the other. This is not the case when parameters are uncorrelated, such as the transcriptional delay and the translation rate (Figure 3D), and experimentally measured values on either of these parameters would not inform the other.

3.3 Parameter inference on single cell data outperforms previous approaches and may reveal underlying mechanisms for population heterogeneity

Next, we seek to evaluate the performance and utility of our method by applying it to experimentally measured data. Specifically, we investigate data on gene expression oscillations in mouse spinal cord neural progenitor cells[2] (see supplementary section S.10.2), and compare our method to results on



parameter inference from ABC (Figure 4A). In this previous approach, inference was performed using population-level summary statistics of the collected time-course data. This resulted in posterior

Figure 4: **Parameter inference on single cell data outperforms previous approaches. A.** Marginal posterior distributions for each parameter are obtained from the ABC algorithm using summary statistics in place of likelihood evaluations. **B.** Detrending of *in vivo* single cell protein expression data (black line). The sampling interval is 15 minutes, and the length of the time course is 12 hours (720 mins). The mean is subtracted from the data and a squared exponential Gaussian process determines the long term trend (pink line). This trend is removed from the data and the mean is added back in (green line). **C.** Marginal posteriors from MALA for the detrended single cell data shown in panel B. **D,E.** Marginal posterior distributions for every cell in cluster 1 and 2 respectively.

distributions with high parameter uncertainty. Specifically, the marginal posterior distributions for the Hill coefficient and the transcriptional delay were close to uniform, illustrating that the provided summary statistics did not contain sufficient information to constrain the uniform prior distribution. The remaining parameters had distinct modes. Nonetheless, parameter uncertainty was high since the spread of the posterior distribution was comparable to that of the prior[2]. Importantly, this previous approach did not allow for comparison of posterior distributions between single cells.

When applying our method to time series data from fluorescence microscopy experiments, it is necessary to address that our model of the auto-negative feedback motif cannot describe long-term trends in data. Specifically, the model of the auto-negative feedback loop considered here is designed to describe ultradian oscillations that typically have periods shorter than 10 hours[21;22;55], and cannot describe variations in protein numbers on longer timescales, such as one would expect from a slow up- or down-regulation of the gene in the tissue. Hence, we only apply our algorithm to protein expression time series that we expect to be accurately modelled by eqs. (2) and (3) by excluding data that show significant long-term trends. In order to identify such time series, we first remove trends from the time series that vary on lengthscales longer than 10 hours by using Gaussian process regression (see Section 2.4). Then, we manually identify all time series for which the detrended and raw time series visually agree (Figure 4B) and select these for inference.

In order to identify a suitable value for the measurement variance Σ_ϵ we rely on previous estimates[2]. Manning et al. (2019) decomposed the measured time series into two contributions, one from a time-varying signal with finite auto-correlation time, and one from a time-varying signal for which consecutive observations are uncorrelated[2]. This second contribution follows an identical distribution as the measurement error in our model, and was estimated to contribute 10% of the

total variance across all detrended time series. Hence, we set

$$\Sigma_\epsilon = \frac{0.1}{N_c} \sum_{i=1}^{N_c} \sigma_i^2, \tag{9}$$

where N_c is the number of considered traces, and σ_i^2 is the variance for the ith detrended data set.

We find that our method can identify more accurate posterior distributions than the previous ABC-based approach by using single cell time series of gene expression only (Figure 4C vs. Figure 4A.). For the single-cell gene expression time course in Figure 4B, we find that there is still comparatively high uncertainty on the basal transcription rate (α_m in Figure 4C), as the support of the marginal posterior distribution reflects that of the uniform prior distribution . However, for all other model parameters that are inferred from this time course, the marginal posterior distributions are narrower than the prior, and than previously identified marginal posterior distributions from ABC (Figure 4C).

Having investigated marginal posterior distributions from a single cell we proceed to analyse to what extent these posterior distributions can vary across multiple cells in the population. Among the experimental data considered here, hierarchical clustering has previously identified two subpopulations (denoted as clusters 1 and 2) which have distinct gene expression dynamics and which also do not have strong long-term trends[2], such as down-regulation of gene expression. For clusters 1 and 2 there are 19 and 22 cells respectively which we have selected for negligible trends (see Section 2.4).

We find that the posterior distributions inferred from multiple cells share features that are conserved across all cells and both populations (Figure 4D,E). Specifically, the marginal posterior distributions of the translation rate α_p are all larger than exp(2)/min, and biased to larger values. Similarly, the marginal posterior distributions for the delay τ cover the entire range of considered values, and are biased towards smaller values, with most likely values below 10 minutes. These observations appear to hold true for both clusters considered here, and they highlight that parameter inferences obtained from our method are biologically reproducible, which is a necessary feature to enable the use of the method in practical applications.

In contrast, for the basal transcription rates α_m and the Hill coefficient h, marginal posterior distributions vary between individual cells, suggesting that there is an underlying heterogeneity of these parameters across the cell population. However, the remaining parameter uncertainty is too high to reliably identify differences between cells and clusters of cells, raising the question of how imaging protocols may need to be changed in order to achieve lower uncertainty on typical parameter estimates.

3.4 Longer time course data improves accuracy of inference more effectively than more frequent sampling

Typically, longer imaging time series can only be collected at the cost of a lower imaging frequency. When designing experiments, it may be desirable to choose an imaging protocol that optimises the parameter inference towards high accuracy and low uncertainty. However, parameter uncertainty may not only be influenced by the imaging protocol, but also by the bifurcation structure of the underlying dynamical system[63]. Hence, we next analyse how posterior distributions depend on the frequency of sampling, on the length of the imaging interval, and on the position in parameter space. To evaluate the performance of our inference, we investigate the uncertainty using *relative uncertainty*, $\mathrm{RU}_{\boldsymbol{\theta}}$ (Supplementary Section S.7, equation (35)), which quantifies the spread of the posterior distribution. We use this metric to quantify the performance of our inference method on a number of synthetic data sets with different lengths and sampling frequencies, and for different locations in parameter space.

We choose two locations in parameter space that correspond to two different values of oscillation coherence, thus producing qualitatively different expression dynamics (Figure 5A, Table S2). The oscillation coherence is a measure for the quality of observed oscillations (supplementary Section S.8). Choosing parameter combinations with different coherence thus ensures that these correspond to different positions within the bifurcation structure of the auto-negative feedback loop[55;64;65].

We first analyse to what extent collecting data for a longer sampling duration may reduce parameter uncertainty (Figure 5B,D). We find that a longer sampling duration can strongly decrease parameter uncertainty. Doubling the length of the time series reduces the uncertainty by 19% on

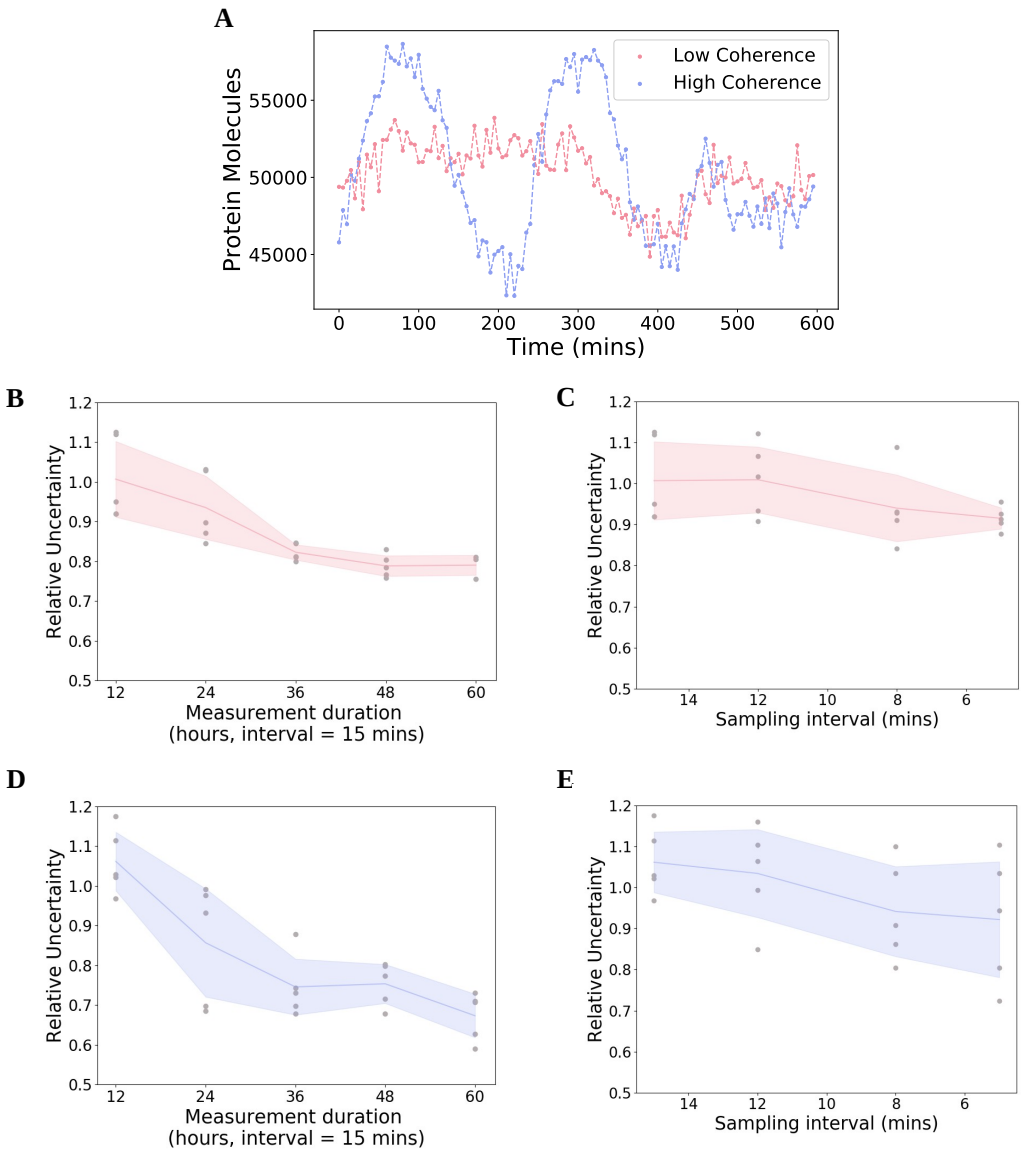


Figure 5: **Increasing the length of time course data improves inference more than increased sampling frequency. A.** Two examples of *in silico* protein observations, one which has low coherence (pink) and another with high coherence (purple). Exact parameter combinations can be found in Table S2. **B.** $\mathrm{RU}_{\boldsymbol{\theta}}$ for low coherence data sets sampled with different lengths, from 12 hours to 60 hours. As length increases from 12 hours to 36 hours, both the mean and standard deviation decrease. **C.** $\mathrm{RU}_{\boldsymbol{\theta}}$ for low coherence data sets sampled with different frequencies. As frequency increases from 15 minutes to 5 minutes, the mean decreases. **D.** Same as B with the high coherence data sets. **E.** Same as C with the high coherence data sets.

average for the high coherence parameter combination, and 7.1% on average for the low coherence parameter combination. A tripling of the available data leads to reductions in uncertainty by 29.8% and 18.3% and for high and low coherence respectively.

In contrast, an increase in sampling frequency leads to a smaller decrease in parameter uncertainty on average (Figure 5C,E). Specifically, doubling the amount of data only leads to a decrease by 11.3% in the case of the high coherence parameter combination, and 6.7% in the case of low coherence. A tripling of the available data lead to reductions in uncertainty of 13.2% and 9.1% for low and high coherence respectively.

We find that analogous conclusions hold true if inference accuracy is analysed ($\mathrm{ME}_{\boldsymbol{\theta}}$, supplementary Section S.7, equation (36)), instead of uncertainty (Figure S3). Accuracy increases with longer sampling durations and shorter imaging intervals, and longer sampling durations have a stronger effect than shorter imaging intervals.

3.5 Additional measurements of mRNA copy numbers improve estimates of the average transcription rate

In the previous section, we have analysed the impact of changes in the imaging protocol on parameter uncertainty overall. Alternatively, it may be desirable to identify interventions that reduce uncertainty for particular parameters of interest. For example, an important quantity of interest may be the average rate of transcription of the investigated gene, introduced as α_T in equation (5). In many *in silico* examples of our parameter inference, this average rate of transcription α_T is poorly inferred, with the mean of the posterior distribution being up to 5 times larger than the ground truth value. This is for example the case in Figure 6A. In this and other examples, the ground truth value lies outside the 85% highest density interval (HDI) of the posterior distribution (Figure 6A–C). Intuitively, one may assume that estimates for the rate of transcription are improved if measurements of mRNA copy numbers, in addition to protein expression dynamics, are considered in the inference.

