# Parameter Estimation of Kinetic Rates in Stochastic Reaction Networks by the EM Method

András Horváth and Daniele Manini

Dipartimento di Informatica, Università di Torino
Corso Svizzera 185, 10149 Torino, Italy
{horvath,manini}@di.unito.it

## Abstract

*Gillespie's algorithm serves to simulate a network of stochastic reactions with given initial quantities and kinetic rate constants. In this paper we consider the estimation of the kinetic rate constants of the reactions based on a set of discrete observations generated by Gillespie's algorithm. In particular, we present an Expectation Maximisation (EM) method to perform maximum likelihood estimation of the rate constants. Applicability of the method is tested on a simple reaction network.*

## 1. Introduction

The algorithm introduced by Gillespie [8] performs stochastic simulation of reaction networks, it allows to model chemical or biochemical kinetics. When applying this simulation method the kinetic rate constants (called simply kinetic rates from now on) of the reactions must be set. Most studies that deal with the identification of these parameters are based on the deterministic model of the reaction networks, i.e., the dynamics of the model is described by a set of ordinary differential equations (ODEs). Usually the kinetic rates are identified by applying optimisation methods [12, 7, 15]. The obtained rates can be transformed into the kinetic rates of the stochastic model. As pointed out however in [14] this is not always possible. For this reason some recent studies have attempted to give an estimate of the kinetic rates based on the stochastic view introduced by [8]. Bayesian inference methods were used in [10, 2], maximum likelihood methods were applied in [14, 4, 16].

In this paper, we study, to the best of our knowledge for the first time, the applicability of the EM method for the estimation of kinetic rates. The EM method, which is an approach to maximum likelihood estimation, has the advantage that it allows to incorporate partial knowledge on the model into the parameter identification procedure.

The paper is organised as follows. Section 2 gives a brief introduction to the EM method. Section 3 describes the proposed parameter estimation procedure. In Section 4 we test the procedure on a simple reaction network. Conclusions are drawn in Section 5.

## 2. The EM method

A widely applicable parameter estimation procedure for incomplete data problems is the EM algorithm. Formulated in [6], it uses an iterative scheme to maximise the likelihood (or, equivalently, the log likelihood).

In order to briefly describe the EM method, the following notation is introduced. Let $X$, $Y$ and $Z$ be the random variables representing the complete, the known and the missing data, respectively. Assume that the actual values of these random variables are $x$, $y$ and $z$ of which we know only $y$. Let $\mathcal{P}$ denote the actual set of parameters that we aim to modify in such way that the likelihood increases. The EM algorithm proceeds by repeating the following two steps.

The *E-step*, where E stands for expectation, aims to reconstruct the complete data based on the known data, $y$, and the actual set of parameters, $\mathcal{P}$. This is done by calculating the conditional expectation

$$z' = E\left[Z|Y = y, \mathcal{P}\right] \tag{1}$$

and adding the result, $z'$, to the known part of data, $y$, forming the expected complete data $x'$.

In the *M-step* of the procedure, where M stands for maximisation, we choose a new set of parameters, $\mathcal{P}'$, in such a way that the likelihood of observing $x'$ is maximal. With the resulting set of parameters, $\mathcal{P}'$, we go back to the *E-step*. The iteration is stopped when the likelihood function cannot be increased anymore.

There are several extensions to the original EM method. The one of most interest in our case is the Monte Carlo EM method presented in [17]. The authors suggests that if the

computation of the missing data, $z'$, according to (1) is too hard either computationally or conceptually, then augmentation of the incomplete data can still be done by simulation. If simulation is used in the *E-step* the monotonicity property of the original EM method is lost but, as it is shown in [1], the algorithm gets close to a maximum with high probability.

## 3. Parameter estimation by the EM method

In this section, first, we formulate the problem and then show how it can be tackled by the Monte Carlo EM method. Finally, implementation is discussed briefly.

### 3.1. Formulation of the problem

Assume that we are given a network of reactions with unknown kinetic rates and experimental observations of quantities of the involved species at $N$ time instants. We assume that the reactions follow the stochastic dynamics described by Gillespie in [8]. Our aim is to give a maximum likelihood estimate for the kinetic rates of the reactions.

