The Forward-Reverse Algorithm for Stochastic Reaction Networks

Christian Bayer[2], Alvaro Moraes[1], Raúl Tempone[1] and Pedro Vilanova[1]

[1] Computer, Electrical and Mathematical Sciences & Engineering Division
King Abdullah University of Science and Technology (KAUST), Saudi Arabia
[2] Weierstrass Institute for applied analysis and stochastics, Berlin, Germany

{alvaro.moraesgutierrez,raul.tempone,pedro.guerra}@kaust.edu.sa,
christian.bayer@wias-berlin.de

Abstract

In this work, we present an extension of the forward-reverse algorithm by Bayer and Schoenmakers [2] to the context of stochastic reaction networks (SRNs). We then apply this bridge-generation technique to the statistical inference problem of approximating the reaction coefficients based on discretely observed data. To this end, we introduce a two-phase iterative inference method in which we solve a set of deterministic optimization problems where the SRNs are replaced by the classical ODE rates; then, during the second phase, the Monte Carlo version of the EM algorithm is applied starting from the output of the previous phase. Starting from a set of over-dispersed seeds, the output of our two-phase method is a cluster of maximum likelihood estimates obtained by using convergence assessment techniques from the theory of Markov chain Monte Carlo.

Statement of the problem

Let X be a Stochastic Reaction Network (SRN)

$$X = (X_1, \dots, X_d) : [0,T] \times \Omega \to \mathbb{Z}_+^d$$

described by

- Finite number of possible reactions $\nu_j \in \mathbb{Z}^d$. $x \in \mathbb{Z}_+^d,\ x \to x + \nu_j$
- and propensity (jump intensity) functions, $a_j : \mathbb{R}^d \to \mathbb{R}^+$ such that

$$\mathrm{P}\left(X(t+\mathrm{d}t) = x + \nu_j \,\middle|\, X(t) = x\right) = a_j(x)\mathrm{d}t + o\left(\mathrm{d}t\right),$$

Typically, $X_k(t)$ is the population size at time t of the $k-th$ species in the chemical kinetics jargon.
Goal: Estimate the set of unknown reaction rates c_j, assuming that $a_j(x) = c_j\, g_j(x)$, where g_j are known functions of x, from a discretely observed data set $\mathcal{D}$, that is, a finite collection,

$$\mathcal{D} = ([s_k, t_k], x(s_k), x(t_k))_{k=1}^K,$$

such that, for each k, $I_k := [s_k, t_k]$ is the time interval determined by two consecutive observational points, s_k and t_k, where the states $x(s_k)$ and $x(t_k)$ have been observed, respectively.

The EM algorithm

The EM algorithm [3, 4, 5, 6] due its name to its two steps: Expectation and Maximization steps. It is an iterative algorithm that, given an initial guess and a stopping rule, provides an approximation for a local maximum or saddle point of the likelihood function, lik$(\theta \,|\, \mathcal{D})$. Given an initial guess $\theta^{(0)}$, the EM algorithm maps $\theta^{(p)}$ into $\theta^{(p+1)}$ by

1. Expectation step: $Q_{\theta^{(p)}}(\theta \,|\, \mathcal{D}) := \mathrm{E}_{\theta^{(p)}}\left[\log(\text{lik}^c(\theta \,|\, \mathcal{D}, \tilde{\mathcal{D}})) \,\middle|\, \mathcal{D}\right]$.
2. Maximization step: $\theta^{(p+1)} := \arg\max_\theta Q_{\theta^{(p)}}(\theta \,|\, \mathcal{D})$.

Here, $\mathrm{E}_{\theta^{(p)}}\left[\cdot \,\middle|\, \mathcal{D}\right]$, denotes the expectation associated to the distribution of $\tilde{\mathcal{D}}$, which depends on $\theta^{(p)}$, conditional to the data, $\mathcal{D}$.

The Monte Carlo EM: If we know how to sample a sequence of M independent variates $(\tilde{\mathcal{D}}_i)_{i=1}^M \sim \tilde{\mathcal{D}} \,|\, \mathcal{D}$, with parameter $\theta^{(p)}$, then, we can define the following Monte Carlo estimator of $Q_{\theta^{(p)}}(\theta \,|\, \mathcal{D})$,

$$\hat{Q}_{\theta^{(p)}}(\theta \,|\, \mathcal{D}) := \frac{1}{M}\sum_{i=1}^{M} \log(\text{lik}^c(\theta \,|\, \mathcal{D}, \tilde{\mathcal{D}}_i)).$$

MCEM algorithm based on SRN-bridges

In this work, we present a two-phase algorithm that approximates the Maximum Likelihood Estimator, $\hat{\theta}_{\text{MLE}}$, of the vector, $\theta := (c_1, c_2, \dots, c_J)$, using the collected data, $\mathcal{D}$.