Hence we next assume that, in addition to data on the dynamics of protein expression, measurements of mRNA copy numbers have been conducted on the observed cells. Specifically, we generate

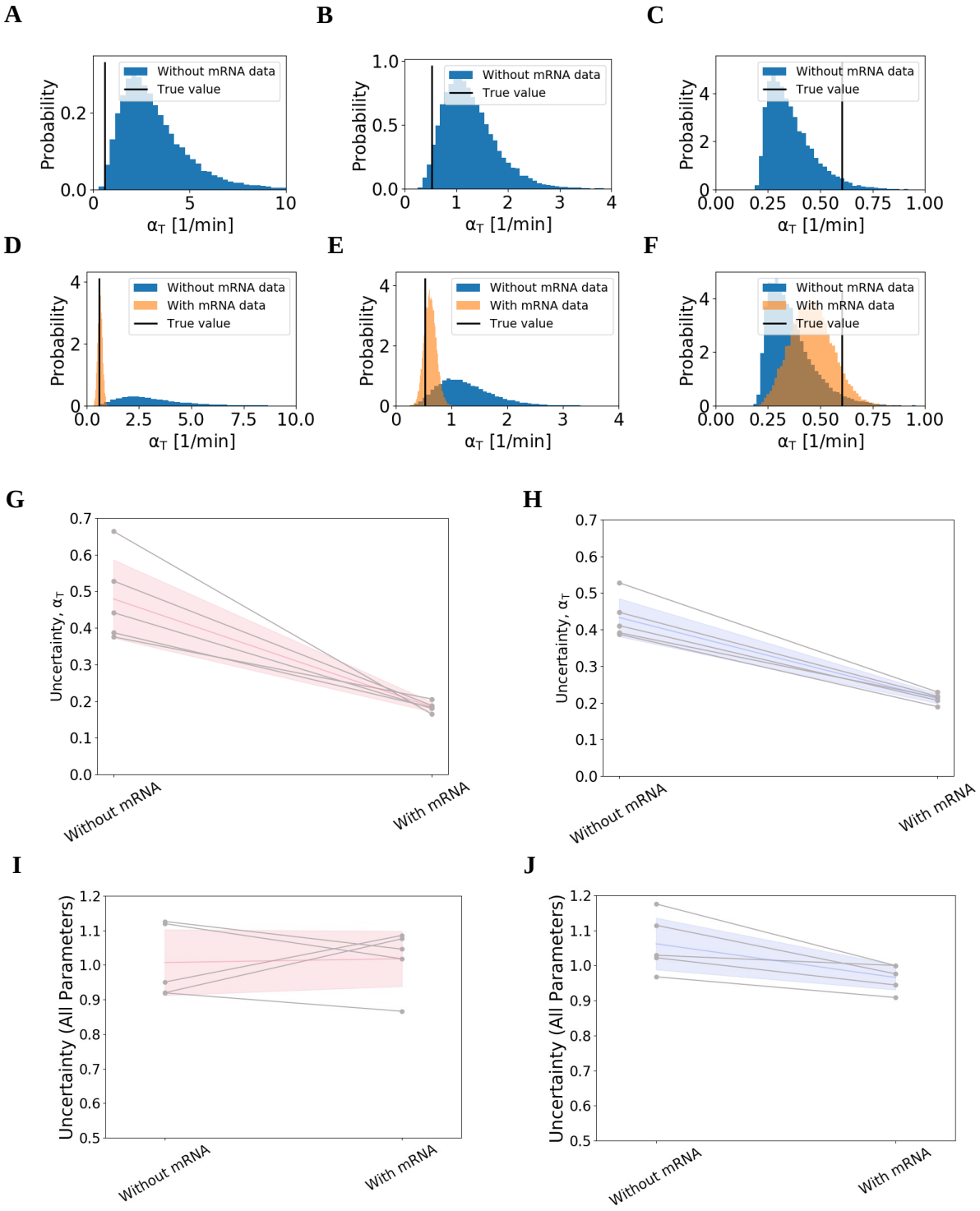

A
B
C
D
E
F
G
H
I
J
Probability
Without mRNA data
With mRNA data
True value
αT [1/min]
Uncertainty, αT
Uncertainty (All Parameters)
Without mRNA
With mRNA

Figure 6: **Additional measurements of mRNA copy numbers improve estimates of the average transcription rate. A,B,C.** Posterior distributions of the average rate of transcription, α_T, calculated using the posterior samples of three example data sets from Figure 5. The ground truth value (vertical black line) is poorly estimated by these posteriors. **D,E,F.** The same three posterior distributions for α_T as in A–C, this time comparing posterior samples drawn without mRNA information (blue) and with mRNA information (orange). Here the ground truth value (vertical black line) is much more closely inferred when mRNA information is included. **G.** Uncertainty of α_T for the low coherence data sets from Figure 5. Uncertainty in α_T is reduced by more than 60% when mRNA information is included. Uncertainty on α_T is calculated using the coefficient of variation, defined by the posterior standard deviation over the posterior mean, $\hat{\sigma}_{\alpha_T}/\hat{\mu}_{\alpha_T}$. **H.** Same as in G but for high coherence data sets. Here mRNA information reduces uncertainty in α_T by more than 50%. **I.** Values of relative uncertainty $\mathrm{RU}_{\boldsymbol{\theta}}$ (equation (35) in supplementary Section S.7) for low coherence data sets (cf. Figure 5) with and without additional data on mRNA copy numbers. **J.** $\mathrm{RU}_{\boldsymbol{\theta}}$ for high coherence data sets from Figure 5 with and without additional data on mRNA copy numbers. In G–J the coloured lines and shaded areas represent the mean and standard deviation across observed values, respectively. Data sets in A–C are all sampled for a duration of 12 hours, and with sampling intervals of 8 minutes, 8 minutes, and 5 minutes respectively. All data sets considered in G–J are sampled every 15 minutes for a duration of 12 hours and the parameters correspond to the low and high coherence parameter sets respectively (see Table S2).

in silico data mimicking a single-molecule *in situ* hybridisation (smFISH) experiment. Such smFISH experiments generate distributions of mRNA copy numbers, thus providing a snapshot of mRNA levels across a population at a fixed time point[66;67]. To account for this additional data, we incorporate the observed distribution of mRNA copy numbers into our likelihood function, such that it effectively penalises parameters for which inferred copy numbers of mRNA are outside the experimentally observed range (see supplementary Section S.9).

We find that this inclusion of mRNA information collected from a cell population leads to more accurate inference of the average transcription rate for single cells, using our algorithm. Observing example data sets from Figure 5, the posterior distributions cover multiple orders of magnitude if only protein expression data is considered in Figure 6D,E, with the mean of the distribution being 5.4 times larger than the true value in Figure 6D, and 2.4 times larger in Figure 6E, respectively. Upon inclusion of mRNA information, these posterior distributions are instead concentrated around the true value, with a relative error below 15.3%. In both examples the ground truth is contained within the 65% HDI. In Figure 6F a posterior distribution that is already close to the true value gets further constrained by the additional mRNA data. In these examples, the observed reduction of uncertainty on the inferred transcription rate is accompanied by a reduction of uncertainty on

estimated mRNA copy numbers for individual cells, as inferred by the Kalman filter (see e.g. supplementary Figure S5A vs. S5B). Investigating the uncertainty on the average inferred transcription rate across data sets introduced in Figure 5, we observe a reduction in uncertainty of 61.8% for low coherence parameter combinations (Figure 6G), and a reduction in uncertainty of 51.2% for high coherence parameter combinations (Figure 6H).

How does this improved estimate of transcription rate affect overall uncertainty across parameter space, as analysed in Figure 5? Counter-intuitively, we find that this inclusion of mRNA data into our parameter inference does not reduce overall parameter uncertainty (Figure 6I,J). For data sets from the low coherence parameter combination, the relative uncertainty increases by 1.1% on average when mRNA information is included (Figure 6I). For data sets from the high coherence parameter combination, uncertainty decreases slightly (9.1% on average (Figure 6J)). Importantly, this reduction of uncertainty is considerably smaller than the reduction of uncertainty observed when longer measurement durations are considered (cf. Figure 5D). We make analogous observations as inference accuracy is analysed (Figure S4), instead of uncertainty. Inference accuracy is not reduced for high coherence data sets when data on mRNA copy numbers is included, and it is only slightly reduced for some of the low coherence data sets, with the effect being much smaller than the effect of considering longer time course data (cf. Figure S3).

The effect that overall uncertainty is not decreased as new data on mRNA copy numbers is included contradicts the intuition that more accurate inference of the average rate of transcription α_T will also reduce uncertainty on model parameters regulating α_T, such as the basal transcription rate, α_m and the repression threshold, P_0. This effect may be attributed to correlations between these parameters, which we typically observe in our posterior distributions (see Figure 3F). For the data set in Figure 6A, inference of α_T is improved upon inclusion of the mRNA. This leads to a tighter coupling between the parameters α_m and P_0 (Figure S6). However, this constraining of the posterior distribution is not reflected in either of the marginal posterior distributions. Thus, although the inclusion of *in silico* smFISH data reduces the spread of the posterior distribution overall, uncertainty within the marginal posterior distributions is not reduced, and individual parameter estimates are not improved. An additional factor is that data from smFISH experiments may be considered to

reflect the time-averaged mRNA copy number distribution of single cells. Hence, these data might not reduce uncertainty on parameters that are expected to predominantly alter the dynamics rather than the level of expression, such as the transcriptional delay τ and Hill coefficient h. Hence, to better infer these parameters, other strategies, e.g. those discussed in Figure 5, may be required.

We conclude that distributions of mRNA copy numbers from population-level measurements can be used to infer average transcription rates for individual cells, using our inference method, which may facilitate the study of transcriptional dynamics in the context of gene expression oscillations. Together with results from Figure 5, this illustrates how our method may be used to evaluate the benefit of different experiments *in silico*, and highlights that our method can be naturally extended to utilise additional data of different types.

4 Discussion

The aim of this work was to develop a statistical tool that can be used to infer kinetic parameters of the auto-negative feedback motif, based on typically collected protein expression time series data from single cells. Importantly, the stochastic nature of the involved processes demanded a method that enables uncertainty quantification. We have achieved our aim by embedding a non-linear delay-adapted Kalman filter into the MALA sampling algorithm. Our method can generate accurate posterior distributions for the simultaneous inference of multiple parameters of the auto-negative feedback motif. The produced posterior distributions are more informative than those from previous approaches. Since our method can be applied to data from single cells, it enables the investigation of cell-to-cell heterogeneity within cell populations. It can further be used to make experimental design recommendations, which we demonstrated by investigating how parameter uncertainty may depend on the position in parameter space, the sampling frequency, and the length of the collected time series data. Additionally, our method may be extended to account for the presence of different types of data, for example to improve estimates of the transcription rate for individual cells.

Often, new inference algorithms are presented on a single data set, and due to necessary tuning requirements of the involved sampling methods, further data sets are not considered. However, it is important to understand the behaviour of a method for a range of data sets if we wish to make

experimental design recommendations. It is an achievement of this paper that we provide a method that demonstratively can reliably infer parameters, even when the size and structure of the data are changed significantly.

The mathematical model underlying our method aims to describe the dynamic expression of a protein which is controlled by auto-negative feedback. The success of our inference relies upon how well this model approximates reality. Mathematical models for the oscillatory expression of transcription factors are informed by experimental research[57;68] and have been developed over time[3;8;22;23;64]. Existing model extensions include interactions with other genes or microRNAs[21] and future models could include effects of transcriptional bursting[69]. The simple model used here provides a starting point for developing inference algorithms for further models including non-linear, stochastic interactions as well as delays, and future validation of experimental predictions can be used to guide data-driven model improvements. To this end, our algorithm may enable model selection to identify gene regulatory network properties, such as interactions between multiple transcription factors.

Chemical Langevin Equations such as eqs. (2) and (3) approximate the full stochastic dynamics of the system by assuming Gaussian increments. Further, our Kalman filter assumes that measurement errors follow a Gaussian distribution, and are not correlated between consecutive time points. The likelihood calculations within the Kalman filter assume that distributions of protein copy numbers, which are predicted by eqs. (2) and (3), can be approximated by Gaussian distributions.

The Gaussian approximation within the Chemical Langevin Equation can break down when molecule concentrations are very low, resulting in an inaccurate simulation of the dynamics. We do not expect this to be a problem for data analysed in this paper, since protein copy numbers throughout our analysis are around 50,000 protein molecules per nucleus. In other applications, the validity of the Chemical Langevin Equation may be explicitly tested on samples from the posterior distribution by directly comparing simulated expression time series with those obtained from an exact sampling algorithm, such as the Gillespie algorithm[70]. Similarly, simulations of the Chemical Langevin Equation can be used to test assumptions on the Gaussianity of the state space made within the Kalman filter. In cases where these assumptions do not hold, alternative inference

algorithms, such as particle filter methods, may need to be developed.

For Bayesian inference problems it is common to use MCMC samplers, such as MH or MALA. We have found that combining a delay-adapted non-linear Kalman filter and MALA can allow us to infer parameters of the auto-negative feedback motif. This builds on previous approaches which applied a Kalman filter in the context of a different transcriptional feedback motif with delay[52]. MCMC algorithms typically require tuning which may be data specific. We have taken steps to reduce additional input from the user by using MALA, which proposes new samples based on the gradient of the target posterior, hence accounting for geometric properties of parameter space, which can result in faster, more robust performance on some distributions[44]. MALA also has fewer tuning parameters than other algorithms, such as HMC. This allows us to more easily incorporate adaptation into our algorithm[71]. Surprisingly, the MALA sampler did not result in faster convergence than MH on example posteriors from our model (see Supplementary Figure S7). Hence, the added computational cost of calculating likelihood gradients will not be beneficial in all applications, especially since, in our model, gradient calculations increase the computational cost of individual parameter samples by a factor 12. We expect the availability of likelihood gradients to achieve a speed-up in high-dimensional problems, where convergence speeds of MALA scale with $d^{1/3}$, rather than d^1 for MH[72], for model dimension d. Note, that more efficient MCMC algorithms can eliminate the problem of tuning entirely[44]. These methods rely on the computation of the Hessian, i.e. the second derivative of the likelihood function. Deriving expressions for the Hessian and investigating the efficiency of the resulting algorithm is thus a potential avenue for future work.