For what concerns notation, $t_i, 1 \leq i \leq N$, denote the time instants at which we have the observations while the quantity of the species at time $t_i$ is given by vector $Y_i$, i.e., $t_i$ and $Y_i$ represent the known data.

We describe the problem as we had observations connected to a single experiment, however, it is straightforward to extend it to the multiple experiment case.

### 3.2. Starting values for the parameters

First of all, we have to find reasonable starting values for the kinetic rates. This can be done by considering the deterministic model of the reactions described by ODEs. In particular, we choose such vector of initial rates, $\bar{c}_0$, that the sum of the relative errors between the available observations, $Y_i, 1 \leq i \leq N$, and the deterministic behaviour described by the ODEs is minimal.

### 3.3. E-step

In the *E-step* we have to form complete traces based on the observations, $Y_i, 1 \leq i \leq N$, and the actual vector of the kinetic rates, $\bar{c}_k$. Ideally, one should create such trace that the process is in state $Y_i$ at time $t_i$ and the rest of the trace is the expected behaviour assuming that the parameters are $\bar{c}_k$. This problem is hard both conceptually and computationally because Gillespie's algorithm is the simulation of a Markov chain that can have huge state space even in case of the simplest reaction networks. For this reason we opt for applying the Monte Carlo EM method. Even though the *E-step* is still not straightforward because generating such simulation traces that are in state $Y_i$ at time $t_i, 1 \leq i \leq N$, is not simple. Hence we apply an approximation.

In order to form a complete trace between $t_i$ and $t_{i+1}$, $1 \leq i < N-1$, we perform the following steps.

**Step 1.** We generate $M_s$ simulation traces with initial state $Y_i$ up to time $t_{i+1} - t_i$ and choose the one among these traces that arrived closest to $Y_{i+1}$, where closest means that the sum of the relative errors in the components is the smallest. Note that since the underlying Markov chain can be huge and parameters $\bar{c}_k$ can be far from the real parameters, only a very high number of simulation runs could guarantee that a trace arrives close to $Y_{i+1}$.

**Step 2.** In order to avoid the generation of a very high number of simulations, we "improve" the best trace we find among a relatively small number of simulation runs. Assume that the best trace is

$$Y_i = Z_0 \xrightarrow{r_1,s_1} Z_1 \xrightarrow{r_2,s_2} Z_2 \dots \xrightarrow{r_L,s_L} Z_L$$

where $r_i$ denotes the reactions, $s_i$ the sojourn times and $L$ is such that $\sum_{j=1}^{L} s_j \leq t_{i+1} - t_i < \sum_{j=1}^{L+1} s_j$. Let $M_i$ denote the fraction of reactions that we modify during the improvement of the trace. We repeat $\lfloor M_i L \rfloor, 0 < M_i < 1$, times the following step. We pick up one of the $L$ reaction in a random manner and check if that single reaction can be substituted by another reaction in such a way that the rest of the trace is still possible (i.e., the reactions after the modified one are still possible to perform) and the final state of the trace is closer to $Y_{i+1}$ than it was before. The sojourn times remain unchanged. Substituting reaction $r_l$ the trace becomes

$$Y_i = Z_0 \xrightarrow{r_1,s_1} Z_1 \dots \xrightarrow{r_{l-1},s_{l-1}} Z_{l-1} \xrightarrow{r'_l,s_l} Z'_l \xrightarrow{r_{l+1},s_{l+1}} Z'_{l+1} \dots \xrightarrow{r_{L-1},s_{L-1}} Z'_{L-1} \xrightarrow{r_L,s_L} Z'_L \,.$$

Note that the substitution change the likelihood of the trace but if the number of molecules is not close to 0 this alteration is minimal. Choosing $M_i$ is crucial because a too high value can lead to an improbable trace while with a too low value the traces are not improved enough which can slow down convergence.