The phase I starts at $\theta_I^{(0)}$ and provides $\theta_{II}^{(0)}$. In the phase II, we run a Monte Carlo EM stochastic sequence, $(\hat{\theta}_{II}^{(p)})_{p=1}^{+\infty}$ until a certain convergence criterion is fulfilled. See Figure 1 for a schematic representation of our two-phase approach.

$$\theta_I^{(0)} \to \theta_{II}^{(0)} \to \hat{\theta}_{II}^{(1)} \to \cdots\ \hat{\theta}_{II}^{(p)} \to \cdots \to \hat{\theta}$$

Figure 1: *The two-phase estimation process. In the first step, we obtain $\theta_{II}^{(0)}$ from $\theta_I^{(0)}$ by solving an optimization problem* (1)*). In the subsequent steps, we generate the stochastic sequence $(\theta_{II}^{(p)})_{p=1}^{+\infty}$ using Monte Carlo EM* (5)*.*

During the phase II, we intensively use a computationally efficient implementation of the SRN-bridge simulation algorithm for simulating the "missing data" that feeds the Monte Carlo EM algorithm. Our two-phase algorithm is named FREM as the acronym for Forward-Reverse Expectation Maximization.

Phase I: Using approximating ODEs

The main goal of Phase I is to address the key problem of finding a suitable initial point, $\theta_{II}^{(0)}$ for Phase II. The idea is to increase (in some cases dramatically) the number of SRN-bridges from the sampled forward-reverse trajectories for all time intervals.

Let us now describe Phase I. From the user-selected seed, $\theta_I^{(0)}$, we solve the following deterministic optimization problem using some appropriate numerical iterative method:

$$\theta_{II}^{(0)} := \arg\min_{\theta \geq 0} \sum_k w_k\, d\left(\tilde{Z}^{(f)}(t_k^*;\theta), \tilde{Z}^{(b)}(t_k^*;\theta)\right),\ \text{starting from } \theta_I^{(0)}. \tag{1}$$

Here $\tilde{Z}^{(f)}$ is the ODE approximation (2) in the interval $[s_k, t_k^*]$, to the SRN defined by the reaction channels, $((\nu_j, a_j))_{j=1}^J$, and the initial condition $x(s_k)$; and, $\tilde{Z}^{(r)}$, is the ODE approximation (2) in the interval $[t_k^*, t_k]$, to the SRN defined by the reaction channels, $((-\nu_j, \tilde{a}_j))_{j=1}^J$, and by the initial condition $x(t_k)$. Notice that $\tilde{a}_j(x)$ is defined as $a_j(x{-}\nu_j)$.

$$\begin{cases} dZ(t) = \nu a(Z(t))dt,\ t \in \mathbb{R}_+, \\ Z(0) \ = x_0, \end{cases} \tag{2}$$

where the j-column of the matrix ν is ν_j and a is a column vector with components a_j.

We define $\tilde{Z}^{(b)}(u,\theta) {:=} \tilde{Z}^{(r)}(t_k^*{+}t_k{-}u, \theta)$ for $u \in [t_k^*, t_k]$. Here $w_k {:=} (t_k{-}s_k)^{-1}$ and $d(\cdot,\cdot)$ is an appropriate distance in $\mathbb{R}^d$.

Phase II: The Monte Carlo EM

This phase implements the Monte Carlo EM Algorithm for SRNs. In our statistical estimation approach, the Monte Carlo EM Algorithm uses data (pseudodata) generated by those forward and backward simulated paths that result in SRN-bridges, either exact or approximate. This last notion is associated to the use of kernels. In Figure 2 we illustrate this idea.