In our applications of the algorithm to experimentally measured data, we detrended the data before applying our inference (Figure 4B). Such detrending is commonly used when analysing time series of oscillatory signals[2;73;74]. The detrending removes signal fluctuations from the recorded time series that vary on a much longer time scale than the ultradian oscillations that are being considered. This is necessary, since our model cannot describe such long-term fluctuations. Specifically, independently of the model parameter chosen, simulated traces from the Chemical Langevin Equation (eqs. (2) and (3)) do not include long-term trends. Hence, detrending prevents any bias that the presence of a long-term trend in the data may introduce to the parameter inference. When the

algorithm will be applied to data from other transcription factors, we recommend to exclude data that contains trends with timescales that are longer than the fluctuations and oscillations that are expected to emerge from the auto-negative feedback, in line with previous detrending recommendations[73;74]. Presumably, variations in the long-term trend of the data stem from a time-dependence of one or multiple of the model parameters due to regulatory processes that our model does not account for. Hence, future improvements to our algorithm may be developed where the temporal variation of model parameters is inferred, instead of one static value.

When applying our inference method to experimental data (Figure 4), we relied on previously reported values for the measurement variance, Σ_ϵ, in the data set that we considered[2]. When users seek to apply our algorithm to other data where previously published values are not available for Σ_ϵ, this parameter can be inferred following the same procedure as reported in Manning et al. (2019).

Our algorithm opens up the investigation of research problems, such as cell-to-cell heterogeneity in dynamic gene expression, which would previously not have been accessible. In future applications, our algorithm may provide a non-invasive method to measure the kinetic parameters of the gene of interest, such as the translation and transcription rates, or properties of the gene's promoter, which are described by the repression threshold and Hill coefficient parameters in our model. On experimental data sets where multiple, qualitatively different dynamics are observed[75–77], our algorithm may provide insights into the mechanistic origin of these different dynamics, by identifying differences in inferred parameter values between the observed cells or cell populations. In order to classify whether observed differences between posterior distributions are significant, one can construct the posterior distribution describing the difference between parameter values from both cells or populations, and test whether the credible interval of that distribution contains zero[78]. To facilitate such analysis, our method may for example be combined with clustering algorithms on the time series data, such as Gaussian mixture modelling. Since different dynamic patterns of gene expression have been observed in multiple studies of auto-repressing transcription factors[2;3], we anticipate that these approaches will spark new scientific investigations.

Throughout, we have assumed that measurements in the form of protein copy numbers per nucleus are available over time. To collect such data, it is necessary to combine endogenous fluorescent

reporters with FCS in order to translate reporter intensity values to molecule concentrations. Future versions of our algorithm may be applicable to data where FCS is not available, if one extends our measurement model (F, Section 2.2) to include an unknown, linear scaling parameter between protein copy numbers and imaged intensity values.

We highlight that the impact of this work is not limited to a single gene in a single model system. The conceptual framework and derivations described here are applicable to any system which can be described by delayed stochastic differential equations, although there may be computational limitations as model sizes increase.

Ethics statement

Animal experimentation: All animal work was performed under regulations set out by the UK Home Office Legislation under the 1986 United Kingdom Animal Scientific Procedures Act.

Data accessibility

All code and data used in this article are freely available on our github repository, https://github.com/kursawe/hesdynamics.

Authors' contributions

J.B., C.M., M.R., N.P. and J.K. conceived the study and contributed to the manuscript. J.B. developed and implemented the algorithm, ran the simulations, performed the analysis, and produced the figures. J.K. contributed to the algorithm development. M.R., N.P. and J.K. supervised the study. J.B. and J.K. co-wrote the manuscript with input from C.M., M.R. and N.P.. C.M. provided microscopy data and FCS analysis. All authors gave final approval for publication and agree to be held accountable for the work performed therein.

Competing interests

We declare we have no competing interests.

Funding

This work was supported by a Wellcome Trust Four-Year PhD Studentship in Basic Science to J.B. (219992/Z/19/Z) and a Wellcome Trust Senior Research Fellowship to N.P. (090868/Z/09/Z). C.M. was supported by a Sir Henry Wellcome Fellowship (103986/Z/14/Z) and University of Manchester Presidential Fellowship. M.R.'s work was supported by a Wellcome Trust Investigator Award (204832/B/16/Z).

Acknowledgements

The authors would like to acknowledge the use of the Computational Shared Facility at The University of Manchester.

References

[1] Imayoshi, I.; Isomura, A.; Harima, Y.; Kawaguchi, K.; Kori, H.; Miyachi, H.; Fujiwara, T.; Ishidate, F.; Kageyama, R. Oscillatory control of factors determining multipotency and fate in mouse neural progenitors. *Science* **2013**, *342*, 1203–1208.

[2] Manning, C. S.; Biga, V.; Boyd, J.; Kursawe, J.; Ymisson, B.; Spiller, D. G.; Sanderson, C. M.; Galla, T.; Rattray, M.; Papalopulu, N. Quantitative single-cell live imaging links HES5 dynamics with cell-state and fate in murine neurogenesis. *Nature communications* **2019**,

[3] Soto, X.; Biga, V.; Kursawe, J.; Lea, R.; Doostdar, P.; Thomas, R.; Papalopulu, N. Dynamic properties of noise and Her6 levels are optimized by miR-9, allowing the decoding of the Her6 oscillator. *The EMBO Journal* **2020**, *39*, 1–23.

[4] Bansod, S.; Kageyama, R.; Ohtsuka, T. Hes5 regulates the transition timing of neurogenesis and gliogenesis in mammalian neocortical development. *Development* **2017**, *144*, 3156–3167.

[5] Tan, S. L.; Ohtsuka, T.; González, A.; Kageyama, R. MicroRNA9 regulates neural stem cell differentiation by controlling Hes1 expression dynamics in the developing brain. *Genes to Cells* **2012**, *17*, 952–961.

[6] Kobayashi, T.; Kageyama, R. *Current Topics in Developmental Biology*, 1st ed.; Elsevier Inc., 2014; Vol. 110; pp 263–283.

[7] Purvis, J. E.; Karhohs, K. W.; Mock, C.; Batchelor, E.; Loewer, A.; Lahav, G. P53 Dynamics Control Cell Fate. *Science* **2012**, *336*, 1440–1444.

[8] Phillips, N. E.; Manning, C. S.; Pettini, T.; Biga, V.; Marinopoulou, E.; Stanley, P.; Boyd, J.; Bagnall, J.; Paszek, P.; Spiller, D. G.; White, M. R.; Goodfellow, M.; Galla, T.; Rattray, M.; Papalopulu, N. Stochasticity in the miR-9/Hes1 oscillatory network can account for clonal heterogeneity in the timing of differentiation. *eLife* **2016**, *5*, 1–33.

[9] Longo, D.; Hasty, J. Dynamics of single-cell gene expression. *Molecular Systems Biology* **2006**, *2*.

[10] Soroldoni, D.; Oates, A. C. Live transgenic reporters of the vertebrate embryo's Segmentation Clock. *Current Opinion in Genetics and Development* **2011**, *21*, 600–605.

[11] Yang, N. et al. Quantitative live imaging of Venus::BMAL1 in a mouse model reveals complex dynamics of the master circadian clock regulator. *PLoS Genetics* **2020**, *16*, 1–24.

[12] Pepe-Mooney, B. J.; Dill, M. T.; Alemany, A.; Ordovas-Montanes, J.; Matsushita, Y.; Rao, A.; Sen, A.; Miyazaki, M.; Anakk, S.; Dawson, P. A.; Ono, N.; Shalek, A. K.; van Oudenaarden, A.; Camargo, F. D. Single-Cell Analysis of the Liver Epithelium Reveals Dynamic Heterogeneity and an Essential Role for YAP in Homeostasis and Regeneration. *Cell Stem Cell* **2019**, *25*, 23–38.e8.

[13] Alber, A. B.; Paquet, E. R.; Biserni, M.; Naef, F.; Suter, D. M. Single Live Cell Monitoring of Protein Turnover Reveals Intercellular Variability and Cell-Cycle Dependence of Degradation Rates. *Molecular Cell* **2018**, *71*, 1079–1091.e9.

[14] Novák, B.; Tyson, J. J. Design principles of biochemical oscillators. *Nature Reviews Molecular Cell Biology* **2008**, *9*, 981–991.

[15] Alon, U. *An Introduction to Systems Biology: Design Principles of Biological Circuits*, 2nd ed.; Chapman and Hall/CRC, 2019; p 342.

[16] Chou, J.; Provot, S.; Werb, Z. GATA3 in development and cancer differentiation: Cells GATA have it! *Journal of Cellular Physiology* **2010**, *222*, 42–49.

[17] Chang, P. J.; Hsiao, Y. L.; Tien, A. C.; Li, Y. C.; Pi, H. Negative-feedback regulation of proneural proteins controls the timing of neural precursor division. *Development* **2008**, *135*, 3021–3030.

[18] Iwasaki, T.; Takiguchi, R.; Hiraiwa, T.; Yamada, T. G.; Yamazaki, K.; Hiroi, N. F.; Funahashi, A. Neural Differentiation Dynamics Controlled by Multiple Feedback Loops in a Comprehensive Molecular Interaction Network. *Processes* **2020**, *8*.

[19] Schwanhüusser, B.; Busse, D.; Li, N.; Dittmar, G.; Schuchhardt, J.; Wolf, J.; Chen, W.; Selbach, M. Global quantification of mammalian gene expression control. *Nature* **2011**, *473*, 337–342.

[20] Honkela, A.; Peltonen, J.; Topa, H.; Charapitsa, I.; Matarese, F.; Grote, K.; Stunnenberg, H. G.; Reid, G.; Lawrence, N. D.; Rattray, M. Genome-wide modeling of transcription kinetics reveals patterns of RNA production delays. *Proceedings of the National Academy of Sciences of the United States of America* **2015**, *112*, 13115–13120.

[21] Goodfellow, M.; Phillips, N. E.; Manning, C.; Galla, T.; Papalopulu, N. MicroRNA input into a neural ultradian oscillator controls emergence and timing of alternative cell states. *Nature Communications* **2014**, *5*.

[22] Monk, N. A. Oscillatory expression of Hes1, p53, and NF-κB driven by transcriptional time delays. *Current Biology* **2003**, *13*, 1409–1413.

[23] Lewis, J. Autoinhibition with transcriptional delay: A simple mechanism for the zebrafish somitogenesis oscillator. *Current Biology* **2003**, *13*, 1398–1408.

[24] Liepe, J.; Kirk, P.; Filippi, S.; Toni, T.; Barnes, C. P.; Stumpf, M. P. A framework for parameter estimation and model selection from experimental data in systems biology using approximate Bayesian computation. *Nature Protocols* **2014**, *9*, 439–456.

[25] Babtie, A. C.; Stumpf, M. P. How to deal with parameters for whole-cell modelling. *Journal of the Royal Society Interface* **2017**, *14*.

[26] Warne, D. J.; Baker, R. E.; Simpson, M. J. Simulation and inference algorithms for stochastic biochemical reaction networks: From basic concepts to state-of-the-art. *Journal of the Royal Society Interface* **2019**, *16*.

[27] Browning, A. P.; Warne, D. J.; Burrage, K.; Baker, R. E.; Simpson, M. J. Listen to the noise: identifiability analysis for stochastic differential equation models in systems biology. *bioRxiv* **2020**, 2020.08.10.245233.

[28] McAdams, H. H.; Arkin, A. It's a noisy business! Genetic regulation at the nanomolar scale. *Trends in Genetics* **1999**, *15*, 65–69.

[29] Toni, T.; Welch, D.; Strelkowa, N.; Ipsen, A.; Stumpf, M. P. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface* **2009**, *6*, 187–202.

[30] Gómez-Schiavon, M.; Chen, L. F.; West, A. E.; Buchler, N. E. BayFish: Bayesian inference of transcription dynamics from population snapshots of single-molecule RNA FISH in single cells. *Genome Biology* **2017**, *18*, 1–12.

[31] Vallejos, C. A.; Richardson, S.; Marioni, J. C. Beyond comparisons of means: Understanding changes in gene expression at the single-cell level. *Genome Biology* **2016**, *17*, 1–14.

[32] Cao, Z.; Grima, R. Accuracy of parameter estimation for auto-regulatory transcriptional feedback loops from noisy data. *Journal of the Royal Society Interface* **2019**, *16*.

[33] Jenkins, D. J.; Finkenstädt, B.; Rand, D. A. A temporal switch model for estimating transcriptional activity in gene expression. *Bioinformatics (Oxford, England)* **2013**, *29*, 1158–1165.

[34] Finkenstädt, B.; Woodcock, D. J.; Komorowski, M.; Harper, C. V.; Davis, J. R.; White, M. R.; Rand, D. A. Quantifying intrinsic and extrinsic noise in gene transcription using the linear noise approximation: An application to single cell data. *Annals of Applied Statistics* **2013**, *7*, 1960–1982.

[35] Munsky, B.; Neuert, G.; Van Oudenaarden, A. Using gene expression noise to understand gene regulation. *Science* **2012**, *336*, 183–187.

[36] Ramachandra Murthy, K.; Singh, S.; Tuck, D.; Varadan, V. *Handbook of Statistics*, 1st ed.; Elsevier B.V., 2019; Vol. 40; pp 355–404.

[37] Beaumont, M. A.; Zhang, W.; Balding, D. J. Approximate Bayesian computation in population genetics. *Genetics* **2002**, *162*, 2025–2035.

[38] Fearnhead, P.; Prangle, D. Constructing summary statistics for approximate Bayesian computation: Semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society. Series B: Statistical Methodology* **2012**, *74*, 419–474.

[39] Hastings, W. K. Monte carlo sampling methods using Markov chains and their applications. *Biometrika* **1970**, *57*, 97–109.