### 3.4. M-step

In the *M-step* we have to find such new kinetic rates, $\bar{c}_{k+1}$, that the traces generated in the *E-step* has maximal likelihood. Assume that the traces are given by

$$Z_{i,0} = Y_i \xrightarrow{r_{i,1},s_{i,1}} Z_{i,1} \xrightarrow{r_{i,2},s_{i,2}} Z_{i,2} \dots \xrightarrow{r_{i,L_i},s_{i,L_i}} Z_{i,L_i}$$

for $1 \leq i \leq N-1$ where $L_i$ is the number of reactions between $t_i$ and $t_{i+1}$. Then the likelihood of the traces can

be calculated as

$$\prod_{i=1}^{N-1}\Bigg[\Bigg(\prod_{l=0}^{L_i-1} p(Z_{i,l}, r_{i,l+1}, \bar{c}_{k+1}) \qquad (2)$$

$$\lambda(Z_{i,l}, \bar{c}_{k+1}) \exp(-\lambda(Z_{i,l}, \bar{c}_{k+1}) s_{i,l+1})\Bigg) \qquad (3)$$

$$\exp\Bigg(-\lambda(Z_{i,L_i}, \bar{c}_{k+1})\Bigg(t_{i+1} - \sum_{j=1}^{L_i} s_{i,j}\Bigg)\Bigg)\Bigg] \qquad (4)$$

where $p(Z, r, \bar{c})$ stands for the probability that if the process is in state $Z$ and the kinetic rates are according to $\bar{c}$ then the next reaction that takes place is $r$; $\lambda(Z, \bar{c})$ stands instead for the sum of the intensities (propensities) of all the reactions that are possible in $Z$ given the parameters in $\bar{c}$. Both $p(Z, r, \bar{c})$ and $\lambda(Z, \bar{c})$ are straightforward to calculate, see [8] for details. Let us provide some explanation for (2-4). Part (2) gives simply the probability that the next reaction will be the one given in the traces. In part (3) we multiply by the density that the reaction will take place after the amount of time given in the trace. Finally, (4) gives the probability that no reaction takes place between arriving to the last state of the trace and $t_{i+1}$. By appropriate multiplication of these quantities we get the likelihood.

### 3.5. Implementation

The whole procedure has been implemented in a Matlab prototype tool. The optimisation required to determine the starting values (Section 3.2) and to maximise the log-likelihood function (Section 3.4) was performed by the *fmincon* function.

## 4. Numerical example

As an example to illustrate the method we consider the Lotka reactions as given in [8]

$$\bar{X} + Y_1 \xrightarrow{c_1} 2Y_1, \;\; Y_1 + Y_2 \xrightarrow{c_2} 2Y_2, \;\; Y_2 \xrightarrow{c_3} Z$$

where the *bar* over $X$ indicates that its population is assumed to remain constant. A simulation trace, generated according to the Gillespie algorithm, is depicted in Figure 1 with initial values $X = 1, Y_1 = 2000, Y_2 = 2000, Z = 0$ and reaction rates $\bar{c} = |c_1\ c_2\ c_3| = |10\ 0.01\ 10|$. We assume to know the number of $Y_1$ and $Y_2$ molecules at time instants $t_i = (i-1)\ 0.1, 1 \le i \le N = 10$.

We apply the ODE view of the process, as it is briefly described in Section 3.2, to obtain the starting values which result to be $\bar{c}_0 = |1.003\ 0.4558\ 6.563|$. Figure 2 shows the deterministic behaviour with the original parameters, with the starting parameters, $\bar{c}_0$, and the samples as well. It can be observed that $\bar{c}_0$ is a very poor estimation of the original values and that the simulation trace does not follow precisely the behaviour described by the ODE.