Simulating Forward and Backward Paths

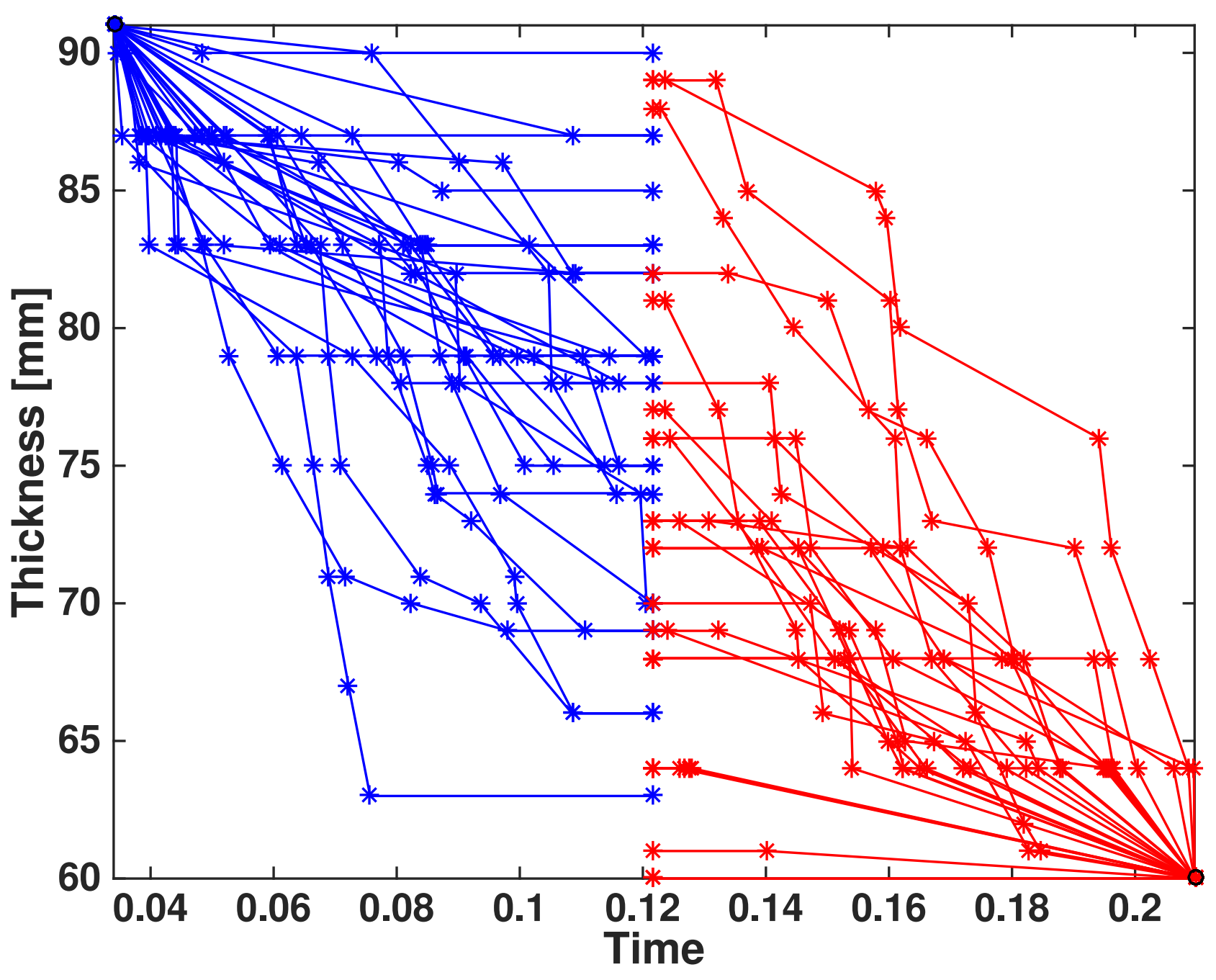

Figure 2: *Illustration of the forward reverse path simulation in Phase II.*

Given an estimation of the true parameter θ, say, $\hat{\theta} = (\hat{c}_1, \hat{c}_2, \dots, \hat{c}_J)$, the fist step is to simulate M_k forward paths with reaction channels $(\nu_j, \hat{c}_j g_j(x))_{j=1}^J$ in $[s_k, t_k^*]$, all of them starting at s_k from $x(s_k)$. Then, we simulate M_k backward paths with reaction channels $(-\nu_j, \hat{c}_j g_j(x-\nu_j))_{j=1}^J$ in $[t_k^*, t_k]$, all of them starting at t_k from $x(t_k)$. Let $(\tilde{X}^{(f)}(t_k^*, \tilde{\omega}_m))_{m=1}^{M_k}$ and $(\tilde{X}^{(b)}(t_k^*, \tilde{\omega}_{m'}))_{m'=1}^{M_k}$ denote the values of the simulated forward and backward paths at the time, t_k^*, respectively.