[40] Hoffman, M. D.; Gelman, A. The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research* **2014**, *15*, 1593–1623.

[41] Duane, S.; Kennedy, A. D.; Pendleton, B. J.; Roweth, D. Hybrid Monte Carlo. *Physics Letters B* **1987**, *195*, 216–222.

[42] Betancourt, M. A Conceptual Introduction to Hamiltonian Monte Carlo. **2017**,

[43] Roberts, G. O.; Tweedie, R. L. Exponential Convergence of Langevin Distributions and Their Discrete Approximations. *Bernoulli* **1996**, *2*, 341.

[44] Girolami, M.; Calderhead, B. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society* **2011**, *73*, 123–214.

[45] Lillacci, G.; Khammash, M. Parameter estimation and model selection in computational biology. *PLoS Computational Biology* **2010**, *6*.

[46] Mbalawata, I. S.; Särkkä, S.; Haario, H. Parameter estimation in stochastic differential equations with Markov chain Monte Carlo and non-linear Kalman filtering. *Computational Statistics* **2013**, *28*, 1195–1223.

[47] Kojima, K.; Fujita, A.; Shimamura, T.; Imoto, S.; Miyano, S. Estimation of nonlinear gene regulatory networks via L1 regularized NVAR from time series gene expression data. *Genome informatics. International Conference on Genome Informatics* **2008**, *20*, 37–51.

[48] Fan, M.; Kuwahara, H.; Wang, X.; Wang, S.; Gao, X. Parameter estimation methods for gene circuit modeling from time-series mRNA data: A comparative study. *Briefings in Bioinformatics* **2015**, *16*, 987–999.

[49] Moles, C. G.; Mendes, P.; Banga, J. R. Parameter estimation in biochemical pathways: A comparison of global optimization methods. *Genome Research* **2003**, *13*, 2467–2474.

[50] Clerx, M.; Robinson, M.; Lambert, B.; Lei, C. L.; Ghosh, S.; Mirams, G. R.; Gavaghan, D. J. Probabilistic Inference on Noisy Time Series (PINTS). *Journal of Open Research Software* **2019**, *7*.

[51] Minas, G.; Momiji, H.; Jenkins, D. J.; Costa, M. J.; Rand, D. A.; Finkenstädt, B. ReTrOS: A MATLAB toolbox for reconstructing transcriptional activity from gene and protein expression data. *BMC Bioinformatics* **2017**, *18*, 1–11.

[52] Calderazzo, S.; Brancaccio, M.; Finkenstädt, B. Filtering and Inference for stochastic oscillators with distributed delays. *Bioinformatics* **2018**,

[53] Hatakeyama, J. Hes genes regulate size, shape and histogenesis of the nervous system by control of the timing of neural stem cell differentiation. *Development* **2004**, *131*, 5539–5550.

[54] Gillespie, D. T. The chemical Langevin equation. *The Journal of Chemical Physics* **2000**, *113*, 297–306.

[55] Galla, T. Intrinsic fluctuations in stochastic delay systems: Theoretical description and application to a simple model of gene regulation. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* **2009**, *80*, 1–9.

[56] Brett, T.; Galla, T. Gaussian approximations for stochastic systems with delay: Chemical Langevin equation and application to a Brusselator system. *The Journal of Chemical Physics* **2014**, *140*, 124112.

[57] Hirata, H.; Yoshiura, S.; Ohtsuka, T.; Bessho, Y.; Harada, T.; Yoshikawa, K.; Kageyama, R. Oscillatory Expression of the bHLH Factor Hes1 Regulated by a Negative Feedback Loop. *Science* **2002**, *298*, 840–843.

[58] Weiss, J. N. The Hill equation revisited: uses and misuses. *The FASEB Journal* **1997**, *11*, 835–841.

[59] Särkkä, S. *Bayesian Filtering and Smoothing*; Cambridge University Press, 2013.

[60] Singer, H. Parameter estimation of nonlinear stochastic differential equations: Simulated maximum likelihood versus extended kalman filter and itô-taylor expansion. *Journal of Computational and Graphical Statistics* **2002**, *11*, 972–995.

[61] Edward Rasmussen, C.; K. I. Williams, C. *Gaussian Processes for Machine Learning*; The MIT Press, 2006.

[62] Pedregosa, F.; Grisel, O.; Weiss, R.; Passos, A.; Brucher, M.; Varoquax, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Brucher, M. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

[63] Roesch, E.; Stumpf, M. P. Parameter inference in dynamical systems with co-dimension 1 bifurcations. *Royal Society Open Science* **2019**, *6*.

[64] Momiji, H.; Monk, N. A. Dissecting the dynamics of the Hes1 genetic oscillator. *Journal of Theoretical Biology* **2008**, *254*, 784–798.

[65] Zhuang, K.; Zhu, H. Stability and bifurcation analysis in a model of Hes1 selfregulation with time delay. *World Academy of Science, Engineering and Technology* **2010**, *43*, 469–471.

[66] Femino, A. M.; Fay, F. S.; Fogarty, K.; Singer, R. H. Visualization of single RNA transcripts in situ. *Science* **1998**, *280*, 585–590.

[67] Raj, A.; van den Bogaard, P.; Rifkin, S. A.; van Oudenaarden, A.; Tyagi, S. Imaging individual mRNA molecules using multiple singly labeled probes. *Nature Methods* **2008**, *5*, 877–879.

[68] Bonev, B.; Stanley, P.; Papalopulu, N. MicroRNA-9 Modulates Hes1 Ultradian Oscillations by Forming a Double-Negative Feedback Loop. *Cell Reports* **2012**, *2*, 10–18.

[69] Tunnacliffe, E.; Chubb, J. R. What Is a Transcriptional Burst? *Trends in Genetics* **2020**, *36*, 288–297.

[70] Gillespie, D. T. Stochastic Simulation of Chemical Kinetics. *Annual Review of Physical Chemistry* **2007**, *58*, 35–55.

[71] Andrieu, C.; Thoms, J. A tutorial on adaptive MCMC. *Statistics and Computing* **2008**, *18*, 343–373.

[72] Roberts, G. O.; Rosenthal, J. S. Optimal Scaling for Various Metropolis-Hastings Algorithms. *Statistical Science* **2001**, *16*, 351–367.

[73] Phillips, N. E.; Manning, C.; Papalopulu, N.; Rattray, M. Identifying stochastic oscillations in single-cell live imaging time series using Gaussian processes. *PLoS Computational Biology* **2017**, *13*, 1–30.

[74] Mönke, G.; Sorgenfrei, F. A.; Schmal, C.; Granada, A. E. Optimal time frequency analysis for biological data - pyBOAT. *bioRxiv* **2020**, 27.

[75] D'Urso, A.; Brickner, J. H. Epigenetic transcriptional memory. *Current Genetics* **2017**, *63*, 435–439.

[76] Elowitz, M. B.; Levine, A. J.; Siggia, E. D.; Swain, P. S. Stochastic gene expression in a single cell. *Science* **2002**, *297*, 1183–1186.

[77] Godwin, S.; Ward, D.; Pedone, E.; Homer, M.; Fletcher, A. G.; Marucci, L. An extended model for culture-dependent heterogenous gene expression and proliferation dynamics in mouse embryonic stem cells. *npj Systems Biology and Applications* **2017**, *3*, 1–12.

[78] Gelman, A.; Hill, J.; Vehtari, A. *Regression and Other Stories*; Analytical Methods for Social Research; Cambridge University Press, 2020.

[79] Powell, M. J. D. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal* **1964**, *7*, 155–162.

[80] Haario, H.; Saksman, E.; Tamminen, J. An adaptive Metropolis algorithm. *Bernoulli* **2001**, *7*, 223–242.

[81] Roberts, G. O.; Stramer, O.; Science, A. Langevin Diffusions and Metropolis-Hastings Algorithms. *Computing* **2003**, 337–357.

[82] Durmus, A.; Roberts, G. O.; Vilmart, G.; Zygalakis, K. C. Fast langevin based algorithm for MCMC in high dimensions. *Annals of Applied Probability* **2017**, *27*, 2195–2237.

[83] Singh, J.; Padgett, R. A. Rates of in situ transcription and splicing in large human genes. *Nature structural and molecular biology* **2009**, *16*, 1128–1133.

[84] M. Suter, D.; Molina, N.; Naef, F.; Schibler, U. Origins and consequences of transcriptional discontinuity. *Current Opinion in Cell Biology* **2011**, *23*, 657–662.

[85] Gelman, A.; Rubin, D. B.; Gelman, A.; Rubin, D. B. Inference from Iterative Simulation Using Multiple Sequences Linked references are available on JSTOR for this article : Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science* **1992**, *7*, 457–472.

[86] Vehtari, A.; Gelman, A.; Simpson, D.; Carpenter, B.; Bürkner, P.-C. Rank-Normalization, Folding, and Localization: An Improved R for Assessing Convergence of MCMC. *Bayesian Analysis* **2020**, 1–27.

[87] Alonso, D.; McKane, A. J.; Pascual, M. Stochastic amplification in epidemics. *Journal of the Royal Society Interface* **2007**, *4*, 575–582.

[88] Pollard, S. M. In vitro expansion of fetal neural progenitors as adherent cell lines. *Methods in Molecular Biology* **2013**, *1059*, 13–24.

Supplementary Information

Here we provide full derivations for the prediction and update step in the non-linear, delay-adapted Kalman filter. We detail the MCMC sampling algorithms discussed in the main text, as well as our chosen method for adaptive step size selection within the MCMC. Then, we show how the gradient of the likelihood can be computed within the Kalman filter. We further describe our process for choosing prior distributions, and for assessing convergence of our Markov chains, before discussing how oscillation quality of time series data is quantified, and how the likelihood function is extended to account for data on mRNA copy numbers. Finally, we give details about the data collection and analysis. Supplementary figures are provided at the end.

S.1 Kalman filter prediction and update step

In order to predict the probability distribution over the state space $\mathbf{X}$ at observation k, it is necessary to calculate ρ_t and P_t as defined in the main manuscript (Section 2.2). Here, we show how we derive delay differential equations for both quantities, closely following Calderazzo et al. (2018).

Prediction Step We first rewrite eqs. (2) and (3) in the general form of a vector-valued, delayed stochastic differential equation:

$$\begin{aligned} d\mathbf{X}(t) &= g(\mathbf{X}(t))dt + f\left(\mathbf{X}(t-\tau)\right)dt \\ &\quad + \sqrt{l(\mathbf{X}(t)) + q\left(\mathbf{X}(t-\tau)\right)}dB(t), \end{aligned} \tag{10}$$

where $B(t)$ is a 2-dimensional Wiener process, and

$$g(\mathbf{X}(t)) = \begin{bmatrix} -\mu_m m(t) \\ \alpha_p m(t) - \mu_p p(t) \end{bmatrix},$$

$$f(\mathbf{X}(t-\tau)) = \begin{bmatrix} \alpha_m f(p(t-\tau)) \\ 0 \end{bmatrix},$$

$$l(\mathbf{X}(t)) = \begin{bmatrix} \mu_m m(t) & 0 \\ 0 & \alpha_p m(t) + \mu_p p(t) \end{bmatrix},$$

$$q(\mathbf{X}(t-\tau)) = \begin{bmatrix} \alpha_m f(p(t-\tau)) & 0 \\ 0 & 0 \end{bmatrix}.$$

Next, we aim to use this definition to derive delay differential equations for ρ and P. To do so, we employ Taylor expansions. The first order Taylor expansions of $g(\cdot)$ and $f(\cdot)$ about ρ_t are

$$g(X_t) \approx g(\rho_t) + J_g(\rho_t) \cdot (X_t - \rho_t),$$

and

$$f(X_{t-\tau}) \approx f(\rho_{t-\tau}) + J_f(\rho_{t-\tau}) \cdot (X_{t-\tau} - \rho_{t-\tau}).$$

We can hence write

$$X_{t+\delta_t} = X_t + g(X_t)\delta_t + f(X_{t-\tau})\delta_t + \sqrt{l(X_t) + q(X_{t-\tau})}\mathcal{N}(0, \delta_t) + \mathcal{O}(\delta_t^2), \tag{11}$$

where $\mathcal{N}(0, \delta_t)$ is a Gaussian distribution with mean zero and variance δ_t. Inserting this into the definition of ρ_t, we find

$$d\rho_t = g(\rho_t)dt + f(\rho_{t-\tau})dt. \tag{12}$$

Similarly, we insert eq. (11) into the definition of P to find

$$\begin{aligned} dP_t = &\left[J_g(\rho_t)P_t + P_t^T J_g(\rho_t)^T\right] dt \\ &+ \left[J_f(\rho_{t-\tau})P_{t-\tau,t}\right] dt \\ &+ \left[P_{t,t-\tau} J_f(\rho_{t-\tau})^T\right] dt \\ &+ A\left(\rho_t, \rho_{t-\tau}\right) dt, \end{aligned} \tag{13}$$

where $A(\rho_t, \rho_{t-\tau}) = l(\rho_t) + q(\rho_{t-\tau})$ and $P_{t,s} = \mathrm{Cov}(\mathbf{X}(t), \mathbf{X}(s) \mid y_{0:t-1})$.