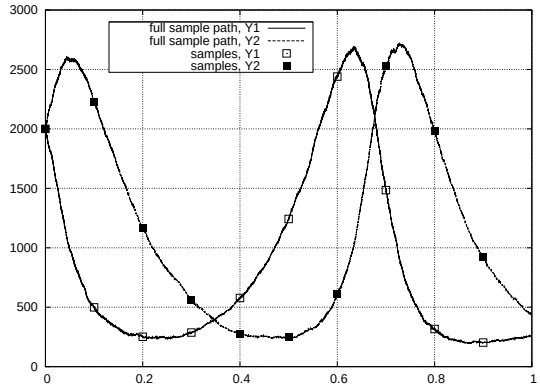


**Figure 1. A simulation run with associated samples and the ODE trajectory**

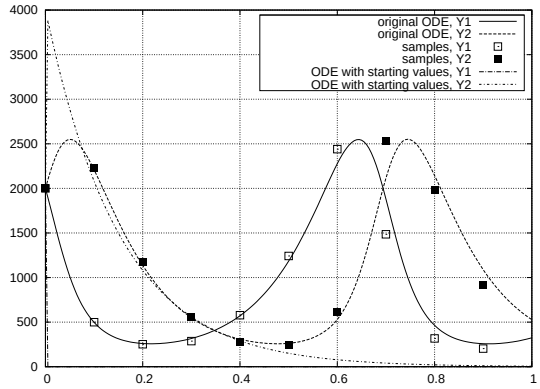


**Figure 2. Original ODE trajectory and ODE trajectory with starting values**

Figure 3 depicts the relative error with respect to the original parameters as the function of the number of iterations ($c_1$ is found exactly after 20 iterations and hence the corresponding error disappears from the figure) and gives the value of the log-likelihood function as well. Parameters for this run were $M_s = 50$ and $M_i = 0.1$. Up to 25 iterations the likelihood is increasing and the parameters are clearly getting closer to the original ones. After 25 iterations, even if there is no improvement, the parameters are still changing because of the random effect caused by the Monte Carlo simulation. The calculations took one and a half hour on a standard portable computer with a 1.5GHz processor. We believe that a more efficient non-Matlab implementation could drastically speed up the calculations.

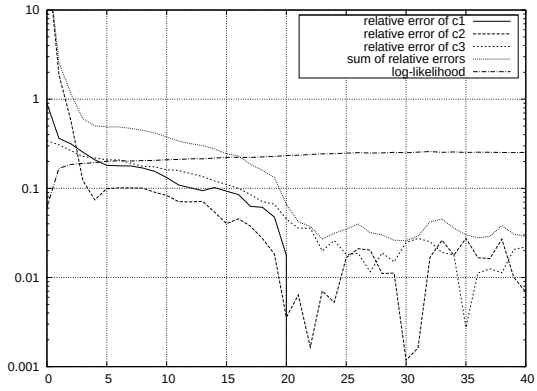


**Figure 3. Relative errors and log-likelihood as function of number of iterations for $M_s = 50$ and $M_i = 0.1$**

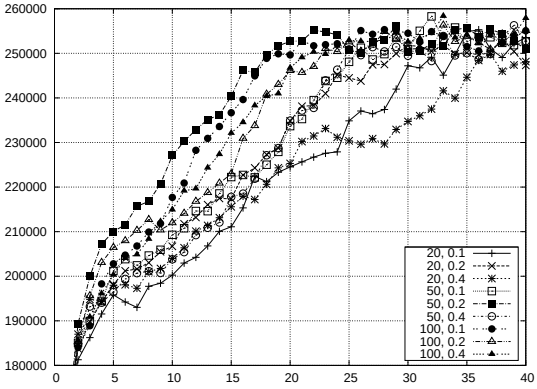


**Figure 4. Likelihood as function of the number of iterations for different values of $M_s$ and $M_i$**

Figure 4 depicts the value of the log-likelihood function for different values of $M_s$ and $M_i$. The fastest convergence is achieved with $M_s = 50$ and $M_i = 0.2$. In general, for fixed $M_s$, there is a tradeoff between performing the improvement of the traces with too high or too low value of $M_i$. Considering, for example, the case when $M_s = 50$, the fastest convergence is obtained with $M_i = 0.2$ while both $M_i = 0.1$ and $M_i = 0.4$ result in slower convergence. For what concerns instead $M_s$, higher values result in faster convergence apart from some random effects caused by the Monte Carlo simulation. According to our experience, when the parameters are close to the original ones higher value of $M_s$ results in lower oscillation of the likelihood.