When the number of simulated paths, M_k, is large enough, and an appropriate guess of the parameter θ is used to generate those paths, then, due to the discrete nature of our state space, $\mathbb{Z}_+^d$, we expect to generate a number of *exact SRN-bridges* sufficiently large to perform statistical inference. However, at early stages of the Monte Carlo EM algorithm, our approximations to the unknown parameter, θ, are not expected to provide a large number of exact SRN-bridges. In such a case, we can use kernels to relax the notion of exact SRN-bridge.

Using the Epanechnikov Kernel

To make an efficient use of kernels, we first transform the endpoints of the forward and backward paths generated in the interval I_k,

$$\begin{aligned}\mathcal{X}_k := (&\tilde{X}^{(f)}(t_k^*, \tilde{\omega}_1), \tilde{X}^{(f)}(t_k^*, \tilde{\omega}_2), \dots, \tilde{X}^{(f)}(t_k^*, \tilde{\omega}_{M_k}), \\ &\tilde{X}^{(b)}(t_k^*, \tilde{\omega}_{M_k+1}), \tilde{X}^{(b)}(t_k^*, \tilde{\omega}_{M_k+2}), \dots, \tilde{X}^{(b)}(t_k^*, \tilde{\omega}_{2M_k}))\end{aligned}$$

into

$$\begin{aligned}H(\mathcal{X}_k) := (&\tilde{Y}^{(f)}(t_k^*, \tilde{\omega}_1), \tilde{Y}^{(f)}(t_k^*, \tilde{\omega}_2), \dots, \tilde{Y}^{(f)}(t_k^*, \tilde{\omega}_{M_k}), \\ &\tilde{Y}^{(b)}(t_k^*, \tilde{\omega}_{M_k+1}), \tilde{Y}^{(b)}(t_k^*, \tilde{\omega}_{M_k+2}), \dots, \tilde{Y}^{(b)}(t_k^*, \tilde{\omega}_{2M_k}))\end{aligned}$$

by a linear transformation, H, with the aim of eliminate possibly high correlations in the components of $\mathcal{X}_k$.

In our numerical examples, we use Epanechnikov's kernel:

$$\kappa(\eta) := \left(\frac{3}{4}\right)^d \prod_{i=1}^{d}(1-\eta_i^2)\mathbf{1}_{\{|\eta_i|\leq 1\}},$$

where η is defined as

$$\eta \equiv \eta_k(m, m') := \tilde{Y}^{(f)}(t_k^*, \tilde{\omega}_m) - \tilde{Y}^{(b)}(t_k^*, \tilde{\omega}_{m'}). \tag{3}$$

Kernel-weighted Averages for the Monte Carlo EM

Denote the number of times that the reaction ν_j occurred in the interval I by $R_{j,I}$, and define $F_{j,I} := \int_0^T g_j(X(s))\,ds$

The only available data in the interval I_k correspond to the observed values of the process, X, at its extremes. Therefore, the expected values, $\mathrm{E}_{\theta^{(p)}}\left[R_{j,I_k} \,\middle|\, \mathcal{D}\right]$ and $\mathrm{E}_{\theta^{(p)}}\left[F_{j,I_k} \,\middle|\, \mathcal{D}\right]$must be approximated by SRN-bridge simulation. To this end, we generate a set of M_k forward paths in the interval I_k using $\hat{\theta}_{II}^{(p)}$ as the current guess for the unknown parameter $\theta^{(p)}$. Having generated those paths, we record $R_{j,I_k}^{(f)}(\tilde{\omega}_m)$ and $F_{j,I_k}^{(f)}(\tilde{\omega}_m)$ for all $j = 1, 2, \dots, J$ and $m = 1, 2, \dots, M_k$. Analogously, we record $R_{j,I_k}^{(b)}(\tilde{\omega}_{m'})$ and $F_{j,I_k}^{(b)}(\tilde{\omega}_{m'})$ for all $j = 1, 2, \dots, J$ and $m' = 1, 2, \dots, M_k$.