Lastly, we have

$$dP_{s,t} = P_{s,t} J_g^T(\rho_t)dt + P_{s,t-\tau} J_f^T(\rho_{t-\tau})dt. \tag{14}$$

We can then use a standard forward Euler method with a time step which represents the absolute time difference between consecutive hidden states, to numerically integrate these delay differential equations between observations. Before the first prediction step can be executed, it is necessary to define initial and past conditions for these delay differential equations. The state space mean is initialised at negative times with the steady state solutions to the deterministic approximation of the model, i.e. eqs. (2) and (3) of the main manuscript without their respective noise terms. At steady state, we have

$$p^* = \frac{\alpha_m \alpha_p}{\mu_m \mu_p} f(p^*) \tag{15}$$

and

$$m^* = \frac{\mu_p}{\alpha_p} p^*. \tag{16}$$

The state space variance is initialised with a multiple of this mean. Specifically, the variance of the mRNA is initialised at 20 times its mean value, and the variance of the protein is initialised at 100 times its mean value. These are chosen to reflect the variance of protein and mRNA expression from the Chemical Langevin Equations. For simplicity, off-diagonal elements of the state space covariance are initialised at zero. We expect the influence of these initial and past conditions to diminish after the first update step has been applied and as the number of data points increases.

Update Step For our update step we derive an expression for the mean and variance of the state space distribution $\pi(x_{t-\tau:t} \mid y_{0:t})$, denoted $\rho^*_{t-\tau:t}$ and $P^*_{t-\tau:t}$ respectively. That is, the likelihood of our state space estimates from the past time $t-\tau$ to the current time, t, given all of our current observations. This is necessary in order to accurately predict the state space distribution at the next observation time point, $\pi(x_{t+\Delta t} \mid y_{0:t})$, as past states can affect future states due to the presence of delays.

Assuming approximate Gaussianity of the joint distribution $\pi(x_{t+\Delta t}, y_{t+\Delta t} \mid y_{0:t})$, following

Särkkä (2013), we can obtain the following expression for the updated state space mean and variance:

$$\rho^*_{t-\tau:t} = \rho_{t-\tau:t} + K_{t-\tau:t}\left(y_t - F\rho_t\right), \tag{17}$$

$$P^*_{t-\tau:t} = P_{t-\tau:t} - K_{t-\tau:t} S_t K^T_{t-\tau:t}, \tag{18}$$

$$K_{t-\tau:t} = P_{t-\tau:t,t} F^T S_t^{-1}, \tag{19}$$

where $S_t = F P_t F^T + \Sigma_\epsilon$.

S.2 Markov Chain Monte Carlo sampling algorithms

For completeness, we provide here the algorithms for the Metropolis-Hasting random walk, and the Metropolis-adjusted Langevin Algorithm.

Metropolis-Hastings Random Walk

We initialise the algorithm with a set of the model parameters, $\boldsymbol{\theta}_0$, chosen by minimising the log likelihood function using Powell's method[79]. In the case of our model, each parameter vector is composed as $\boldsymbol{\theta} = [P_0, h, \mu_m, \mu_p, \alpha_m, \alpha_p, \tau]^T$. At each iteration, i, we propose a combination of parameters, $\boldsymbol{\theta}'$, from a symmetric distribution, $q(\cdot)$, centered around the current parameters, $\boldsymbol{\theta}_i$. In this paper we use the normal distribution $\mathcal{N}(\boldsymbol{\theta}, \epsilon\Sigma)$, where ϵ is called the step size, and Σ is the proposal covariance matrix. An acceptance ratio, $\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, defined by

$$\begin{aligned}\alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}') &= \min\left\{1, \frac{\pi(\boldsymbol{\theta}' \mid \mathbf{y})}{\pi(\boldsymbol{\theta}_m \mid \mathbf{y})}\right\} \\ &= \min\left\{1, \frac{\pi(\mathbf{y} \mid \boldsymbol{\theta}')\pi(\boldsymbol{\theta}')}{\pi(\mathbf{y} \mid \boldsymbol{\theta}_i)\pi(\boldsymbol{\theta}_m)}\right\}\end{aligned}$$

is calculated, and a uniform random number, u, is generated on $[0, 1]$.

1. If $u \leq \alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, we *accept* the proposal $\boldsymbol{\theta}'$ and set $\boldsymbol{\theta}' = \boldsymbol{\theta}_{i+1}$.

2. If $u \geq \alpha(\boldsymbol{\theta}_i, \boldsymbol{\theta}')$, we *reject* the proposal $\boldsymbol{\theta}'$ and set $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{i+1}$.

The distribution of random draws $\{\boldsymbol{\theta}_i \mid i \in \{0, 1, \ldots, N_s\}\}$ converges to our posterior distribu-

tion, $\pi(\boldsymbol{\theta} \mid \mathbf{y})$, as $N_s \to \infty$. In practice, the number of iterations necessary for accurate inference depends on a number of factors, including the complexity of the problem, the number of parameters to be inferred, and the relationship between the posterior and proposal distributions. We outline our choice of N_s in Section S.3

Metropolis-adjusted Langevin Algorithm (MALA)

Whilst the MH algorithm is relatively simple to understand and implement, it can fail to identify the posterior if it has a non-trivial, curved correlation structure[80], or if it is multi-modal[81]. It can also become slow when dimensionality is high, i.e. when there are a large number of parameters to infer. In these cases the algorithm typically requires a very large number of samples, which is time consuming, especially if the likelihood calculation is slow. One way to address this is to use a sampling algorithm that uses the gradient of the posterior distribution, as well as it's absolute values, to draw new samples. The gradient can help guide the search towards the modes of the posterior distribution, making for a more efficient algorithm which typically needs fewer samples. A number of different MCMC approaches utilise the gradient of the distribution, such as MALA, and the NO-U-Turn sampler[40], which is an extension on the original Hamiltonian Monte Carlo (HMC)[42].

We use MALA, since it is well-established that it has better convergence properties than the MH algorithm[82]. Our algorithm for MALA iterates over the same steps as the MH algorithm, with an adjusted expression for the proposed parameter combination. Where MH uses the proposal

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \sqrt{\epsilon}\Sigma^{1/2}\xi, \tag{20}$$

MALA instead uses

$$\boldsymbol{\theta}' = \boldsymbol{\theta} - (\epsilon/2)\Sigma U(\boldsymbol{\theta}) + \sqrt{\epsilon}\Sigma^{1/2}\xi, \tag{21}$$

where $\xi \sim \phi(0, I_d)$, with I_d denoting the d-dimensional identity matrix, d is the dimension of the posterior distribution, and $U(\boldsymbol{\theta}) = -\nabla \log \pi(\mathbf{y} \mid \boldsymbol{\theta})$.

S.3 Implementation of adaptive sampling strategies

We seek to generate N_s independent samples that accurately characterise our posterior distribution, but samples are dependent on the starting point of the Markov chain. As increasingly many samples are generated, this dependence diminishes. The correlation between samples can be reduced if an appropriate covariance matrix is chosen for the proposal distribution. One suitable covariance matrix is the covariance of the target posterior distribution, which may be estimated from samples of the MCMC itself. To do so, we *warm up* the Markov chains in two steps by drawing N_{w_1} samples and discarding the first N_{b_1} samples as *burn-in*. With the remaining samples we construct a proposal covariance matrix, and repeat the process with N_{w_2} samples and a burn-in of size N_{b_2}. The adaptive MCMC is then started using the covariance matrix estimated in this way as the proposal covariance matrix. We adaptively update our step size at iteration i, ϵ_i, using a modified version of Algorithm 4 from Andrieu and Thoms (2008).

At each iteration i, the step size is updated using

$$\begin{aligned} \gamma_1 &= \frac{1}{i^{c_1}}, \\ \gamma_2 &= c_0 \gamma_1, \\ \log(\epsilon_{i+1}^2) &= \log(\epsilon_i^2) + \gamma_2(\alpha_i - 0.574), \end{aligned} \tag{22}$$

where $c_0 = 1$, $c_1 = \log(10)/\log(N_s/5)$, and α_i is the current acceptance probability. We use 0.574 as the target acceptance rate for MALA, as shown above, and 0.234 for MH [72].

In our experience, this approach results in far better performance than without warm up or adaptively updating the step size. In practice, we found that consistent posteriors are obtained by running 8 Markov chains in parallel, where each chain has $N_s = 80,000$, $N_{w_1} = 0.3 \times N_s$, $N_{w_2} = 0.7 \times N_s$, and $N_{b_i} = 0.5 \times N_{w_i}$, $i = 1, 2$. Warm up samples from all eight chains are pooled to calculate the covariance matrix for the final run of MCMC. This approach to warm up is used in the same way when either sampler, MH or MALA, is chosen.

S.4 Computing the gradient of the log-posterior

The Kalman filter has previously been used within gradient based sampling schemes[46], which illustrated that it is possible to calculate the gradient with an iterative scheme similar to the scheme for the direct likelihood calculation. In the following, we show that these previous approaches can be extended to derive the gradient of the delay-mediated Kalman filter used here.

For MALA, we compute the derivative of the negative log-posterior $U = -\log(\pi(\boldsymbol{\theta}|\mathbf{y})$ with respect to the parameters, $\boldsymbol{\theta}$,

$$\begin{aligned}
\frac{\partial U(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \frac{\partial(-\log(\pi(\mathbf{y} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})))}{\partial \boldsymbol{\theta}} \\
&= \frac{\partial\left(-\log\left(\prod_{t\in\mathcal{T}} \phi(y_t; F\rho_t, FP_tF^T + \Sigma_\epsilon)\pi(\boldsymbol{\theta})\right)\right)}{\partial \boldsymbol{\theta}} \\
&= -\frac{\partial}{\partial \boldsymbol{\theta}} \log\left(\prod_{t\in\mathcal{T}} \frac{\exp\left(-\frac{1}{2}(y_t - F\rho_t)^T(FP_tF^T + \Sigma_\epsilon)^{-1}(y_t - F\rho_t)\right)\pi(\boldsymbol{\theta})}{\sqrt{\det(2\pi(FP_tF^T + \Sigma_\epsilon))}}\right) \\
&= \frac{\partial}{\partial \boldsymbol{\theta}}\left[\sum_{t\in\mathcal{T}} \frac{1}{2}\log\left(\det(2\pi S_t(\boldsymbol{\theta}))\right) + \right. \\
&\qquad \left. \frac{1}{2}\sum_{t\in\mathcal{T}}(y_t - F\rho_t(\boldsymbol{\theta}))^T(S_t(\boldsymbol{\theta}))^{-1}(y_t - F\rho_t(\boldsymbol{\theta})) - \log(\pi(\boldsymbol{\theta}))\right],
\end{aligned}$$

where $S = FP_tF^T + \Sigma_\epsilon$ has been used in the last line, and the dependence of ρ and S on $\boldsymbol{\theta}$ has been emphasized. $\mathcal{T}$ denotes the set of time points for each observation, $\{i \cdot l \mid i \in \{0, 1, \ldots, n-1\}\}$. For a single parameter this simplifies to

$$\begin{aligned}
\frac{\partial U(\boldsymbol{\theta})}{\partial \theta_k} &= \frac{1}{2}\sum_{t\in\mathcal{T}}\left(\mathrm{Tr}\left(S_t^{-1}\frac{\partial S_t}{\partial \theta_k}\right)\right) - \frac{1}{2}\sum_{t\in\mathcal{T}}\left(F\frac{\partial \rho_t}{\partial \theta_k}\right)^T S_t^{-1}(y_t - F\rho_t) \\
&\quad - \frac{1}{2}\sum_{t\in\mathcal{T}}(y_t - F\rho_t)^T S_t^{-1}\frac{\partial S_t}{\partial \theta_k}S_t^{-1}(y_t - F\rho_t) \\
&\quad - \frac{1}{2}\sum_{t\in\mathcal{T}}(y_t - F\rho_t)^T S_t^{-1}F\frac{\partial \rho_t}{\partial \theta_k} - \frac{\partial(\log(\pi(\boldsymbol{\theta})))}{\partial \theta_k}.
\end{aligned} \tag{23}$$