Figure 5 illustrates how the algorithm proceeds with $M_s = 50$ and $M_i = 0.2$. The plots depict the result of the *E-step* after different number of iterations. We have plotted between every two consecutive samples the trace that arrives closest to the samples (result of Step 1. of the *E-step*) and its improved version (result of Step 2. of the *E-step*). Let us consider the second plot (top right) which depicts the traces after a single iteration. At $t_1 = 0.0$ the state is $Y_1 = 2000, Y_2 = 2000$ and our aim in the *E-step* is to create the trace that arrives close to the samples at $t_2 = 0.1$ which are $Y_1 = 499, Y_2 = 2230$. Applying Step 1 of the *E-step* between $t_1 = 0.0$ and $t_2 = 0.1$ results in a trace for which $Y_1 = 4, Y_2 = 2238$ at time $t_2 = 0.1$, Step 2 is able to improve that in such a way that the trace arrives to $Y_1 = 408, Y_2 = 2003$. After 4 iterations of the *E-* and *M-steps* (bottom left plot) the generated traces starts to behave similarly to the original model. After 30 iterations (bottom right plot) Step 2. of the *E-step* does not lead to visually recognisable improvements, i.e. the parameters are already such that out of only $M_s = 50$ simulation traces we can find one that arrives very close to the samples. The parameters after 30 iterations are $\bar{c}_{30} = |10.0\ 0.0101\ 9.85|$.

## 5. Conclusions and future work

In this paper, we presented an EM method to perform maximum likelihood estimation of the kinetic rate constants of a stochastic reaction network based on a small set of discrete observations. The applicability of the method was illustrated by applying it to the classical Lotka reactions.

In the future we will investigate the possibility of incorporating faster simulation methods (see, e.g., [9, 5, 13, 11, 3]) into the presented procedure, applying ad-hoc state-of-the-art optimisation methods in the *M-step*, finding more efficient ways of improving the subtraces resulting from simulation. Also, we plan to study the role of the parameters, $M_s$ and $M_i$, in order to tune them during the parameter estimation.

## References

[1] J. Booth and J. Hobert. Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *J. Royal Statistical Society, Series B (Methodological)*, 61:265–285, 1999.

[2] R. J. Boys, D. J. Wilkinson, and T. B. L. Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model,. *Statistics and Computing*, 2008. To appear.

[3] K. Burrage, T. Tian, and P. Burrage. A multi-scaled approach for simulating chemical reaction systems. *Progress in Biophysics and Molecular Biology*, 85(2–3):217–234, 2004.

[4] R. Burrows, G. Warnes, and R. Choudary Hanumara. Statistical modeling of biochemical pathways. Technical Re-

