JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 13, Number 3, 2006

Pp. 838–851

Bayesian Sequential Inference for Stochastic Kinetic Biochemical Network Models

ANDREW GOLIGHTLY and DARREN J. WILKINSON

ABSTRACT

As postgenomic biology becomes more predictive, the ability to infer rate parameters of genetic and biochemical networks will become increasingly important. In this paper, we explore the Bayesian estimation of stochastic kinetic rate constants governing dynamic models of intracellular processes. The underlying model is replaced by a diffusion approximation where a noise term represents intrinsic stochastic behavior and the model is identified using discrete-time (and often incomplete) data that is subject to measurement error. Sequential MCMC methods are then used to sample the model parameters on-line in several data-poor contexts. The methodology is illustrated by applying it to the estimation of parameters in a simple prokaryotic auto-regulatory gene network.

Key words: Bayesian inference, particle filter, missing data, nonlinear diffusion, stochastic differential equation.

1. INTRODUCTION

Traditionally, the time evolution of a biochemical network is described by a set of coupled differential equations derived using the law of mass action and the concentrations of each species. This widely used approach, however, assumes that the system is both continuous and deterministic. In reality, chemical reactions are intrinsically stochastic and occur as discrete events resulting from random molecular collisions (Gillespie, 1977). Although relatively little work has addressed the stochasticity of biochemical networks (Arkin *et al.*, 1998; McAdams and Arkin, 1999), it is clear that many important intracellular processes, such as signal transduction and gene expression, can be effectively described only by stochastic processes. Stochastic effects at this level can have large significance even on high-level outcomes, such as an organism's aging (Finch and Kirkwood, 2000).

In order to perform analysis on a stochastic biochemical network model, it is essential that each network parameter is obtained (Kitano, 2001). The resulting problem is known as reverse engineering (Bower and Bolouri, 2000) and presents the challenge of how to estimate key rate parameters given observed time course data. Although inference for "exact" stochastic kinetic models is possible, it is computationally problematic for models of realistic size and complexity (Boys *et al.*, 2004). We therefore work with the diffusion approximation which, though often inadequate for simulation, can be satisfactory for inferential purposes (Golightly and Wilkinson, 2005a).

Typically, since biochemical data arrive at discrete times, yet the model is formulated in continuous time, it is natural to work with the first-order Euler discretization. As interobservation times are usually too large

School of Mathematics and Statistics, University of Newcastle upon Tyne, NE1 7RU, UK.

to be used as a time step in the Euler scheme, we follow Pedersen (1995) and augment the observed low frequency data by introducing $m-1$ latent data points in between each pair of observations. Markov chain Monte Carlo (MCMC) methods, which sample the posterior distribution of model parameters, have been proposed independently by Jones (1997), Elerian *et al.* (2001), and Eraker (2001). Unfortunately, inference can be complicated if the amount of augmentation is large. It is well known that high dependence between the parameters and missing data results in slow rates of convergence of basic algorithms such as Gibbs samplers (Roberts and Stramer, 2001).

In this paper, we utilize recently developed simulation-based sequential algorithms (Golightly and Wilkinson, 2005b) to conduct inference for a partially and discretely observed stochastic kinetic model. Our proposed simulation filter does not break down as either the degree of augmentation or the number of observations increases and can be implemented for multiple, partially observed datasets.

The structure of this paper is organized as follows. In Section 2, methods for modeling stochastic kinetics are described, Section 2.1 describes the continuous time Markov process model, and Section 2.2 gives the diffusion approximation. Inference for nonlinear, partially observed diffusion models is outlined in Section 3 before an application is presented in Section 4. Conclusions are drawn in Section 5.

2. STOCHASTIC KINETICS

2.1. Continuous time Markov process model

We typically consider a system of reactions involving k species $Y_1, Y_2, \ldots, Y_k$ and r reactions R_1, $R_2, \ldots, R_r$ in thermal equilibrium inside some fixed volume. The system will take the form

$$\begin{array}{lllll} R_1\colon & u_{11}Y_1 + u_{12}Y_2 + \cdots + u_{1k}Y_k & \longrightarrow & q_{11}Y_1 + q_{12}Y_2 + \cdots + q_{1k}Y_k \\ R_2\colon & u_{21}Y_1 + u_{22}Y_2 + \cdots + u_{2k}Y_k & \longrightarrow & q_{21}Y_1 + q_{22}Y_2 + \cdots + q_{2k}Y_k \\ \vdots & \vdots & \vdots & \vdots \\ R_r\colon & u_{r1}Y_1 + u_{r2}Y_2 + \cdots + u_{rk}Y_k & \longrightarrow & q_{r1}Y_1 + q_{r2}Y_2 + \cdots + q_{rk}Y_k \end{array} \tag{1}$$

where u_{ij} is the stoichiometry associated with the j^{th} reactant of the i^{th} reaction and q_{ij} is the stoichiometry associated with the j^{th} product of the i^{th} reaction. Each reaction, R_i, has a stochastic rate constant, c_i, and a rate law or hazard, $h_i(Y, c_i)$, where $Y = (Y_1, Y_2, \ldots, Y_k)'$ is the current state of the system and each hazard is determined by the order of reaction R_i under an assumption of mass action kinetics. Note that for transparency, we denote by Y_i both the species and the number of molecules it represents in the system. We may represent (1) somewhat more compactly as

$$\mathrm{U}Y \longrightarrow \mathrm{Q}Y,$$

where $U = (u_{ij})$ and $Q = (q_{ij})$ are $r \times k$ dimensional matrices (obtained from the stoichiometry of the system). As the net effect of reaction i is a change of $a_{ij} = q_{ij} - u_{ij}$, the reaction network can be represented by the (net effect reaction) matrix $\mathrm{A} = \mathrm{Q} - \mathrm{U}$.

Stochastic models of cellular processes are reasonably well developed and are commonly based on techniques for solving the "chemical master equation." The main element of the master equation is the function $P(Y_1, Y_2, \ldots, Y_k; t)$, which gives the probability that there will be at time t, $Y_1, Y_2, \ldots, Y_k$ molecules of each respective species. We write this function as the sum of the probabilities of the number of ways in which the network can arrive in state $Y = (Y_1, Y_2, \ldots, Y_k)'$ at time $t + \Delta t$ to give

$$\begin{aligned} P(Y; t+\Delta t) = & \sum_{i=1}^{r} h_i(Y - \mathrm{A}_i, c_i) P(Y - \mathrm{A}_i; t)\Delta t \\ & + \left\{ 1 - \sum_{i=1}^{r} h_i(Y, c_i)\Delta t \right\} P(Y; t) + o(\Delta t) \end{aligned} \tag{2}$$

where A_i denotes the i^{th} row of the net effect matrix A. Rearrangement of (2) and taking $\Delta t \to 0$ leads to the master equation,

$$\frac{\partial}{\partial t}P(Y;t) = \sum_{i=1}^{r}\{h_i(Y - A_i, c_i)P(Y - A_i; t) - h_i(Y, c_i)P(Y;t)\}, \tag{3}$$

further details of which have been given by van Kampen (2001) and Doraiswamy and Kulkarni (1987) among others. Although the master equation is exact, it is only tractable for a handful of cases, and those exactly solvable cases have been summarized by McQuarrie (1967). Therefore, stochastic models are typically examined using a discrete event simulation algorithm known in the physical sciences as the "Gillespie algorithm" (Gillespie, 1977). Although using the latter algorithm is straightforward for simulation, inference for "exact" stochastic-kinetic models is computationally problematic for models of realistic size and complexity (Boys *et al.*, 2004). We therefore use a continuous approximation of (3)—the diffusion approximation.