This expression can be explicitly calculated if $\mathrm{d}\rho_t/\mathrm{d}\theta_k$ and $\mathrm{d}P_t/\mathrm{d}\theta_k$ are known at each observation

time point t. These quantities are the predicted state space mean and variance in the prediction step of the Kalman filter. This implies that eq. (23) can be calculated if the prediction step is adjusted to predict the derivative of ρ and P in addition to their absolute values. To enable this adjustment of the prediction step, we derive differential equations for $\frac{\mathrm{d}}{\mathrm{d}t}(\mathrm{d}\rho_t/\mathrm{d}\theta_k)$ and $\frac{\mathrm{d}}{\mathrm{d}t}(\mathrm{d}P_t/\mathrm{d}\theta_k)$:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \rho_t}{\partial \theta_k}\right) &= \frac{\partial}{\partial \theta_k}\left(\frac{\mathrm{d}\rho_t}{\mathrm{d}t}\right) \\
&= \frac{\partial}{\partial \theta_k}\left(g(\rho_t) + f(\rho_{t-\tau})\right) \\
&= \frac{\partial g(\rho_t)}{\partial \rho_t}\frac{\partial \rho_t}{\partial \theta_k} + \frac{\partial g(\rho_t)}{\partial \theta_k} + \frac{\partial f(\rho_{t-\tau})}{\partial \rho_{t-\tau}}\frac{\partial \rho_{t-\tau}}{\partial \theta_k} + \frac{\partial f(\rho_{t-\tau})}{\partial \theta_k} \\
&= J_g(\rho_t)\frac{\partial \rho_t}{\partial \theta_k} + \frac{\partial g(\rho_t)}{\partial \theta_k} + J_f(\rho_{t-\tau})\frac{\partial \rho_{t-\tau}}{\partial \theta_k} + \frac{\partial f(\rho_{t-\tau})}{\partial \theta_k},
\end{aligned} \tag{24}$$

and

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial P_t}{\partial \theta_k}\right) &= \frac{\partial}{\partial \theta_k}\left(\frac{\mathrm{d}P_t}{\mathrm{d}t}\right) \\
&= \frac{\partial}{\partial \theta_k}\Big(J_g(\rho_t)P_t + P_t^T J_g(\rho_t)^T \\
&\qquad + J_f(\rho_{t-\tau})P_{t-\tau,t} + P_{t,t-\tau}J_f(\rho_{t-\tau})^T \\
&\qquad + l(\rho_t) + q(\rho_{t-\tau})\Big) \\
&= J_g(\rho_t)\frac{\partial P_t}{\partial \theta_k} + \frac{\partial P_t^T}{\partial \theta_k}J_g(\rho_t)^T + J_f(\rho_{t-\tau})\frac{\partial P_{t-\tau,t}}{\partial \theta_k} + \frac{\partial P_{t,t-\tau}}{\partial \theta_k}J_f(\rho_{t-\tau})^T \\
&+ \left(\sum_{j=1}^{2}\left(\frac{\partial J_g(\rho_{t_j})}{\partial \rho_{t_j}}\frac{\partial \rho_{t_j}}{\partial \theta_k}\right) + \frac{\partial J_g(\rho_t)}{\partial \theta_k}\right)P_t + P_t^T\left(\sum_{j=1}^{2}\left(\frac{\partial J_g(\rho_{t_j})}{\partial \rho_{t_j}}\frac{\partial \rho_{t_j}}{\partial \theta_k}\right) + \frac{\partial J_g(\rho_t)}{\partial \theta_k}\right)^T \\
&+ \left(\sum_{j=1}^{2}\left(\frac{\partial J_f(\rho_{t-\tau_j})}{\partial \rho_{t-\tau_j}}\frac{\partial \rho_{t-\tau_j}}{\partial \theta_k}\right) + \frac{\partial J_f(\rho_{t-\tau})}{\partial \theta_k}\right)P_{t-\tau,t} \\
&+ P_{t,t-\tau}\left(\sum_{j=1}^{2}\left(\frac{\partial J_f(\rho_{t-\tau_j})}{\partial \rho_{t-\tau_j}}\frac{\partial \rho_{t-\tau_j}}{\partial \theta_k}\right) + \frac{\partial J_f(\rho_{t-\tau})}{\partial \theta_k}\right)^T
\end{aligned}$$

$$+\left(\sum_{j=1}^{2}\left(\frac{\partial l(\rho_{t_j})}{\partial \rho_{t_j}}\frac{\partial \rho_{t_j}}{\partial \theta_k}\right)+\frac{\partial l(\rho_t)}{\partial \theta_k}\right)+\left(\sum_{j=1}^{2}\left(\frac{\partial q(\rho_{t_j})}{\partial \rho_{t_j}}\frac{\partial \rho_{t_j}}{\partial \theta_k}\right)+\frac{\partial q(\rho_t)}{\partial \theta_k}\right) \tag{25}$$

where ρ_{t_j} means the jth element of ρ_t. Similarly,

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial P_{s,t}}{\partial \theta_k}\right) &= \frac{\partial}{\partial \theta_k}\left(\frac{\mathrm{d}P_{s,t}}{\mathrm{d}t}\right) \\
&= \frac{\partial}{\partial \theta_k}\left(P_{s,t}J_g^T(\rho_t)+P_{s,t-\tau}J_f^T(\rho_{t-\tau})\right) \\
&= \frac{\partial P_{s,t}}{\partial \theta_k}J_g^T(\rho_t)+\frac{\partial P_{s,t-\tau}}{\partial \theta_k}J_f^T(\rho_{t-\tau}) \\
&+ P_{s,t}\left(\sum_{j=1}^{2}\left(\frac{\partial J_g(\rho_{t_j})}{\partial \rho_{t_j}}\frac{\partial \rho_{t_j}}{\partial \theta_k}\right)+\frac{\partial J_g(\rho_t)}{\partial \theta_k}\right)^T \\
&+ P_{s,t-\tau}\left(\sum_{j=1}^{2}\left(\frac{\partial J_f(\rho_{t-\tau_j})}{\partial \rho_{t-\tau_j}}\frac{\partial \rho_{t-\tau_j}}{\partial \theta_k}\right)+\frac{\partial J_f(\rho_{t-\tau})}{\partial \theta_k}\right)^T .
\end{aligned} \tag{26}$$

Similar to the delay differential equations for the state space mean and variance, it is necessary to define values for non-positive times for the delay differential equations provided here before the first prediction step can be executed. To do so, we need to provide derivatives of the initialisation of the state space mean and variance provided in Section S.1. We calculate the derivatives of p^* and m^* with respect to each parameter, θ_k, as follows:

$$\frac{\partial p^*}{\partial \theta_k} = \frac{-\frac{\partial}{\partial \theta_k}\left(\frac{\alpha_m\alpha_p}{\mu_m\mu_p}\right)f(p^*)-\left(\frac{\alpha_m\alpha_p}{\mu_m\mu_p}\right)\frac{\partial}{\partial \theta_k}\left(f(p^*)\right)}{\left(\frac{\alpha_m\alpha_p}{\mu_m\mu_p}\right)\frac{\partial}{\partial p^*}f(p^*)-1}, \tag{27}$$

$$\frac{\partial m^*}{\partial \theta_k} = \frac{\partial}{\partial \theta_k}\left(\frac{\mu_p}{\alpha_p}p^*\right). \tag{28}$$

S.5 Update step derivatives

The delay differential equations given by eqs. (24) to (26) can be used to predict the derivatives of the state space mean and variance with respect to each model parameter, using $\mathrm{d}\rho_t^*/\mathrm{d}\theta_k$ and

$\mathrm{d}P_t^*/\mathrm{d}\theta$ as initial (and past) conditions. We incorporate the integration of these ODEs into the prediction step of our Kalman filter. The derivatives of $\mathrm{d}\rho_t^*/\mathrm{d}\theta_k$ and $\mathrm{d}P_t^*/\mathrm{d}\theta$ at each observation can be calculated by extending the update step. Using eqs. (17) to (19), we have

$$\begin{aligned}
\frac{\partial \rho^*_{t-\tau:t}}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k}\left(\rho_{t-\tau:t} + K_{t-\tau:t}\left(y_t - F\rho_t\right)\right) \\
&= \frac{\partial \rho_{t-\tau:t}}{\partial \theta_k} + \frac{\partial K_{t-\tau:t}}{\partial \theta_k}\left(y_t - F\rho_t\right) - K_{t-\tau:t}F\frac{\partial \rho_t}{\partial \theta_k}, \qquad (29)\\
\frac{\partial P^*_{t-\tau:t}}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k}\left(P_{t-\tau:t} - K_{t-\tau:t}S_tK^T_{t-\tau:t}\right) \\
&= \frac{\partial P_{t-\tau:t}}{\partial \theta_k} - \frac{\partial K_{t-\tau:t}}{\partial \theta_k}S_tK^T_{t-\tau:t} - K_{t-\tau:t}\frac{\partial S_t}{\partial \theta_k}K^T_{t-\tau:t} - K_{t-\tau:t}S_t\left(\frac{\partial K_{t-\tau:t}}{\partial \theta_k}\right)^T, \qquad (30)\\
\frac{\partial K_{t-\tau:t}}{\partial \theta_k} &= \frac{\partial}{\partial \theta_k}\left(P_{t-\tau:t,t}F^TS_t^{-1}\right) \\
&= \frac{\partial P_{t-\tau:t,t}}{\partial \theta_k}F^TS_t^{-1} - P_{t-\tau:t,t}F^TS_t^{-1}\frac{\partial S_t}{\partial \theta_k}S_t^{-1} \\
&= \frac{\partial P_{t-\tau:t,t}}{\partial \theta_k}F^TS_t^{-1} - P_{t-\tau:t,t}F^TS_t^{-1}F\frac{\partial P_t}{\partial \theta_k}F^TS_t^{-1}. \qquad (31)
\end{aligned}$$

Here, $\frac{\partial \rho_{t-\tau:t}}{\partial \theta_k}$ and $\frac{\partial P_{t-\tau:t}}{\partial \theta_k}$ are the derivatives for the state space mean and variance that have been predicted in the previous prediction step. Hence, we can evaluate eq. (23) for each of the parameters iteratively by evaluating eqs. (24) to (26), (29) and (30) at each update step, and integrating eqs. (12) to (14) within each prediction step of the Kalman filter.

S.6 Choice of priors and reparameterisation of variables

Bayesian methods require the choice of prior parameter distributions. For the mRNA and protein degradation rates, μ_m and μ_p, we use fixed values throughout, which are based on previously identified experimental measurements[2]. This facilitates our analysis of experimental time course data from the same paper in Figure 4. For the transcription and translation rates, α_m and α_p, we use prior distributions that are log-uniform. For all other parameters, we choose uniform prior distributions. Our chosen parameter ranges are informed by existing literature values[2;8;19;22;83;84],

and summarised in Table S1.

Parameter	Range of Values
Repression threshold, P_0	0 - 120,000[22]
Hill Coefficient, h	2 - 6[8;22]
Protein degradation rate, μ_p	log(2)/(90 min)[2]
mRNA degradation rate, μ_m	log(2)/(30 min)[2]
Basal transcription rate, α_m	0.01/min - 60/min[83;84]
Translation rate, α_p	1/min - 40/min[19]
Transcriptional delay, τ	5 min - 40 min[8;22]

Table S1: The values each of the parameters of the model can take, in relation to the HES5 system, informed by experimental work and biophysical limitations. These values define the prior distributions which we use in our parameter inference algorithms.

The choice of log-uniform priors for the transcription and translation rates, α_m and α_p, is motivated by the fact that possible values for these parameters span multiple orders of magnitude, from 0.01 – 120/min and 0.01 – 40/min respectively. To enable sampling from these log-uniform priors, we convert the parameters into logarithmic space using

$$\tilde{\theta} = \ln(\theta),$$

where θ represents the parameter that is being transformed into logarithmic space, i.e. α_m or α_p.

Let $P_1(\theta \mid \mathbf{y})$ and $P_2(\tilde{\theta} \mid \mathbf{y})$ define the posterior distributions for the original and transformed parameter respectively. A transformation of variables changes the shape of the posterior distribution according to

$$P_2(\tilde{\theta} \mid \mathbf{y})\mathrm{d}\tilde{\theta} = P_1(\theta \mid \mathbf{y})\mathrm{d}\theta$$

which in turn gives

$$\begin{aligned} P_2(\tilde{\theta} \mid \mathbf{y}) &= \theta P_1(\theta \mid \mathbf{y}) \\ &= e^{\tilde{\theta}} P_1(e^{\tilde{\theta}} \mid \mathbf{y}), \end{aligned} \tag{32}$$

since $\dfrac{\mathrm{d}\tilde{\theta}}{\mathrm{d}\theta} = \dfrac{1}{\theta}$.

Using $P_1(\theta \mid \mathbf{y}) = L(\mathbf{y} \mid \theta)\pi_1(\theta)$ and considering a prior that is uniform in logarithmic space,

i.e. $\pi_1(\theta) = C/\theta$ with an appropriately chosen constant C, eq. (32) leads to

$$\begin{aligned}\ln(P_2(\tilde{\theta} \mid \mathbf{y})) &= \tilde{\theta} + \ln(L(\mathbf{y} \mid e^{\tilde{\theta}})) + \ln\left(\frac{C}{\theta}\right) \\ &= \ln(L(\mathbf{y} \mid e^{\tilde{\theta}})) + \ln(C) \\ &= \ln(L(\mathbf{y} \mid \theta)) + \ln(C),\end{aligned} \tag{33}$$

Lastly, taking the derivative gives

$$\frac{\partial}{\partial \tilde{\theta}} \ln(P_2(\tilde{\theta} \mid \mathbf{y})) = \theta \frac{\partial}{\partial \theta} \ln(L(\mathbf{y} \mid \theta)). \tag{34}$$

S.7 MCMC success is qualified by multiple convergence diagnostics

When using MCMC samplers such as MH and MALA, it is necessary to check for convergence of the sampled distribution and to identify whether a sufficient number of samples has been generated. A combination of convergence diagnostics give confidence that the obtained samples are useful and represent the posterior accurately. Specifically, we use the split-$\hat{R}$ and effective sample size (ESS) methods[85;86]. We follow the guidance of Vehtari et. al.[86], running multiple chains ($m > 4$), and conclude convergence when split-$\hat{R} < 1.01$ and the total ESS is at least $50 \times 2 \times m$.

When testing our algorithm on *in silico* data, we use test diagnostics to determine inference uncertainty and accuracy. Specifically, for a given set of *in silico* data, let μ_{θ_i} define the true value of each parameter that generated the data set, and let $\hat{\mu}_{\theta_i}, \hat{\sigma}_{\theta_i}$ define the posterior mean and standard deviation, respectively, for each parameter. Throughout, we discuss the relative uncertainty (RU)

$$\mathrm{RU}_{\boldsymbol{\theta}} = \sum_{\theta_i} \frac{\hat{\sigma}_i}{|\mathrm{supp}\{\pi(\theta_i)\}|}, \tag{35}$$

where $|\mathrm{supp}\{\pi(\theta_i)\}|$ is the size of the support of the marginal posterior of parameter θ_i. The relative uncertainty hence measures how the marginal standard deviations of the posterior distribution

compares to their respective prior widths. Similarly, we introduce the mean error (ME) as

$$\mathrm{ME}_{\boldsymbol{\theta}} = \sum_{\theta_i} \frac{|\mu_{\theta_i} - \hat{\mu}_{\theta_i}|}{|\mathrm{supp}\{\pi(\theta_i)\}|}, \tag{36}$$

which measures how strongly the inferred posterior mean differs from the ground truth. For any parameters that are sampled in logarithmic space (supplementary Section S.6 above), we consider the samples and prior distributions in the original, *non*-transformed, space.