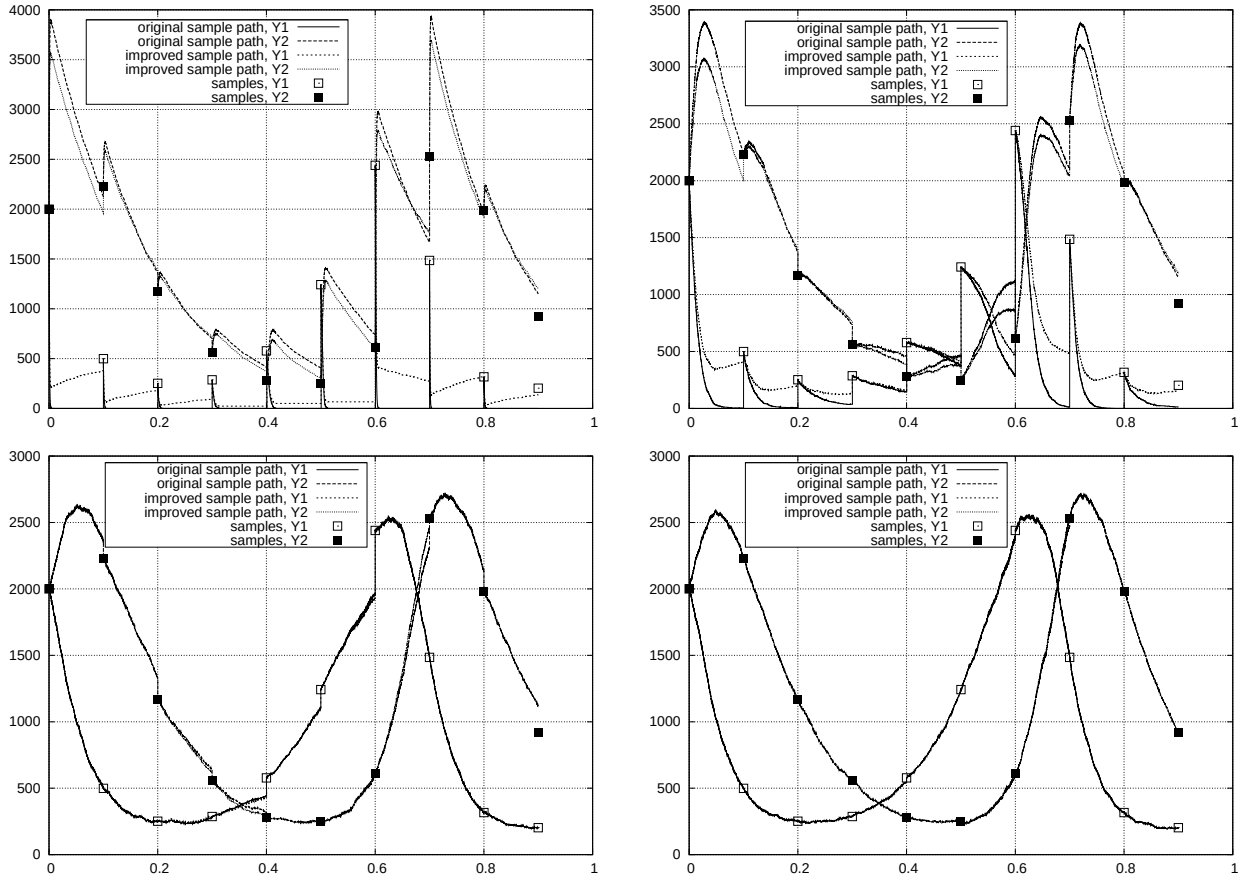


**Figure 5. Traces and improved traces with starting values (top left) and after 1 (top right), 4 (bottom left), and 30 (bottom right) iterations**

port 06/11, Dept. of Biostatistics and Computational Biology, University of Rochester, 2006.

[5] A. Chatterjee, K. Mayawala, J. S. Edwards, and D. G. Vlachos. Time accelerated Monte Carlo simulations of biological networks using the binomial $\tau$-leap method. *Bioinformatics*, 21(9):2136–2137, 2005.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society, Series B (Methodological)*, 39(1):1–38, 1977.

[7] K. G. Gadkar, R. Gunawan, and F. J. Doyle 3rd. Iterative approach to model identification of biological networks. *BMC Bioinformatics*, 6, 2005.

[8] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81(25):2340–2361, 1977.

[9] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically eacting systems. *J. Chem. Phys.*, 115(25):1716–1733, 2001.

[10] A. Golightly and D. Wilkinson. Bayesian inference for stochastic kinetic models using a diffusion approximation. *Biometrics*, 61:781–788, 2005.

[11] E. Haseltine and J. Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys.*, 117:6959–6969, 2002.

[12] C. Moles, P. Mendes, and J. Banga. Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome Res.*, 13:2467–2474, 2003.

[13] C. Rao and A. Arkin. Stochastic chemical kinetics and the quasi-steady-state assumption: application to the gillespie algorithm. *J. Chem. Phys.*, 118:4999–5010, 2003.

[14] S. Reinker, R. Altman, and J. Timmer. Parameter estimation in stochastic chemical reactions. *IEEE Proceedings Systems Biology*, 153:168–178, 2006.

[15] M. Sugimoto, S. Kikuchi, and M. Tomita. Reverse engineering of biochemical equations from time-course data by means of genetic programming. *Biosystems*, 80(2):155–164, 2005.

[16] T. Tian, S. Xu, J. Gao, and K. Burrage. Simulated maximum likelihood method for estimating kinetic rates in gene expression. *Bioinformatics*, 23(1):84–91, 2007.

[17] G. Wei and M. Tanner. A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *J. American Statistical Association*, 85:699–704, 1990.