2.2. *The diffusion approximation*

By assuming that the jumps of the Markov process governed by (3) are "small" and that the solution, $P(Y;t)$, varies slowly with Y, we can expand the first term in (3) by means of a second-order Taylor expansion to give the Fokker–Planck equation (van Kampen, 2001). Formally, for a k dimensional process $Y(t)$ with components $Y_1(t), \ldots, Y_k(t)$, the nonlinear Fokker–Planck equation is given by

$$\frac{\partial}{\partial t}P(Y;t) = -\sum_{i=1}^{k}\frac{\partial}{\partial Y_i}\{\mu_i(Y)P(Y;t)\} + \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{k}\frac{\partial^2}{\partial Y_i \partial Y_j}\{\beta_{ij}(Y)P(Y;t)\}, \tag{4}$$

where we define the infinitesimal means for $i = 1, \ldots, k$ by

$$\mu_i(Y) = \lim_{\Delta t \to 0}\frac{1}{\Delta t}\mathrm{E}[\{Y_i(t + \Delta t) - Y_i(t)\}|Y(t) = Y] \tag{5}$$

and the infinitesimal second moments for $i, j = 1, \ldots, k$ by

$$\beta_{ij}(Y) = \lim_{\Delta t \to 0}\frac{1}{\Delta t}\mathrm{Cov}[\{Y_i(t + \Delta t) - Y_i(t)\}, \{Y_j(t + \Delta t) - Y_j(t)\}|Y(t) = Y]. \tag{6}$$

Now suppose at time t, the state of the system is $Y(t) = (Y_1(t), \ldots, Y_k(t))' = Y$ so that the hazards of $R_1, R_2, \ldots, R_r$ are $h_1(Y, c_1), h_2(Y, c_2), \ldots, h_r(Y, c_r)$. Let N_i denote the number of type i reactions occurring in the interval $(t, t + \Delta t]$. Then for "small" time Δt, $N_i \approx \text{Poisson}(h_i(Y, c_i)\Delta t)$ (due to the assunption of constant reaction hazard), and the change in the number of molecules of Y_j is given by

$$Y_j(t + \Delta t) - Y_j(t) = a_{1j}N_1 + a_{2j}N_2 + \cdots + a_{rj}N_r. \tag{7}$$

For each increment $Y_j(t + \Delta t) - Y_j(t)$, $j = 1, \ldots, k$ given by (7), we calculate the infinitesimal means and variances through straightforward application of (5) and (6). It can be shown under certain conditions (see Kloeden and Platen [1992]) that the solution of (4) satisfies an Itô stochastic differential equation (SDE),

$$dY(t) = \mu(Y, \Theta)\,dt + \beta^{\frac{1}{2}}(Y, \Theta)\,dW(t) \tag{8}$$

where $\mu(Y, \Theta)$ is the column vector of $\mu_i(Y)$ (known as drift), $\beta^{\frac{1}{2}}(Y, \Theta)$ is any matrix satisfying $\beta^{\frac{1}{2}}(\beta^{\frac{1}{2}})' = [\beta_{ij}(Y)] = \beta(Y)$ (known as the diffusion matrix), and we let both functions depend explicity on the parameter vector $\Theta = (c_1, c_2, \ldots, c_r)'$. Finally, $dW(t) = (dW_1(t), \ldots, dW_k(t))'$ is the increment of (standard, k dimensional) Brownian motion. For a reaction network with net effect matrix A, we may compute

$$\mu(Y, \Theta) = \mathrm{A}'h(Y, \Theta), \beta(Y, \Theta) = \mathrm{A}'\mathrm{diag}\{h(Y, \Theta)\}\mathrm{A} \tag{9}$$

where $h(Y, \Theta)$ is the column vector of hazards $h_i(Y, c_i)$. Further details of the diffusion approximation can be found in Allen (2002).

2.3. Example: Prokaryotic auto-regulatory gene network

Transcriptional regulation has been studied extensively in both prokaryotic and eukaryotic organisms (see, for example, McAdams and Arkin [1999], Latchman [2002], and Ng *et al.* [2004]). In a simple model of prokaryotic auto regulation, a protein (l) coded for by a gene (i) represses its own transcription and also the transcription of another gene, (g), by binding to a regulatory region upstream of the gene. The transcription of a gene into mRNA is facilitated by an enzyme, RNA-polymerase, and this process begins with the binding of this enzyme to a site on the gene called a promoter. After the initial binding, RNA-polymerase travels away from the promoter along the gene, synthesizing mRNA as it moves. In our model, transcription is repressed by a repressor protein, l, which can bind to sites on the DNA known as operators. We simplify the repression mechanisms with the following reactions.

$$\begin{array}{lrcl} R_1\colon & \mathsf{l}+\mathsf{i} & \longrightarrow & \mathsf{l}\cdot\mathsf{i} \\ R_2\colon & \mathsf{l}\cdot\mathsf{i} & \longrightarrow & \mathsf{l}+\mathsf{i} \\ R_3\colon & \mathsf{l}+\mathsf{g} & \longrightarrow & \mathsf{l}\cdot\mathsf{g} \\ R_4\colon & \mathsf{l}\cdot\mathsf{g} & \longrightarrow & \mathsf{l}+\mathsf{g} \end{array} \tag{10}$$

We represent the transcription of i, the binding of a ribosome to mRNA, the translation of mRNA, and the folding of the resulting polypetide chain into a folding protein, l, by

$$R_5\colon \quad \mathsf{i} \quad \longrightarrow \quad \mathsf{i}+\mathsf{r_i},\ R_6\colon \quad \mathsf{r_i} \quad \longrightarrow \quad \mathsf{r_i}+\mathsf{l}. \tag{11}$$

Similarly, we represent the transcription of g and translation mechanism by

$$R_7\colon \quad \mathsf{g} \quad \longrightarrow \quad \mathsf{g}+\mathsf{r_g},\ R_8\colon \quad \mathsf{r_g} \quad \longrightarrow \quad \mathsf{r_g}+\mathsf{G}. \tag{12}$$

Finally, the model is completed by mRNA degradation,

$$R_9\colon \quad \mathsf{r_i} \quad \longrightarrow \quad \emptyset,\ R_{10}\colon \quad \mathsf{r_g} \quad \longrightarrow \quad \emptyset \tag{13}$$

and protein degradation,

$$R_{11}\colon \quad \mathsf{l} \quad \longrightarrow \quad \emptyset,\ R_{12}\colon \quad \mathsf{G} \quad \longrightarrow \quad \emptyset. \tag{14}$$

Although the model offers a simplistic view of the mechanisms involved in gene regulation, it will provide insight into how inference might be done in more complex networks. For a detailed discussion of gene regulation, see Ptashne (1992) and Latchman (2002).

We now turn our attention to calculating the diffusion approximation for the model given by (10)–(14). We order the species by setting $Y = (\mathsf{l}, \mathsf{G}, \mathsf{l}\cdot\mathsf{i}, \mathsf{l}\cdot\mathsf{g}, \mathsf{i}, \mathsf{g}, \mathsf{r_i}, \mathsf{r_g})'$ and use the stoichiometry of the system to obtain the net effect matrix,

$$\mathrm{A}' = \begin{pmatrix} -1 & 1 & -1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}. \tag{15}$$

Now assume for reaction i a stochastic rate constant of c_i, and consider the time evolution of the system as a Markov process with state $Y(t) = Y$ at time t. Reactions R_1 and R_3 are second order and their hazards can be computed (using the law of mass action) as $h_1(Y, \Theta) = c_1\mathsf{li}$ and $h_3(Y, \Theta) = c_3\mathsf{lg}$. As the remaining reactions are first order, their hazards are straight forward to compute; for example, $h_{12}(Y, \Theta) = c_{12}\mathsf{G}$.