S.8 Oscillation quality is quantified by the coherence measure

In the presence of noise, time series can exhibit oscillations of varying quality. Here we introduce a measure to quantify the oscillatory properties of a time series, the coherence. We define coherence as the area under the power spectrum, $f(\omega)$, within a 20% band around the peak frequency, divided by the total area under the curve, following previous approaches[8;87]. The power spectrum is defined by

$$f(\omega) = \langle \hat{p}(w)\hat{p}^*(w) \rangle,$$

where $\hat{p}(\omega)$ is the Fourier transform of the individual protein expression time course data, and * denotes complex conjugation. Values of the coherence around zero suggest a lack of any periodic behaviour, and values around one suggest sine-wave-like behaviour. For *in silico* data we estimate coherence by averaging the power spectra from 200 traces, which individually were simulated for 8500 minutes. The initial 1000 minutes of the simulation were discarded, and not used in the averaging, to minimise any influence from the initial conditions. The use of the power spectrum to define coherence ensures that coherence is robust to changes in the sampling interval, provided that this interval is shorter than the dominant period of oscillation.

The parameter combinations that correspond to different coherence values in Figure 5 are listed in supplementary Table S2.

Coherence value	P_0	h	α_m	α_p	τ
0.00599 (low)	88288.602	5.589	0.644	17.321	34.0
0.200 (high)	30108.800	4.950	20.006	20.540	13.0

Table S2: **Parameter combinations for specific coherence values** The first row corresponds to 'low' coherence, and the second to 'high' coherence in Figure 5. To simulate noisy measurements, each synthetic data set is generated using a measurement variance of $\Sigma_\epsilon = 10^6$.

S.9 Extension of the likelihood function to account for data on mRNA expression levels

Our method generates posterior distributions from single-cell time series data of gene expression dynamics, using the likelihood function introduced in section 2.2. It may be desirable to extend this likelihood function to account for further types of data, if these are available, in order to further constrain the model parameters. Here, we briefly describe how our likelihood function can be extended if, in addition to measurements of protein expression over time, data on the population-level distribution of mRNA copy numbers is available. Such data may for example be collected via single-molecule fluorescence *in situ* hybridisation (smFISH) experiments[66;67].

Let us assume a population-level distribution of mRNA copy numbers has been observed, and that this distribution can be approximated by a Gaussian with mean $\hat{\mu}_m$ and variance $\hat{\sigma}_m^2$. We can use these data to define an extended likelihood function as

$$\pi(\mathbf{y}, \hat{\mu}_m, \hat{\sigma}_m^2 \mid \boldsymbol{\theta}) = \pi(\mathbf{y} \mid \boldsymbol{\theta}) \times \phi(\hat{m}; \hat{\mu}_m, \hat{\sigma}_m^2), \tag{37}$$

where $\hat{m}$ is the time-average of the mean inferred mRNA copy number from the Kalman filter (i.e. the first entry of ρ_t in section 2.2 averaged over all discretisation time points), $\pi(\mathbf{y} \mid \boldsymbol{\theta})$ is defined in eq. (6), and $\phi(\cdot)$ is a Normal density function, defined in eq. (7).

To generate data from *in silico* smFISH experiments, we generate the distribution of observed mRNA copy numbers numerically by simulating data from the same parameter combination that is used to generate a given *in silico* protein expression time series. Specifically, we simulate 5000 traces of time course data with a duration of 6000 minutes each, using equations (2) and (3). The first 1000 minutes of each time series are discarded to avoid any influence of the initial and past conditions.

The mean and standard deviation is then calculated from all remaining simulated mRNA values, considering all simulated time points in all time series.

S.10 Data collection and analysis

S.10.1 Imaging of Primary NS cells

To generate data in Figure 1A, primary NS cells were isolated from the dissected cortex of E12.5 Venus::HES5 embryos[1] and cultured as previously described[88]. 200,000 cells up to passage 15 were plated on laminin (Sigma, UK) coated 35mm glass-bottom dishes (Greiner-Bio One) and imaged in NS proliferation media at 37°C in 5% CO2 using a Plan Fluor 40x 1.3NA oil objective on an Nikon A1-R inverted confocal microscope. A 25μm z-range was used to ensure cells maintained in focus. Maximum projections in z direction were generated before manual cell tracking using Imaris spot function (Bitplane).

S.10.2 Conversion of Venus::HES5 intensity to molecule number

Here, we briefly describe how we translate time series of fluorescent intensity from Manning et al. (2019) into time series of molecule numbers. Throughout our analysis, we use time series data of fluorescent intensity that were published by Manning et al. (2019) and which have been corrected for (i) photobleaching and (ii) for weaker intensity signals at deeper z-positions of the imaged nuclei. A quantile-quantile plot was generated between the distribution of tracked Venus::HES5 intensities and the distribution of nuclear Venus::Hes5 concentrations across the tissue. The distribution of tracked Venus::Hes5 intensities was generated from cells in tissues with heterozygous Venus::Hes5 reporter. All recorded single cells at all time points were used. The distribution of nuclear Venus::HES5 concentrations was generated using fluorescent correlation spectroscopy (FCS) on cells in tissue slices from the same domain of the E10.5 spinal cord and with a homozygous Venus::Hes5 reporter. Data from from multiple tissue slices and multiple experiments was used. The required FCS data was also published by Manning et al. (2019). Linear regression on the quantile-quantile plot was used to generate a calibration curve between Venus::HES5 intensity and Venus::HES5 concentration over the middle 90% of the range. The gradient of the line was used as a scaling factor and applied to the

intensity values in the Venus::HES5 expression time-series to transform intensity to concentrations. Finally, the average nuclear volume from Manning et al. (2019) was taken into account to convert concentrations into nuclear molecule numbers.

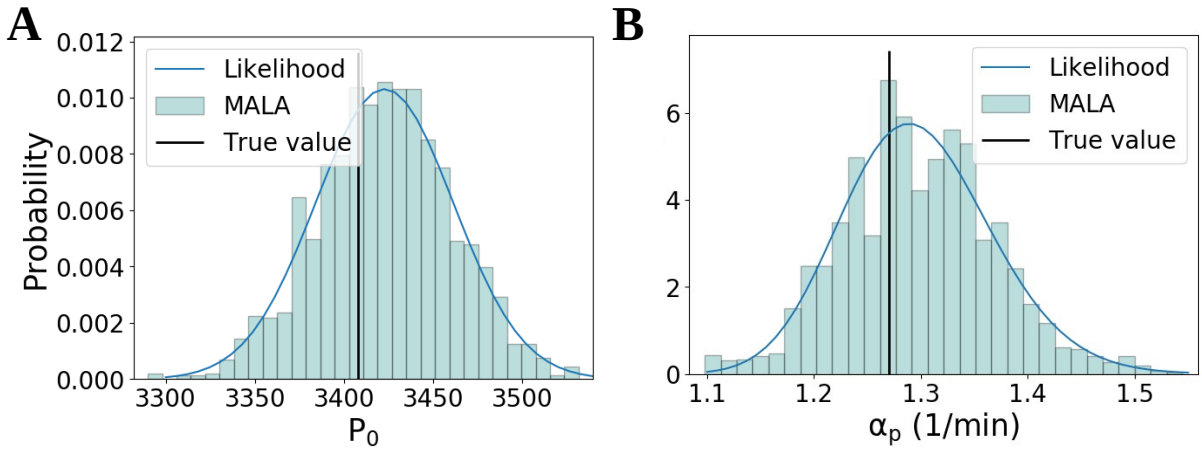


Figure S1: **Our algorithm accurately samples posterior distributions. A, B.** Posterior distributions for one-dimensional inference. For individual model parameters, posterior distributions were inferred while keeping all other parameters fixed, respectively. Shown above are the inferred marginal posteriors for the repression threshold (A) and translation rate (B) respectively as histograms, using MALA as the underlying sampling algorithm for 2500 samples. The blue lines are the analytical likelihood calculations. The sampled and analytical distributions coincide. Other parameters are shown in Figure 2.

Parameter	True Value	$\boldsymbol{\mu}$ (MALA)	$\boldsymbol{\mu}$ (MH)	$\boldsymbol{\sigma}$ (MALA)	$\boldsymbol{\sigma}$ (MH)
log Transcription rate, $\log(\alpha_m)$	2.764	2.784	2.782	0.054	0.052
log Translation rate, $\log(\alpha_p)$	0.239	0.259	0.255	0.055	0.053
Transcriptional delay, τ	30.0	30.546	30.652	6.653	6.578

Table S3: The true values for the parameters which were used to generate the data in Figure 3A, alongside the means, $\boldsymbol{\mu}$, and standard deviations, $\boldsymbol{\sigma}$ of the corresponding one-dimensional posterior distributions, from both the MALA and MH algorithms (Figure S2).

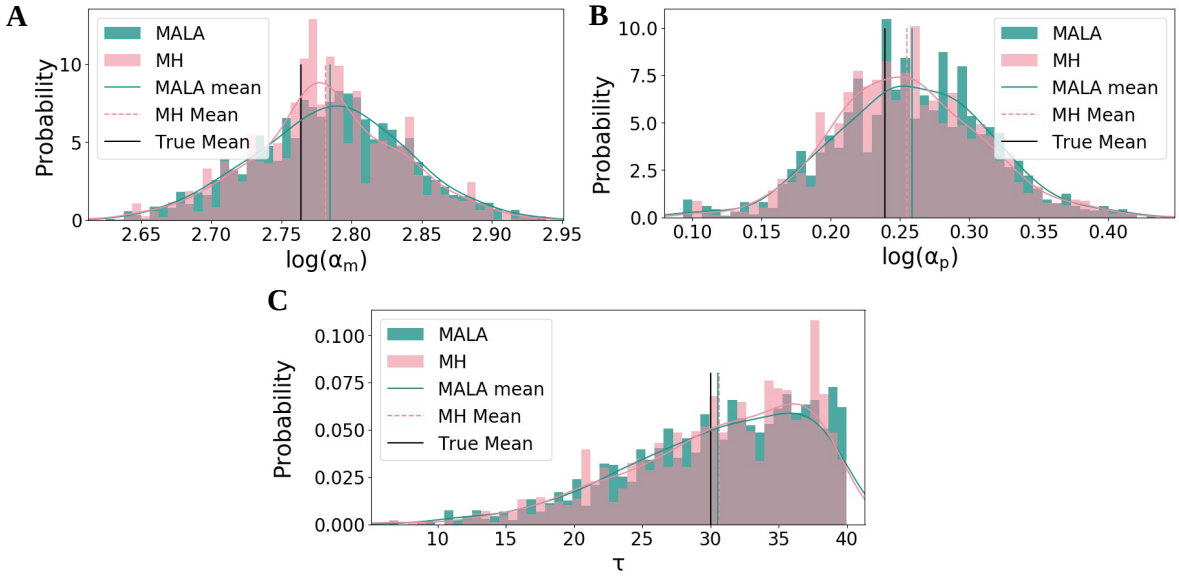


Figure S2: **Our algorithm accurately samples posterior distributions. A, B, C.** Histograms for both MALA and MH on the 1-dimensional problem for the transcription rate (A), translation rate (B) and transcriptional delay (C). Histograms and kernel density estimates are plotted, alongside the mean from each chain and the ground truth value. Other parameters are shown in Figure 2.

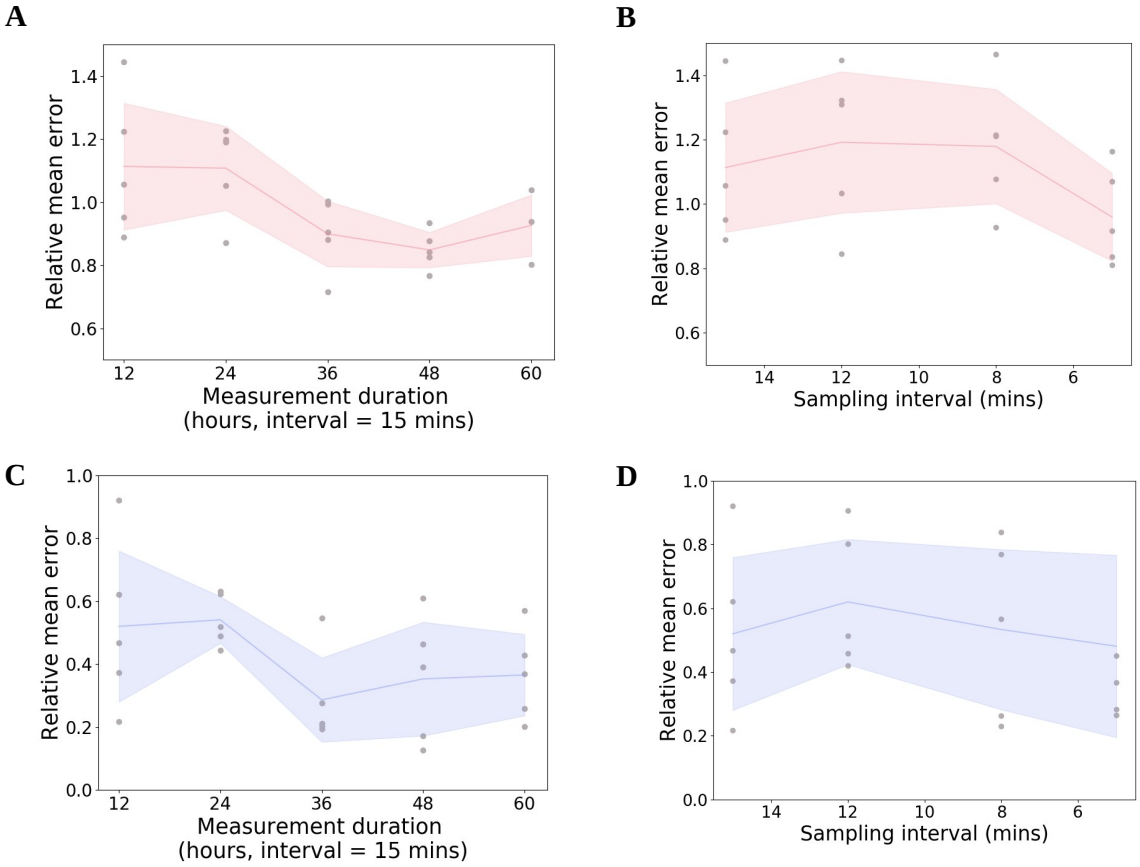


Figure S3: **Increasing the length of time course data improves inference more than increased sampling frequency.** **A.** $\mathrm{ME}_{\boldsymbol{\theta}}$ for low coherence data sets sampled with different lengths, from 12 hours to 60 hours. **B.** $\mathrm{ME}_{\boldsymbol{\theta}}$ for low coherence data sets sampled with different intervals. **C** Same as A with the high coherence data sets. **D.** Same as B with the high coherence data sets.