Consider the following κ-weighted averages that approximate $\mathrm{E}_{\theta^{(p)}}\left[R_{j,I_k} \,\middle|\, \mathcal{D}\right]$ and $\mathrm{E}_{\theta^{(p)}}\left[F_{j,I_k} \,\middle|\, \mathcal{D}\right]$, respectively:

$$\begin{aligned}\mathcal{A}_{\hat{\theta}_{II}^{(p)}}(R_{j,I_k} \,|\, \mathcal{D}; \kappa) &:= \frac{\sum_{m,m'}\left(R_{j,I_k}^{(f)}(\tilde{\omega}_m) + R_{j,I_k}^{(b)}(\tilde{\omega}_{m'})\right)\kappa(\eta_k(m,m'))\psi_k(m')}{\sum_{m,m'}\kappa(\eta_k(m,m'))\psi_k(m')} \\ \mathcal{A}_{\hat{\theta}_{II}^{(p)}}(F_{j,I_k} \,|\, \mathcal{D}; \kappa) &:= \frac{\sum_{m,m'}\left(F_{j,I_k}^{(f)}(\tilde{\omega}_m) + F_{j,I_k}^{(b)}(\tilde{\omega}_{m'})\right)\kappa(\eta_k(m,m'))\psi_k(m')}{\sum_{m,m'}\kappa(\eta_k(m,m'))\psi_k(m')}\end{aligned} \tag{4}$$

where $\eta_\kappa(m, m')$ has been defined in (3) and $m, m' = 1, 2, \dots, M_k$, and $\psi_k(m') := \exp\left(\int_{t_k^*}^{t_k} c_j(\tilde{X}^{(b)}(s, \tilde{\omega}_{m'}))ds\right)$. Observe that we generate M_k forward and reverse paths in the interval I_k, but we do not control directly the number of exact or approximate SRN-bridges that are formed. The number M_k is chosen such that either the number of SRN-bridges is of order $\mathcal{O}\left(M_k\right)$ or we reach a computational budget M_b, which is 200 in our numerical experiments. In [1] we indicate an algorithm to reduce the the computational complexity of computing those κ-weighted averages from $\mathcal{O}\left(M_k^2\right)$ to $\mathcal{O}\left(M_k\right)$.

Finally, the Monte Carlo EM algorithm for this particular problem generates a stochastic sequence $(\hat{\theta}_{II}^{(p)})_{p=1}^{+\infty}$ staring from the initial guess $\theta_{II}^{(0)}$ provided by the phase I (1), and evolving by

$$\hat{c}^{(p+1)} = \frac{\sum_{k=1}^K \mathcal{A}_{\hat{\theta}_{II}^{(p)}}(R_{j,I_k} \,|\, \mathcal{D}; \kappa)}{\sum_{k=1}^K \mathcal{A}_{\hat{\theta}_{II}^{(p)}}(F_{j,I_k} \,|\, \mathcal{D}; \kappa)}, \tag{5}$$

where $\hat{\theta}_{II}^{(p)} = \left(\hat{c}_1^{(p)}, \dots, \hat{c}_J^{(p)}\right)$. A stopping criterion based on techniques widely used in Monte Carlo Markov chains is applied.

A numerical example

Birth-death process

This model has one species and two reaction channels:

$$\emptyset \xrightarrow{c_1} X,\ \ X \xrightarrow{c_2} \emptyset$$

described respectively by the stoichiometric matrix and the propensity function

$$\nu = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \text{ and } a(X) = \begin{pmatrix} c_1 \\ c_2 X \end{pmatrix}.$$

Since we are not continuously observing the paths of X, an increment of size k in the number of particles in a time interval $[t_1, t_2]$, may be the consequence of any combination of $n{+}k$ firings of channel 1 and n firings of channel 2 in that interval. This fact turns non-trivial the estimation of c_1 and c_2.

Data

Set $X_0{=}17$, $T{=}200$ and consider a synthetic data observed in regular time intervals of size $\Delta t{=}5$. This give us a set of 41 observations generated form a single path, using the parameters $c_1{=}1$ and $c_2{=}0.06$. The data trajectory is shown in Figure 3.