Before calculation of $\mu(Y, \Theta)$ and $\beta(Y, \Theta)$, it should be noted that the net effect matrix A is not of full rank (as the number of molecules of $\mathsf{l}\cdot\mathsf{i}$ and $\mathsf{l}\cdot\mathsf{g}$ are related to the number of molecules of i and g, respectively). Inspection of (15) reveals that adding row 3 of A' to row 5 implies

$$\mathsf{l}\cdot\mathsf{i} + \mathsf{i} = K_1 \tag{16}$$

and similarly, adding row 4 to row 6 yields

$$\mathsf{I}\cdot\mathsf{g} + \mathsf{g} = K_2 \tag{17}$$

where K_1 and K_2 are conservation constants. As this rank degeneracy will cause problems for the inference method considered in Section 3, we remove rows 3 and 4 from A' to obtain A of full rank. We then use (16) and (17) to substitute $K_1 - \mathsf{i}$ and $K_2 - \mathsf{g}$ for $\mathsf{I}\cdot\mathsf{i}$ and $\mathsf{I}\cdot\mathsf{g}$ respectively to reduce our model to one involving just six chemical species, $Y = (\mathsf{I}, \mathsf{G}, \mathsf{i}, \mathsf{g}, \mathsf{r_i}, \mathsf{r_g})'$. The full diffusion approximation can then be computed using (9), for example,

$$\mu(Y, \Theta) = \begin{pmatrix} c_2(K_1 - \mathsf{i}) + c_4(K_2 - \mathsf{g}) + c_6\mathsf{r_i} - c_1\mathsf{Ii} - c_3\mathsf{Ig} - c_{11}\mathsf{I} \\ c_8\mathsf{r_g} - c_{12}\mathsf{G} \\ c_2(K_1 - \mathsf{i}) - c_1\mathsf{Ii} \\ c_4(K_2 - \mathsf{g}) - c_3\mathsf{Ig} \\ c_5\mathsf{i} - c_9\mathsf{r_i} \\ c_7\mathsf{g} - c_{10}\mathsf{r_g} \end{pmatrix}.$$

Note that our parameter vector Θ consists of all stochastic rate constants and is given by $\Theta = (c_1, c_2, \ldots, c_{12})'$. For a further discussion of how to calculate the diffusion approximation for a given reaction network, see Golightly and Wilkinson (2005a).

3. INFERENCE FOR NONLINEAR DIFFUSION MODELS

3.1. Models

We consider inference for a d-dimensional Itô diffusion that satisfies a stochastic differential equation of the form given by (8) and assume that the conditions under which the SDE can be solved for $Y(t)$ are satisfied (Øksendal, 1995).

Often, $Y(t)$ will consist of both observable and unobservable components. To deal with this, we define $Y(t) = (X(t), Z(t))'$, where $X(t)$ defines the observable part and $Z(t)$ the unobservable part of the system. Note that $X(t)$ and $Z(t)$ have dimensions d_1 and d_2 respectively such that $Y(t)$ has dimension $d = d_1 + d_2$. We assume further that the process $X(t)$ is subject to measurement error such that we actually observe

$$V(t) = X(t) + \epsilon(t), \tag{18}$$

where $\epsilon(t) \sim \mathrm{N}(0, \Sigma)$ and $\Sigma = \mathrm{diag}\{\sigma_i^2\}$ for $i = 1, \ldots, d_1$. Note that for unknown Σ, we have $\Theta = (\theta_1, \ldots, \theta_p, \sigma_1, \ldots, \sigma_{d_1})'$. The process $V(t)$ will be observed at a finite number of times and the objective is to conduct inference for the (unknown) parameter vector Θ on the basis of these noisy, partial, and discrete observations.

In practice, it is necessary to work with the discretized version of (8), given by the Euler approximation,

$$\Delta Y(t) = \mu(Y(t), \Theta)\Delta t + \beta^{\frac{1}{2}}(Y(t), \Theta)\Delta W(t), \tag{19}$$

where $\Delta W(t)$ is a d dimensional iid $\mathrm{N}(0, I\Delta t)$ random vector. Now suppose we have measurements $v(\tau_i)$ at evenly spaced times $\tau_0, \tau_1, \ldots, \tau_T$ with intervals of length $\Delta^* = \tau_{i+1} - \tau_i$. As Δ^* is often too large to be used as a time step in (19), we put $\Delta t = \Delta^*/m$ for some positive integer $m > 1$. By choosing m to be sufficiently large, we can ensure that the discretization bias is arbitrarily small, but this also introduces the problem of $m - 1$ missing values in between every pair of observations.

We deal with these missing values by dividing the entire time interval $[\tau_0, \tau_T]$ into $mT + 1$ equidistant points $\tau_0 = t_0 < t_1 < \cdots < t_n = \tau_T$ such that $V(t)$ is observed at times $t_0, t_m, \ldots, t_n$. Altogether we have $d(nm + 1)$ missing values which we substitute with simulations $Y(t_i)$. We refer to the collection of simulated data as the augmented data. Eraker (2001) denotes by $\hat{\mathrm{Y}}$ the $d \times (n + 1)$ matrix obtained by

stacking all elements of the augmented data, that is,

$$\hat{\Upsilon} = \begin{pmatrix} X_1(t_0) & X_1(t_1) & \cdots & X_1(t_m) & X_1(t_{m+1}) & \cdots & X_1(t_n) \\ X_2(t_0) & X_2(t_1) & \cdots & X_2(t_m) & X_2(t_{m+1}) & \cdots & X_2(t_n) \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ X_{d_1}(t_0) & X_{d_1}(t_1) & \cdots & X_{d_1}(t_m) & X_{d_1}(t_{m+1}) & \cdots & X_{d_1}(t_n) \\ Z_1(t_0) & Z_1(t_1) & \cdots & Z_1(t_m) & Z_1(t_{m+1}) & \cdots & Z_1(t_n) \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ Z_{d_2}(t_0) & Z_{d_2}(t_1) & \cdots & Z_{d_2}(t_m) & Z_{d_2}(t_{m+1}) & \cdots & Z_{d_2}(t_n) \end{pmatrix}.$$

We now let $Y^i = (X^i, hzi)$ denote the i^{th} column of $\hat{\Upsilon}$. By adopting a fully Bayesian approach, we summarize our a priori beliefs about Θ and Y^0 via the prior distributions $\pi(\Theta)$ and $\pi(Y^0)$, respectively. Then the joint posterior density for parameters and augmented data is given by

$$\pi(\hat{\Upsilon}, \Theta | v_{\text{obs}}) \propto \pi(\Theta)\pi(Y^0) \left[\prod_{i=0}^{n-1} \pi(Y^{i+1} | Y^i, \Theta) \right] \left[\prod_{i \in \{0,m,\ldots,n\}} \pi(v^i | X^i, \Theta) \right], \tag{20}$$

where v^i denotes v_{t_i}, $v_{\text{obs}} = (v^0, v^m, \ldots v^n)$,

$$\pi(Y^{i+1} | Y^i, \Theta) = \phi(Y^{i+1}; Y^i + \mu_i \Delta t, \beta_i \Delta t) \tag{21}$$

and

$$\pi(v^i | X^i, \Theta) = \phi(v^i; X^i, \Sigma). \tag{22}$$

Here, $\mu_i = \mu(Y^i, \Theta)$, $\beta_i = \beta(Y^i, \Theta)$, and $\phi(\cdot; \psi, \gamma)$ denotes the Gaussian density with mean ψ and variance matrix γ. Note that $\pi(Y^{i+1} | Y^i, \Theta)$ is the one step ahead transition density obtained from the Euler discretization.

As discussed in Tanner and Wong (1987), inference may proceed by alternating between simulation of parameters conditional on augmented data and simulation of the missing data given the observed data and the current state of the model parameters. As the joint posterior (20) is usually high dimensional, a Gibbs sampler is a particularly convenient way of sampling from it (Golightly and Wilkinson, 2005a). However, as the augmentation increases, high dependence between missing data and parameters results in arbitrarily slow rates of convergence. Although a solution to this problem is known in the case of univariate diffusions (Roberts and Stramer, 2001), it is not possible to extend this technique to the multivariate diffusions considered here (Wilkinson, 2003).

