

Bayesian inference for stochastic kinetic genetic regulatory networks

Darren J. Wilkinson* & Richard J. Boys

University of Newcastle upon Tyne

1 Introduction

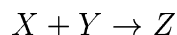
We describe the techniques used to model genetic and biochemical networks, together with the computational tools needed for stochastic simulation and analysis. An overview is also given of the MCMC algorithms which can be used for carrying out Bayesian inference for the parameters underlying the network models, and the problems associated with applying such techniques in practice.

2 Genetic regulatory networks

- These are the complex biochemical mechanisms for controlling and regulating the expression of genes and proteins in a cell
- There are many different kinds of regulatory control mechanisms
 - We are mainly interested in *transcription control*
 - For a gene to be expressed as a protein, it must first be *transcribed* into mRNA, then *translated* into a polypeptide chain which folds to form a protein

3 Stochastic kinetics

Consider a simple bi-molecular reaction:



We assume that any particular pair of molecules (one molecule of X and one molecule of Y) has a *constant probability of reaction* ie. the probability of the pair reacting in time dt is θdt , for some fixed reaction rate θ , which may depend on the volume of the container (Gillespie, 1992).

A general stochastic-kinetic reaction network is therefore Markov process in continuous time with countable state space.

Suppose there are K coupled reaction types:

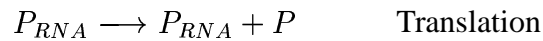
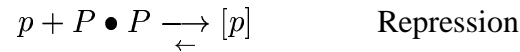
- the *time to next event* is exponential
- reaction k occurs at rate a_k (a_k depends on the current state)
- eg. for a bi-molecular reaction involving X and Y , $a_k = xy\theta_k$ (as there are xy pairs)
- the next reaction will be of type k with probability proportional to a_k

- the next reaction will occur with rate $a = \sum_{k=1}^K a_k$

The discrete-event simulation procedure for this process is known to physical scientists as the *Gillespie algorithm* (Gillespie, 1977)

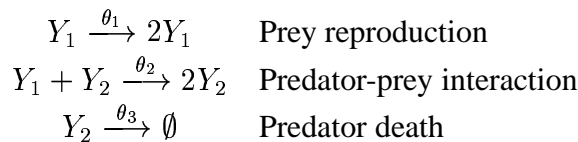
4 Examples

4.1 Auto-regulation



This is a very simple model of the simplest interesting “real” network (McAdams and Arkin, 1997).

4.2 Lotka-Volterra system



This is the simplest system that exhibits many of the interesting non-linear feedback features present in the above auto-regulatory network, and hence is an ideal model system for investigating inferential algorithms.

5 Bayesian inference

Inference is required for the rates of the reactions (at least). Prior information is available in the literature: some rates (almost) known, *expert knowledge* available for many others, and very little knowledge about some. We want to combine *available knowledge* with *experimental data* in order to get improved parameter estimates, etc. Given *complete information* on the initial state, all reactions and all reaction times, the *likelihood is analytically tractable*, and inference is straightforward. However, we typically only observe levels of certain reporter species, measured at discrete time intervals, subject to measurement error. We therefore have a data augmentation problem for a latent process model.

5.1 Inference for the Lotka-Volterra model

- 2 species: Y_1, Y_2
- Stochastic process: $\{Y_1(t), Y_2(t) : t \geq 0\}$ with parameters: $\theta = (\theta_1, \theta_2, \theta_3)$
- Observe: $y = \{y_1(t), y_2(t) : t = 0, 1, 2, \dots, T\}$

5.2 Reversible-jump MCMC

It is possible to implement a reversible-jump MCMC approach to imputation of the latent process, but mixing of the resultant sampler is poor, and does not compare well with the block-updating algorithm considered next. See Gibson and Renshaw (2001) for the use of similar techniques in the context of stochastic compartmental models.

5.3 Block-updating strategy

We want to simulate the latent process conditional on the parameters and the observed data.

Consider block $[i, i + 1)$:

- the number of reactions of each type (r_1, r_2, r_3) are unknown, but subject to 2 constraints
- if we set r_1 , then r_2 and r_3 are determined:

$$r_2 = y_1(i) - y_1(i + 1) + r_1, \quad r_3 = y_1(i) - y_1(i + 1) + y_2(i) - y_2(i + 1) + r_1$$

Basic idea: simulate a proposed new interval from an approximating multivariate inhomogeneous Poisson process with a linear rate function, and then correct for this approximation using a *Metropolis-Hastings* step

The resulting MCMC algorithm is *exact*, in the sense that the Markov chain has as its equilibrium distribution the exact posterior distribution of the parameters and the latent process given the data.

5.4 Approximate algorithm

We can create an approximate version of the previous algorithm, where we *do not correct* for the inhomogeneous Poisson process approximation. This is the only place that *reaction times* come in to the picture, so reaction times no longer need to be simulated or stored — only the total number of reactions in each interval is required. This makes the approximate algorithm *much faster* than the exact algorithm (roughly 10 times faster on this problem, but gains will be even greater on more complex problems). Empirical evidence suggests that the *approximation is good enough* for most purposes — the resulting marginal posteriors look very similar to those from the exact algorithms .

5.5 Partially observed process

Suppose that we *only observe the number of prey* at each of our observation time points, because it is hard to measure the number of predators. Can we still make inferences for all three reaction rates (and the predator numbers), or does the model become *unidentifiable*?

Basic idea: *update intervals in pairs* — keep prey and predator numbers fixed at the two ends, but in the middle keep the prey numbers fixed and allow the predator numbers to vary.

First update intervals (1, 2), then (2, 3), (3, 4), etc. (the updating of overlapping blocks is not a problem)

It is reasonably straightforward to implement this algorithm, but there are problems with the MCMC output due to the weakly identifiable nature of the underlying model — *very* long runs are required (with lots of thinning) in order to get good results. However, the algorithm does work, and the model *is* identifiable.

5.6 General strategy

The Lotka-Volterra system is an ideal “proof-of-concept” system for demonstrating the basic principles one can follow to do inference in stochastic kinetic models, but a general strategy is now required.

- First construct the reaction updating matrix for the model
- Partition this matrix (in a “good” way) into free and determined reaction types
- Intervals can then be updated by perturbing the free types, setting the rest, and proposing a new interval consistent with the new reaction numbers

6 Further reading

Further details regarding the Lotka-Volterra application can be found in Boys and Wilkinson (2004). For information regarding stochastic kinetic models and software for inference, consult the web site: <http://www.ncl.ac.uk/math/research/statistics/bioinformatics/networks.htm>

Acknowledgement: This work is supported by the UK BBSRC Bioinformatics initiative (Grant # BIO14454).

References

- Boys, R.J. and Wilkinson, D.J. (2004) Bayesian inference for a discretely observed stochastic-kinetic Lotka-Volterra model, *in submission*.
- Gillespie, D.T. (1977) Exact stochastic simulation of coupled chemical reaction, *Journal of Physical Chemistry*, **81**:2340–2361.
- Gillespie, D.T. (1992) Markov processes: An introduction for physical scientists, *Academic Press*.
- McAdams, H.H. and Arkin, A. (1997) Stochastic mechanisms in gene expression, *Proceedings of the National Academy of Sciences, USA*, **94**: 814–819.
- Gibson, G.J. & Renshaw, E. (2001) Likelihood estimation for stochastic compartmental models using Markov chain methods, *Statistics and Computing*, **11**: 347–358.