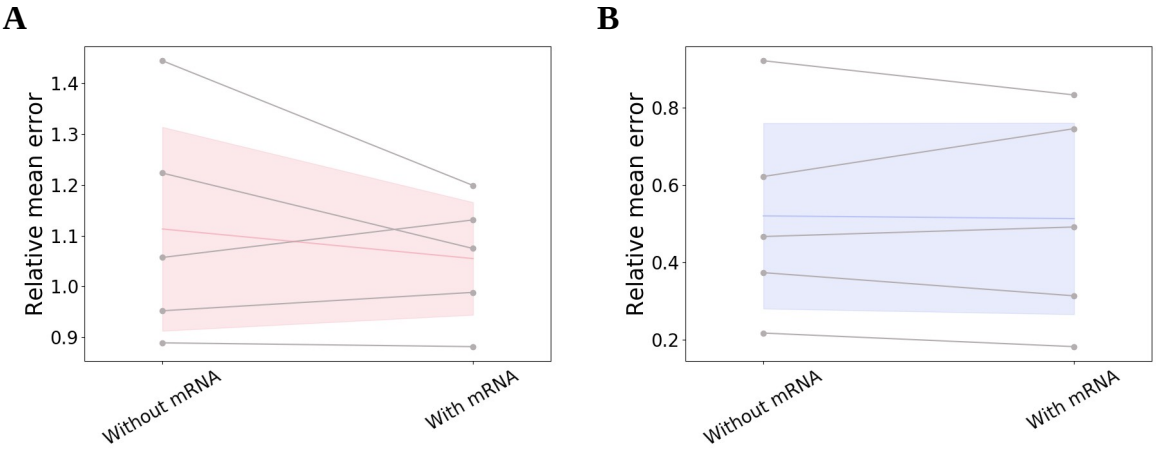
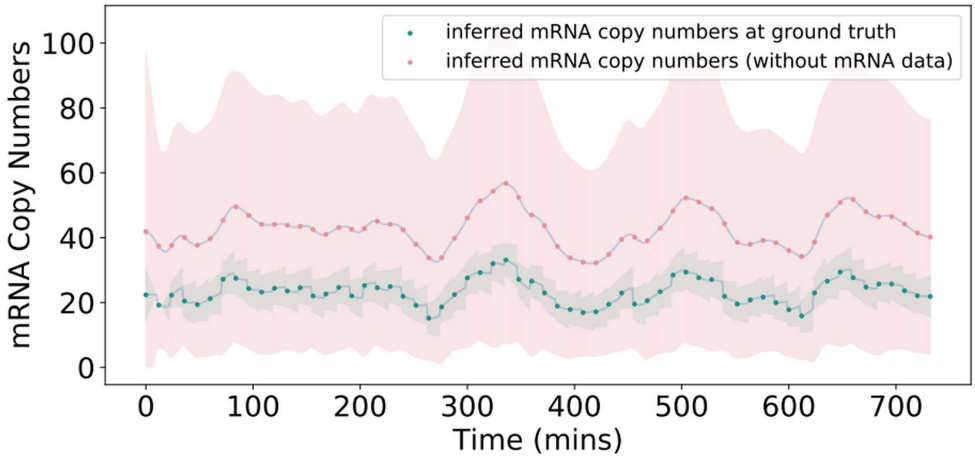


Figure S4: **Additional measurements of mRNA copy numbers do not improve overall inference accuracy. A.** Values of $\mathrm{ME}_{\boldsymbol{\theta}}$ (see equation (36)) for low coherence data sets from Figure 5 with and without additional data on mRNA copy numbers (see supplementary Section S.9 for details). **B.** $\mathrm{ME}_{\boldsymbol{\theta}}$ for high coherence data sets from Figure 5 with and without additional data on mRNA copy numbers. The coloured lines and shaded areas represent the mean and standard deviation across observed values of $\mathrm{ME}_{\boldsymbol{\theta}}$, respectively.

A


inferred mRNA copy numbers at ground truth
inferred mRNA copy numbers (without mRNA data)
mRNA Copy Numbers
0
20
40
60
80
100
0
100
200
300
400
500
600
700
Time (mins)

B

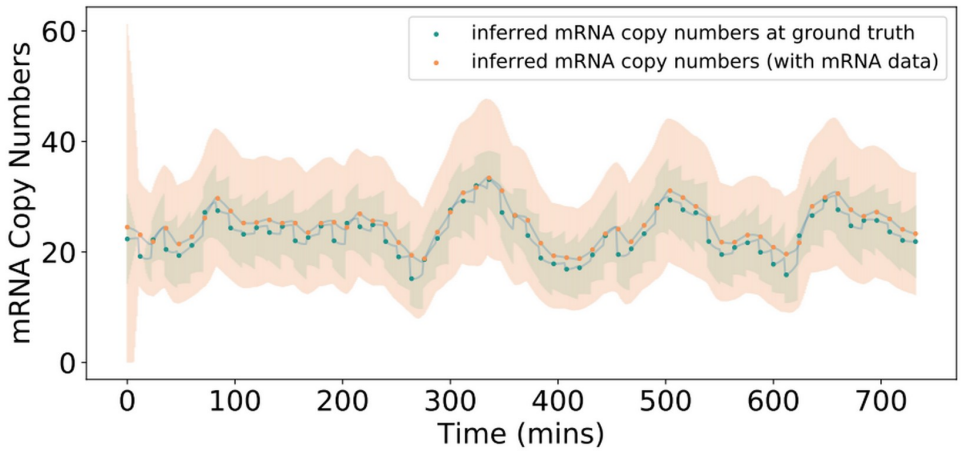

inferred mRNA copy numbers at ground truth
inferred mRNA copy numbers (with mRNA data)
mRNA Copy Numbers
0
20
40
60
0
100
200
300
400
500
600
700
Time (mins)

Figure S5: **Inclusion of data on mRNA copy numbers reduces uncertainty on estimated mRNA time series. A.** The distribution of posterior inferred mRNA copy numbers from the Kalman filter is shown (pink line: mean, pink shaded area: 95% confidence interval). The mean was calculated using mean inferred mRNA copy number across 1000 randomly chosen posterior samples at each time point, and the variance was calculated from the same samples using the law of total variance, thus accounting for uncertainty on the model parameters and for uncertainty on mRNA copy numbers at each posterior parameter combination. Additionally, we show the inferred mRNA copy numbers if only the ground truth parameter combination is considered in the Kalman filter (green line: inferred mean, green shaded area: 95% confidence interval). The posterior mean estimates are far from the ground truth levels, with the confidence interval covering values up to three times larger than the ground truth mean. Ground truth parameters are as in the high coherence parameter set of Table S2. **B.** If we include mRNA distribution information from an *in silico* smFISH experiment, the uncertainty on the inferred mRNA copy numbers is significantly reduced (orange dots: inferred mean, orange shading: 95% confidence interval), and also centered on the ground truth estimate (green dots and shading as in A).

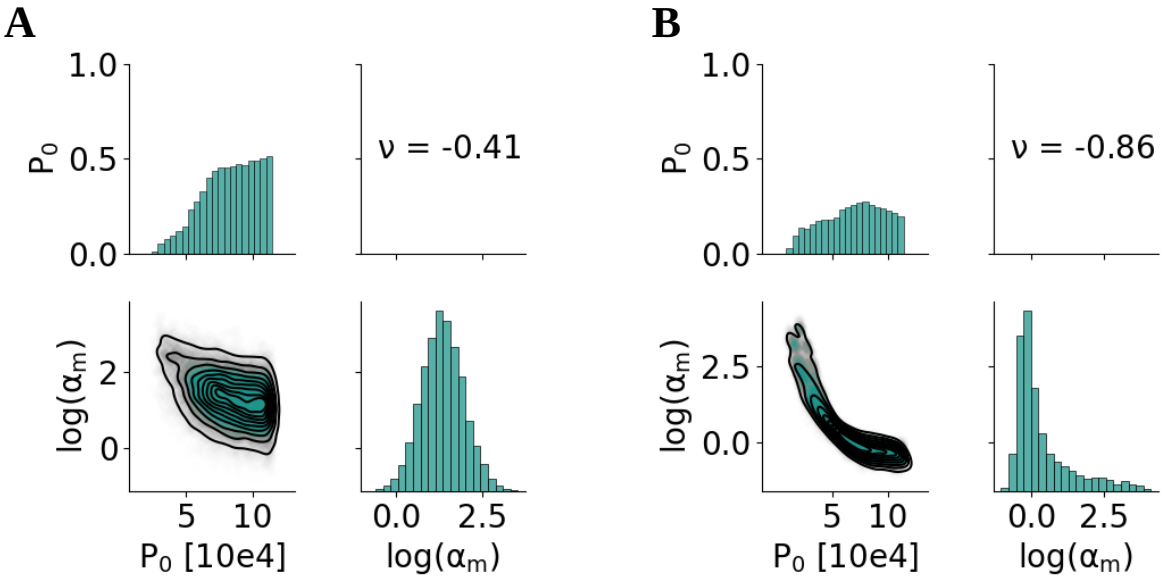


Figure S6: **Additional data on mRNA copy numbers constrains relationships between parameters. A.** Joint posterior distributions for the P_0 and $\log(\alpha_m)$ parameters, corresponding to the data set analysed in Figure 6A. The parameters are not highly correlated. **B.** Posterior distribution on the same data as in A and with mRNA distribution information included in the inference method. The relationship between the parameters is now more strongly defined. However, the support of the individual marginal posteriors remains large. This illustrates that the relative uncertainty $\mathrm{RU}_{\boldsymbol{\theta}}$ may not be reduced despite improvements in the estimation of α_T (Figure 6D).

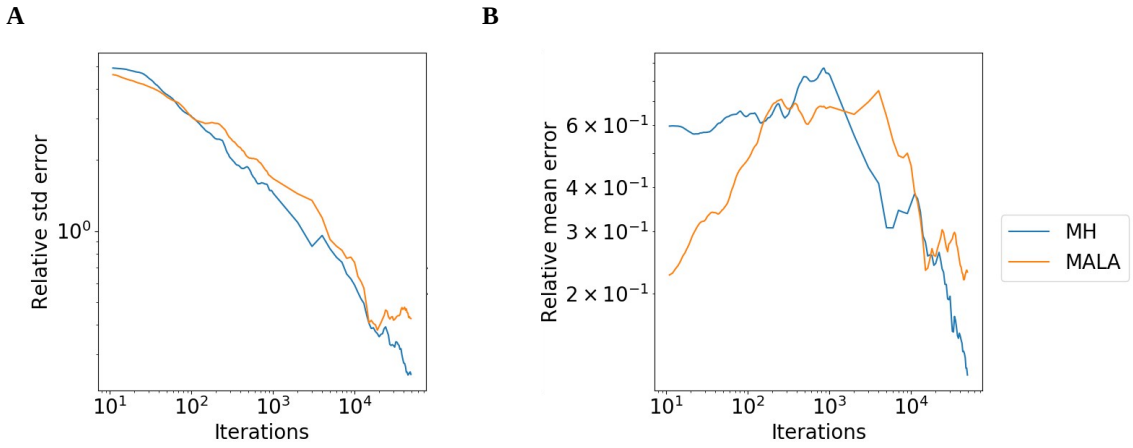


Figure S7: **MH and MALA converge to the true posterior distribution at a similar rate** Relative error in the **A** standard deviation and **B** mean of the posterior distribution as an increasing number of samples is drawn using MALA or MH, for an *in silico* data set of protein expression with a duration of 12 hours and with a sampling interval of 15 minutes, as well as assuming five unknown parameters. Note, that the data in Figures 3, 4 and 5 have the same structure. The *in silico* data set was generated using the parameter combination $P_0 = 47515, h = 4.77, \mu_m = \log(2)/30, \mu_p = \log(2)/90, \alpha_m = 2.65, \alpha_p = 17.61, \tau = 38.0, \Sigma_\epsilon = 10^6$. To calculate the error, a ground truth posterior distribution is generated by drawing 1,600,000 samples using MH. Errors are averaged over 8 chains for each sampler. For each chain, relative differences from the ground truth are calculated on each unknown parameter and then summed, to obtain an error estimate for the standard deviation and the mean. Before errors are calculated, chains are warmed up as described in supplementary Section S.3.