As each new observation arrives, our proposed simulation filter samples a new $(\Theta_*, \hat{\Upsilon}_*)$ in two stages: first Θ_* is sampled from a suitable proposal, and then $\hat{\Upsilon}_*$ is sampled from a tractable approximation to $(\hat{\Upsilon} | \Theta_*, v_{\text{obs}})$. By simulating the latent data to be consistent with Θ_*, the dependence between them is overcome (Golightly and Wilkinson, 2005b). For further discussions on the use of MCMC methods for the Bayesian analysis of diffusions, see Roberts and Stramer (2001), Elerian *et al.* (2001), and Eraker (2001).

3.2. Simulation filter

In the context of discrete time series with unobserved state variables, Bayesian sequential filtering has been discussed extensively, e.g., Berzuini *et al.* (1997), Pitt and Shephard (1999), and Doucet *et al.* (2000). Filtering for both parameters and state has been discussed by Liu and West (2001) among others.

We consider data $D_j = (v^0, v^m, \ldots, v^j)$, (where j is an integer multiple of m) arriving at times $t_0, t_m, \ldots, t_j$ such that at time t_{j+m}, new data v^{j+m} are accompanied by m missing columns, $Y^{j+1}, \ldots, Y^{j+m}$. As each observation becomes available, we are interested in the on-line estimation of the unknown parameter vector, Θ.

We assume that we have an equally weighted sample of size S, $\{(\Theta_{(s)}, Y^j_{(s)}), s = 1, \ldots, S\}$ (with weights $w^j_{(s)} = 1/S$), from the distribution $\pi(\Theta, Y^j | D_j)$, which we will denote by $\pi_j(\Theta, Y^j)$. At time t_{j+m},

we observe v^{j+m}, which we will refer to as v^M (putting $M = j+m$). Assimilation of the information contained in v^M consists of generating a sample, $\{(\Theta_{(s)}, Y^M_{(s)}), s = 1, \ldots, S\}$ from the posterior $\pi_M(\Theta, Y^M)$, which can be found by formulating the posterior for parameters and augmented data, then integrating out the latent data. Using (20), we have

$$\pi_M(\Theta, Y^M) \propto \int_{\hat{\Upsilon}_M} \pi(\Theta)\pi(Y^0) \prod_{i=0}^{M-1} \pi(Y^{i+1}|Y^i, \Theta) \prod_{i\in\{0,m,\ldots,M\}} \pi(v^i|X^i, \Theta) \tag{23}$$

where we define $\hat{\Upsilon}_M = (Y^0, Y^1, \ldots, Y^{M-1})$ and is simply the vector of latent values up to time t_M. Hence, our target is AU1

$$\pi_M(\Theta, Y^M) \propto \pi_j(\Theta, Y^j)\pi(v^M|X^M, \Theta) \prod_{i=j}^{M-1} \pi(Y^{i+1}|Y^i, \Theta) \tag{24}$$

with $Y^j, \ldots, Y^{M-1}$ integrated out. We sample (24) by drawing $(Y^j, Y^{j+1}, \ldots, Y^M, \Theta)$, via MCMC, then discarding all components except (Θ, Y^M).

3.3. *Filtering for parameters and state*

As $\pi_j(\Theta, Y^j)$ has no analytic form, we recursively approximate $\Theta, Y^j|D_j$ by the "particles" $\{(\Theta_{(s)}, Y^j_{(s)}), s = 1, \ldots, S\}$ with each $\Theta_{(s)}, Y^j_{(s)}$ having a discrete probablity mass of $w^j_{(s)} = 1/S$. We assume that as $S \to \infty$, the particles approximate the filtering density, $\pi_j(\Theta, Y^j)$, increasingly well. The class of filters which treat the discrete support generated by the particles as the true (filtering) density are known as particle filters. Various implementations of particle filters have been proposed in the literature including sampling/importance resampling (Doucet *et al.*, 2000) and MCMC (Pitt and Shephard, 1999). Here we focus on an MCMC approach which we refer to as the simulation filter.

In the first step of our MCMC scheme, propose (Θ_*, Y^j_*) from $\pi_j(\Theta, Y^j)$ using the kernel density estimate of $\pi_j(\cdot, \cdot)$. First select an integer, u, uniformly from the set $\{1, \ldots, S\}$, and then put

$$(\Theta_*, Y^j_*)' \sim \mathrm{N}\{(\Theta_{(u)}, Y^j_{(u)})', \omega^2 B\} \tag{25}$$

where B is the Monte Carlo posterior variance and the overall scale of the kernel is a function of the smoothing parameter, ω^2, usually around 0.02. For large datasets, however, Liu and West (2001) suggest that the random disturbances add up to give "information loss" over time (as the kernel density function is always overdispersed relative to the posterior sample by a factor $1 + \omega^2$). To correct this, Liu and West (2001) employ a kernel shrinkage method by setting

$$(\Theta_*, Y^j_*)' \sim \mathrm{N}\{a(\Theta_{(u)}, Y^j_{(u)})' + (1-a)(\bar{\Theta}, \bar{Y}^j)', \omega^2 B\} \tag{26}$$

where $a^2 = 1 - \omega^2$, $\omega^2 = 1 - ((3\delta - 1)/2\delta)^2$, δ is a discount factor usually around 0.99, and $(\bar{\Theta}, \bar{Y}^j)'$ is the Monte Carlo posterior mean of $\pi_j(\Theta, Y^j)$. For the data considered in Section 4, we found that using (25) works sufficiently well. See Liu and West (2001) and also West (1993) for further discussions on kernel smoothing.

Given $X^M_* \sim \pi(\cdot|v^M, \Theta_*)$, we are then tasked with simulating $Y^{j+1}_*, \ldots, Y^{M-1}_*, Z^M_*$ conditional on Θ_*, Y^j_* and X^M_*. However, obtaining the conditional density of missing values between two "observations" that are m steps apart, under the nonlinear structure of the diffusion process, is not trivial. We deal with this problem by adopting a "modified bridge" construct proposed by Durham and Gallant (2002). That is, treating Y^j_* and X^M_* fixed, we draw Y^{i+1}_*, for $i = j, \ldots, M-2$, from a Gaussian approximation to $\pi(Y^{i+1}_*|Y^i_*, X^M_*, \Theta_*)$ for which we denote the approximate density by $\tilde{\pi}(Y^{i+1}_*|Y^i_*, X^M_*, \Theta_*)$ (see Golightly and Wilkinson [2005b], Durham and Gallant [2002], and Elerian *et al.* [2001] for a review).

The final step in our MCMC scheme is to draw $Z_*^M \sim \pi(\cdot|Y_*^{M-1}, X_*^M, \Theta_*)$, that is, from the one step ahead Euler density further conditioned on X_*^M. Hence, if at some iteration, s, of our sampler we have current value, $\Phi_{(s)} = (Y^j, \ldots, Y^M, \Theta)$, then at iteration $s+1$ we accept a move to $\Phi_* = (Y_*^j, \ldots, Y_*^M, \Theta_*)$ with probability $\min\{1, \alpha\}$, where

$$\alpha = \frac{\pi(Z^M|Y^{M-1}, X^M, \Theta) \prod_{i=j}^{M-2} \tilde{\pi}(Y^{i+1}|Y^i, X^M, \Theta) \prod_{i=j}^{M-1} \pi(Y_*^{i+1}|Y_*^i, \Theta_*)}{\pi(Z_*^M|Y_*^{M-1}, X_*^M, \Theta_*) \prod_{i=j}^{M-2} \tilde{\pi}(Y_*^{i+1}|Y_*^i, X_*^M, \Theta_*) \prod_{i=j}^{M-1} \pi(Y^{i+1}|Y^i, \Theta)}, \tag{27}$$

and store $(\Theta_{(s+1)}, Y_{(s+1)}^M)$, ready for the next time point. The Markov chain generated in this way has the posterior distribution of interest, $\pi_M(\Theta, Y^M)$, as its invariant distribution. The simulation filter then has the following algorithmic form:

1. Set $j = 0$. For $s = 1, \ldots, S$ draw $\Theta_{(s)} \sim \pi(\Theta)$, $X_{(s)}^0 \sim \pi(X^0|v^0, \Theta_{(s)})$ and $Z_{(s)}^0 \sim \pi(Z^0)$.
2. Set $M = j + m$. For $s = 1, \ldots, S$,
   - Propose (Θ_*, Y_*^j) using (25)
   - Draw $X_*^M \sim \pi(\cdot|v^M, \Theta_*)$.
   - For $i = j, \ldots, M-2$ simulate $Y_*^{i+1} \sim \tilde{\pi}(\cdot|Y_*^i, X_*^M, \Theta_*)$.
   - Draw $Z_*^M \sim \pi(\cdot|Y_*^{M-1}, X_*^M, \Theta_*)$.
   - Set $\Phi_* = (Y_*^j, \ldots, Y_*^M, \Theta_*)$ and put $\Phi_{(s+1)} = \Phi_*$ with probability $\min\{1, \alpha\}$ (where α is given by (27)) else put $\Phi_{(s+1)} = \Phi_{(s)}$.
   - Store $(\Theta_{(s)}, Y_{(s)}^M)$.
3. Set $j = j + m$.
4. Return to step 2.

Thus step 2 performs the update for a given time point. As with any MCMC sampler, this scheme can be modified by allowing a number of iterations to be discarded as "burn-in." A further S iterations may then be performed to generate the desired sample, $\{(\Theta_{(s)}, Y_{(s)}^M), s = 1, \ldots, S\}$, from $\pi_M(\Theta, Y^M)$. Further modifications may be made by thinning the MCMC output at the expense of running the sampler longer. This is done seperately for each time point, with our final posterior sample used as the prior for the next time point.

4. SIMULATION STUDY: PROKARYOTIC AUTO-REGULATORY GENE NETWORK

To illustrate the methodology of Section 3.2, the simulation filter is applied to the diffusion approximation of the regulatory gene network characterized by the reactions (10)–(14).

As well as exploring the fully observed case, we report results for several data-poor contexts; for example, measuring only protein and RNA levels leads to a model with observable part $X(t) = (\mathsf{l}(t), \mathsf{G}(t), \mathsf{r_i}(t), \mathsf{r_g}(t))'$ and unobservable part of the reduced system, $Z(t) = (\mathsf{i}(t), \mathsf{g}(t))'$. Note that formulating the partially observed model in this way implies that we know only the conservation constants, K_1 and K_2 (see (16) and (17)), and not the split into $\mathsf{l}\cdot\mathsf{i}$ and i or $\mathsf{l}\cdot\mathsf{g}$ and g. In practice, it is reasonable to observe K_1 and K_2 as they correspond to the number of copies of each gene on the genome. In Section 4.2, we assume that both K_1 and K_2 are known, but we do not observe $\mathsf{l}\cdot\mathsf{i}(t)$ and $\mathsf{i}(t)$ or $\mathsf{l}\cdot\mathsf{g}(t)$ and $\mathsf{g}(t)$ at any time t.

Realistically, we may have two (or more) independent experimental datasets, one consisting of measurements only on the proteins, $\mathsf{l}(t), \mathsf{G}(t)$ and another on RNA levels, $\mathsf{r_i}(t), \mathsf{r_g}(t)$. Here, we take advantage of the sequential nature of the simulation filter, running the algorithm for each dataset in turn and using the posterior sample obtained from the first dataset as the prior sample for the second.

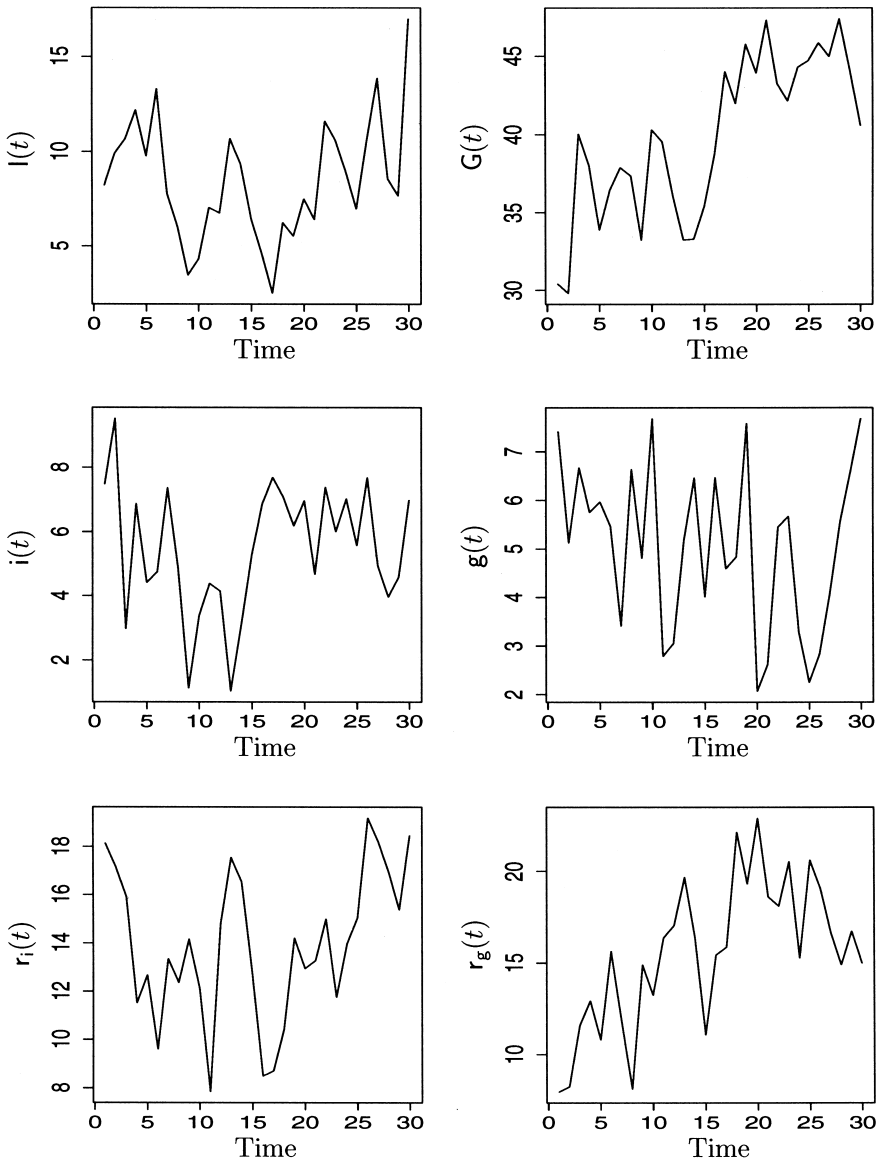

FIG. 1. $\mathcal{D}_1$: 30 (noisy) observations on $(\mathsf{l}(t), \mathsf{G}(t), \mathsf{i}(t), \mathsf{g}(t), \mathsf{r_i}(t), \mathsf{r_g}(t))$.

4.1. Results: Fully observed model

We first implement the MCMC scheme given in Section 3.3 for the fully observed case; we assume that we observe $Y(t) = (\mathsf{l}(t), \mathsf{G}(t), \mathsf{i}(t), \mathsf{g}(t), \mathsf{r_i}(t), \mathsf{r_g}(t))'$ at all times t. We consider equispaced data, $\mathcal{D}_1$, consisting of 30 observations on [0, 29], simulated exactly using the Gillespie algorithm (see Fig. 1). Each data point is subjected to measurement error by adding a Gaussian random variable with zero mean and variance $\sigma^2 = 3$ (so that $\Sigma = \sigma^2 \mathrm{I}$ in (18)), which we assume to be unknown. True values for $(c_1, \ldots, c_{12})$ are chosen to be 0.08, 0.82, 0.09, 0.9, 0.25, 0.1, 0.35, 0.3, 0.1, 0.1, 0.12, 0.1, and we place Uniform U(−5, 1) priors on each $\log(c_i)$, for $i = 1, \ldots, 12$ and σ. Note that K_1 and K_2 (the number of copies of each gene) are set to be 10. **F1**

The simulation filter is run for five million iterations with a thin of 250, giving a final sample of size $S = 20{,}000$. Discretization is set by running the MCMC algorithm with $m = 5, 8, 16, 20$. Figure 2 and Table 1 summarize the posterior distributions; trace, density, and autocorrelation plots can be seen in Fig. 2 **F2, T1**

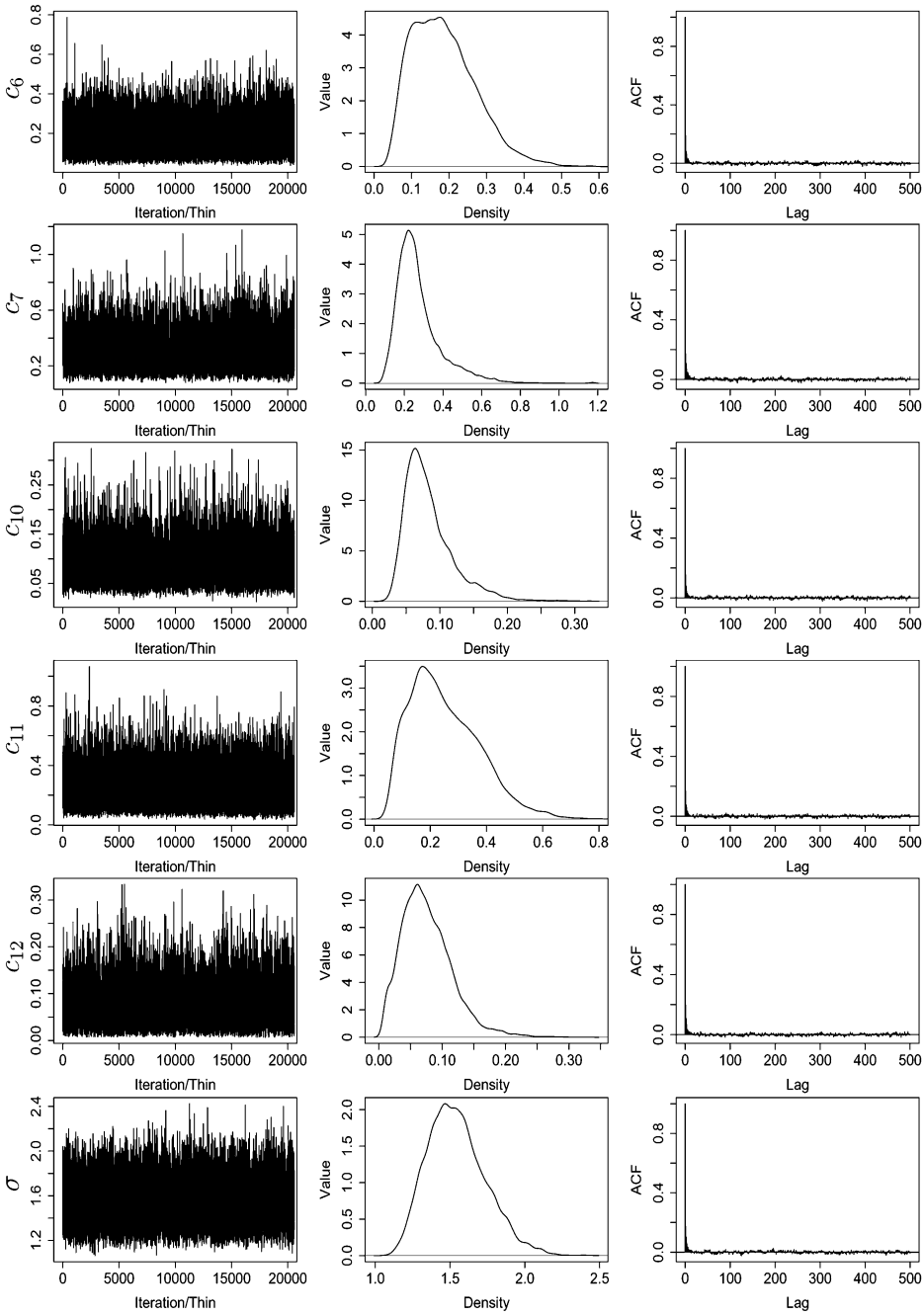

FIG. 2. Trace, density and autocorrelation plots for c_6, c_7, c_{10}, c_{11}, c_{12} and σ for the fully observed model using 30 observations and $m = 20$. Results are based on a final sample of size 20,000, thinned from 5,000,000 MCMC iterations. AU2/QU2

for a selection of parameters with $m = 20$. Table 1 reports posterior means and standard deviations for Θ, based on the output from the simulation filter for each choice of m.

As the estimated MCMC error is related to the autocorrelations within the chains, the relative performance of the simulation filter can be assessed by studying the sample autocorrelation functions for each parameter. Figure 2 shows that autocorrelations die down very quickly despite large m. We also see that the sampler produces estimates close to the true values that generated the data.

TABLE 1. POSTERIOR MEANS AND STANDARD DEVIATIONS FOR Θ ESTIMATED ON 30 OBSERVATIONS ($\mathcal{D}_1$) FROM THE FULLY OBSERVED MODEL[a]

		Mean (standard deviation)			
Parameter	*True value*	$m = 5$	$m = 8$	$m = 16$	$m = 20$
c_1	0.08	0.010 (0.003)	0.034 (0.030)	0.038 (0.033)	0.029 (0.017)
c_2	0.82	0.112 (0.093)	0.455 (0.427)	0.431 (0.447)	0.443 (0.321)
c_1/c_2	0.096	0.119 (0.115)	0.102 (0.087)	0.088 (0.058)	0.088 (0.048)
c_3	0.09	0.010 (0.002)	0.012 (0.006)	0.027 (0.021)	0.043 (0.021)
c_4	0.9	0.070 (0.050)	0.130 (0.112)	0.268 (0.256)	0.374 (0.209)
c_3/c_4	0.1	0.115 (0.115)	0.107 (0.349)	0.138 (0.069)	0.129 (0.057)
c_5	0.25	0.341 (0.208)	0.336 (0.294)	0.325 (0.281)	0.304 (0.218)
c_6	0.1	0.302 (0.168)	0.299 (0.377)	0.276 (0.196)	0.189 (0.085)
c_7	0.35	0.121 (0.071)	0.222 (0.159)	0.267 (0.194)	0.275 (0.112)
c_8	0.3	0.056 (0.031)	0.146 (0.058)	0.149 (0.023)	0.163 (0.074)
c_9	0.1	0.048 (0.037)	0.099 (0.094)	0.069 (0.030)	0.078 (0.061)
c_{10}	0.1	0.031 (0.023)	0.080 (0.059)	0.081 (0.042)	0.084 (0.038)
c_{11}	0.12	0.369 (0.238)	0.321 (0.473)	0.285 (0.125)	0.258 (0.126)
c_{12}	0.1	0.023 (0.015)	0.060 (0.027)	0.064 (0.015)	0.076 (0.040)
σ	1.732	1.846 (0.205)	1.799 (0.250)	1.697 (0.223)	1.647 (0.195)

[a]Estimation results are based on a final sample of size 20,000, thinned from 5,000,000 MCMC iterations.

Inspection of Table 1 reveals the advantage of including latent variables in the estimation framework. For large m, there is a notable decrease in discretization error; for example, c_{10} (the stochastic rate constant for mRNA degradation) has a true value of 0.1 while it was estimated to be 0.031 with $m = 5$ and 0.084 with $m = 20$. Similarly the standard deviation of the measurement error, σ, was estimated to be 1.846 with $m = 5$, 1.697 with $m = 16$, and 1.647 with $m = 20$, and has a true value of 1.732. Note also that although estimates of c_1, c_2, c_3, and c_4 appear imprecise (perhaps due to the small number of observations), estimates of c_1/c_2 and c_3/c_4 (corresponding to the propensities of reactions R_1 and R_3 respectively) are fairly accurate for all choices of m.

4.2. Results: Partially observed model

We now apply the MCMC algorithm to the partially observed model. We consider three equispaced datasets, $\mathcal{D}_2$, $\mathcal{D}_3$, and $\mathcal{D}_4$, each independently simulated using the Gillespie algorithm with stochastic rate constants, $c_1, \ldots, c_{12}$ as in Section 4.1. Dataset $\mathcal{D}_2$ consists of 30 observations on $X(t) = (\mathsf{I}(t), \mathsf{G}(t), \mathsf{r_i}(t), \mathsf{r_g}(t))'$ with each data point subject to measurement error with variance $\sigma^2 = 3.0$. Dataset $\mathcal{D}_3$ contains 40 observations on protein levels only; $X(t) = (\mathsf{I}(t), \mathsf{G}(t))'$ and the variance of the measurement error is $\sigma^2 = 3.0$. Finally, $\mathcal{D}_4$ contains 20 observations on RNA levels; $X(t) = (\mathsf{r_i}(t), \mathsf{r_g}(t))'$ with $\sigma^2 = 2.0$. For each dataset, we assume that the variance of the measurement error is known and that the number of copies of each gene is known to be $K_1 = K_2 = 10$. As in Section 4.1, we place uniform priors on each $\log(c_i)$ and also on $\log(Z^0)$.

The simulation filter is run for each dataset for four million iterations with a thin of 200, giving a final sample of size $S = 20{,}000$. Discretization is set by running the algorithm with $m = 20$. Table 2 summarizes the posterior distribution for each dataset. In addition, we provide summaries of $\pi(\Theta|\mathcal{D}_3, \mathcal{D}_4)$, obtained by using the posterior sample from $\pi(\Theta|\mathcal{D}_3)$ as the prior sample for data $\mathcal{D}_4$. Figure 3 shows posterior densities of c_5, c_7, c_{11}, and c_{12} given data $\mathcal{D}_3$, $\mathcal{D}_4$ and $(\mathcal{D}_3, \mathcal{D}_4)$.

Intuitively, when observing just two species, estimates are generally more accurate for rate constants governing reactions involving those species. For example, c_9 and c_{10} (pertaining to RNA degradation reactions given by (13)), both with true values of 0.1, are estimated to be 0.148 and 0.201 respectively when using 40 observations on protein levels ($\mathcal{D}_3$). However, when using just 20 observations on RNA levels ($\mathcal{D}_4$), we see an increase in accuracy with estimates of 0.099 and 0.097, respectively. Similarly, c_1/c_2 and c_3/c_4, the propensities of repression reactions R_1 and R_2 have true values of 0.096 and 0.1.

TABLE 2. POSTERIOR MEANS AND STANDARD DEVIATIONS FOR PARAMETERS ESTIMATED USING DATASETS $\mathcal{D}_2$, $\mathcal{D}_3$, AND $\mathcal{D}_4$ FROM THE PARTIALLY OBSERVED MODEL[a]

		Mean (standard deviation)			
Parameter	*True value*	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$	$(\mathcal{D}_3, \mathcal{D}_4)$
c_1	0.08	0.024 (0.019)	0.051 (0.050)	0.049 (0.057)	0.067 (0.065)
c_2	0.82	0.254 (0.231)	0.391 (0.356)	0.357 (0.364)	0.496 (0.367)
c_1/c_2	0.096	0.176 (0.224)	0.241 (0.305)	0.414 (0.916)	0.227 (0.339)
c_3	0.09	0.031 (0.022)	0.032 (0.031)	0.037 (0.057)	0.023 (0.023)
c_4	0.9	0.214 (0.244)	0.255 (0.299)	0.225 (0.292)	0.186 (0.238)
c_3/c_4	0.1	0.204 (0.319)	0.290 (0.361)	0.507 (0.951)	0.345 (0.467)
c_5	0.25	0.418 (0.296)	0.232 (0.303)	0.478 (0.285)	0.426 (0.237)
c_6	0.1	0.072 (0.066)	0.058 (0.087)	0.140 (0.230)	0.038 (0.060)
c_7	0.35	0.228 (0.159)	0.211 (0.281)	0.637 (0.353)	0.608 (0.327)
c_8	0.3	0.275 (0.100)	0.526 (0.283)	0.262 (0.341)	0.425 (0.295)
c_9	0.1	0.133 (0.081)	0.148 (0.192)	0.099 (0.089)	0.091 (0.068)
c_{10}	0.1	0.046 (0.037)	0.201 (0.250)	0.097 (0.087)	0.112 (0.090)
c_{11}	0.12	0.076 (0.083)	0.047 (0.047)	0.203 (0.279)	0.057 (0.083)
c_{12}	0.1	0.103 (0.041)	0.093 (0.037)	0.148 (0.226)	0.086 (0.044)

[a]Discretization is set at $m = 20$ and the estimation results are based on a final sample of size 20,000, thinned from 4,000,000 MCMC iterations.

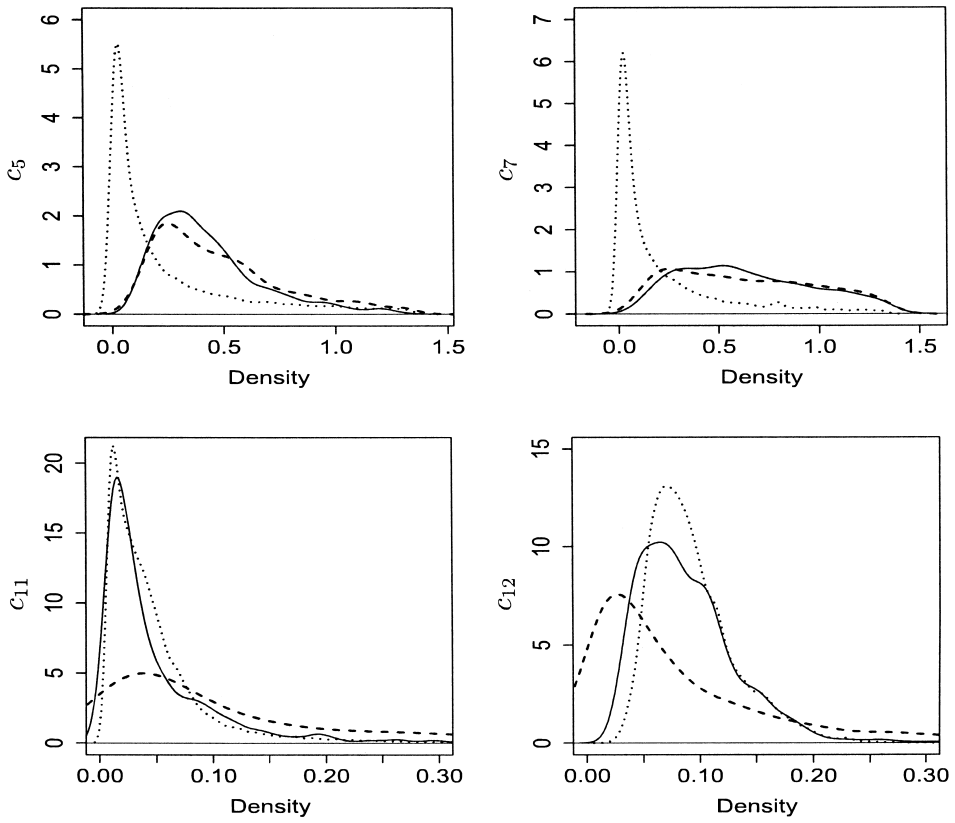

FIG. 3. Posterior density plots for c_5, c_7, c_{11}, and c_{12} obtained from $\mathcal{D}_3$ (dotted line), $\mathcal{D}_4$ (dashed line), and $(\mathcal{D}_3, \mathcal{D}_4)$ (solid line). Results are based on a final sample of size 20,000, thinned from 4,000,000 MCMC iterations.

Using the protein data ($\mathcal{D}_3$) we see estimates of 0.241 and 0.290 respectively, whilst using just RNA levels, estimates are very poor (0.414 and 0.507).

Running our algorithm for the RNA data with a prior, $\pi(\Theta)$ given by the posterior sample for parameters given protein data, $\pi(\Theta|\mathcal{D}_3)$ yields, in general, fairly precise estimates of those parameters governing RNA reactions and also of parameters governing protein reactions. For example, c_1/c_2 and c_3/c_4 are now estimated to be 0.227 and 0.345 whilst posterior means for c_9 and c_{10} are 0.091 and 0.112, respectively. As we would expect, for those parameters governing protein-only reactions, $\pi(\Theta|\mathcal{D}_3, \mathcal{D}_4)$ is dominated by the prior, $\pi(\Theta|\mathcal{D}_3)$, as this is in fact the posterior obtained when using only protein data, $\mathcal{D}_3$ (see Fig. 3).

Finally, it appears that we gain the most information by observing as many species as possible in a single experiment rather than combining datasets on a few species obtained from multiple independent experiments. This can be seen by comparing the columns in Table 2 corresponding to $\mathcal{D}_2$ (30 observations on RNA and protein levels) and $(\mathcal{D}_3, \mathcal{D}_4)$.

5. DISCUSSION

We have implemented a sequential Bayesian approach to conduct rigorous inference for rate constants governing biochemical reactions. By adopting a diffusion approximation, the solution to the problem of reverse engineering rate constants from noisy time course data corresponds to the estimation of nonlinear, discretely (and perhaps partially) observed stochastic differential equations. However, the task of inferring parameters in SDEs is not trivial. The estimation framework necessarily introduces missing values and high dependence between these latent values and parameters results in poor mixing properties of MCMC schemes such as the Gibbs sampler (Roberts and Stramer, 2001). The utility of the simulation filter is two-fold; first, by performing a joint update of the parameters and missing values at each time point, we can overcome the dependence between them (Golightly and Wilkinson, 2005b). Also, as the method is sequential, we can use a posterior sample obtained form one dataset as the prior for the next, allowing us to handle multiple datasets from different experiments.

The methodology was applied to synthetic data generated from a prokaryotic auto regulatory gene network model. Naturally, the integration of actual measurements into the modeling framework remains of great interest and is the subject of ongoing research.

REFERENCES

Allen, L.J.S. 2002. *An Introduction to Stochastic Processes with Applications to Biology*, Prentice Hall, Englewood Cliffs, NJ.

Arkin, A., Ross, J., and McAdams, H.H. 1998. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected *Escherichia coli* cells. *Genetics* 149, 633–648.

Berzuini, C., Best, N.G., Gilks, W.R., and Larizza, C. 1997. Dynamic conditional independence models and Markov chain Monte Carlo methods. *J. Am. Statist. Assoc.* 92(440), 1403–1412.

Bower, J.M., and Bolouri, H. 2000. *Computational Modeling of Genetic and Biochemical Networks*, The MIT Press, London.

Boys, R.J., Wilkinson, D.J., and Kirkwood, T.B.L. 2004. Bayesian inference for a discretely observed stochastic-kinetic model. In submission.

Doraiswamy, L.K., and Kulkarni, B.D. 1987. The analysis of chemically reacting systems, vol. 4 of *Topics in Chemical Engineering*, Gordon and Breach Science Publishers. AU3

Doucet, A., Godsill, S., and Andrieu, C. 2000. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* 10, 197–208.

Durham, G.B., and Gallant, R.A. 2002. Numerical techniques for maximum likelihood estimation of continuous-time diffusion processes. *J. Bus. Econ. Statist.* 20, 279–316.

Elerian, O., Chib, S., and Shephard, N. 2001. Likelihood inference for discretely observed nonlinear diffusions. *Econometrica* 69(4), 959–993.

Eraker, B. 2001. MCMC analysis of diffusion models with application to finance. *J. Bus. Econ. Statist.* 19, 177–191.

Finch, C.E., and Kirkwood, T.B.L. 2000. *Chance, Development and Ageing*, Oxford University Press, New York.

Gillespie, D.T. 1977. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* 81, 2340–2361.

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

Golightly, A., and Wilkinson, D.J. 2005b. Bayesian sequential inference for nonlinear multivariate diffusions. In submission.

Jones, C. 1997. A simple Bayesian approach to the analysis of Markov diffusion processes. Technical report, The Wharton School, University of Pennsylvania.

Kitano, H. 2001. *Foundations of Systems Biology*, The MIT Press, London.

Kloeden, P.E., and Platen, E. 1992. *Numerical Solution of Stochastic Differential Equations*, Springer Verlag, New York.

Latchman, D. 2002. *Gene Regulation: A Eukaryotic Perspective*, BIOS Scientific Publishers. AU4

Liu, J., and West, M. 2001. Combined parameter and state estimation in simulation-based filtering, *in* Doucet, A., de Freitas, N., and Gordon, N., eds., *Sequential Monte Carlo Methods in Practice*. AU5

McAdams, H.H., and Arkin, A. 1999. Its a noisy business: Genetic regulation at the nanomolar scale. *TIG* 15, 65–69.

McQuarrie, D.A. 1967. Stochastic approach to chemical kinetics. *J. Appl. Probab.* 4, 413–478.

Ng, T.W., Wilkinson, D.J., and Kirkwood, T.B.L. 2004. Stochastic modelling, simulation and inference of gene regulatory networks. In submission.

Øksendal, B. 1995. *Stochastic Differential Equations: An Introduction with Applications*, 6th ed., Springer-Verlag, New York.

Pedersen, A. 1995. A new approach to maximum likelihood estimation for stochastic differential equations based on discrete observations. *Scand. J. Statist.* 1995(22), 55–71.

Pitt, M.K., and Shephard, N. 1999. Filtering via simulation: Auxiliary particle filters. *J. Am. Statist. Assoc.* 446(94), 590–599.

Ptashne, M. 1992. *A Genetic Switch: Phage λ and Higher Organisms*, 2nd ed., Cell Press and Blackwell Scientific Publications. AU6

Roberts, G.O., and Stramer, O. 2001. On inference for partially observed nonlinear diffusion models using the Metropolis-Hastings algorithm. *Biometrika* 88(4), 603–621.

Tanner, M.A., and Wong, W.H. 1987. The calculation of posterior distributions by data augmentation. *J. Am. Statist. Assoc.* 82(398), 528–540.

van Kampen, N.G. 2001. *Stochastic Processes in Physics and Chemistry*, North-Holland. AU7

West, M. 1993. Approximating posterior distributions by mixtures. *J. Roy. Stat. Soc. Ser. B Stat. Methodol.* 55, 409–422.

Wilkinson, D.J. 2003. Discussion to 'Non centred parameterisations for hierarchical models and data augmentation' by Papaspiliopoulos, Roberts, and Skold, *in Bayesian Statistics 7*, 323–324, Oxford Science Publications, London.

Address correspondence to:
Andrew Golightly
School of Mathematics and Statistics
University of Newcastle upon Tyne
NE1 7RU, UK

E-mail: a.golightly@ncl.ac.uk

Author
Right running head okay as shown (short title)?

AU1
Missing word or symbol: "and is simply"?

AU2/QU2
Figure 2 caption and artwork okay as shown? No manuscript provided.

AU3
Publisher's city?

AU4
Publisher's city?

AU5
Publisher's city?

AU6
Publisher's city?

AU7
Publisher's city?