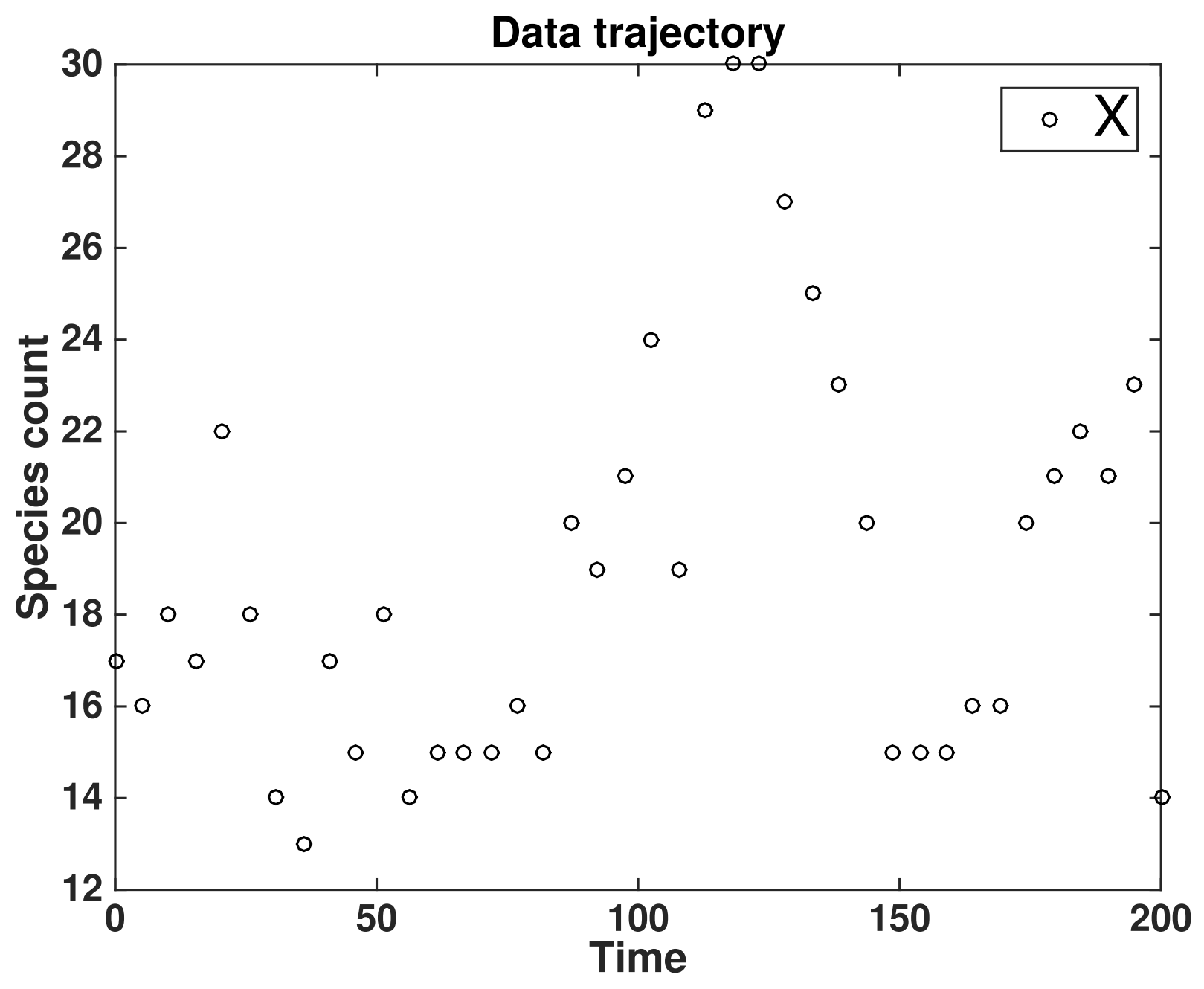

Figure 3: *Data trajectory for the Birth-death example. This is obtained by observing the values of an SSA path at regular time intervals of size $\Delta t{=}5$.*

Results

For this example we ran $N{=}4$ FREM sequences starting at $\theta_{I,1}^{(0)}{=}(0.5, 0.04)$, $\theta_{I,2}^{(0)}{=}(0.5, 0.08)$, $\theta_{I,3}^{(0)}{=}(1.5, 0.04)$ and $\theta_{I,4}^{(0)}{=}(1.5, 0.08)$. Those points where chosen after a previous exploration with the phase I.

Our FREM estimation gave us a cluster average of $\hat{\theta}{=}(1.22, 0.065)$. The FREM algorithm took $p^*{=}95$ iterations to converge. Details can be found in Table 1 and Figure 4.

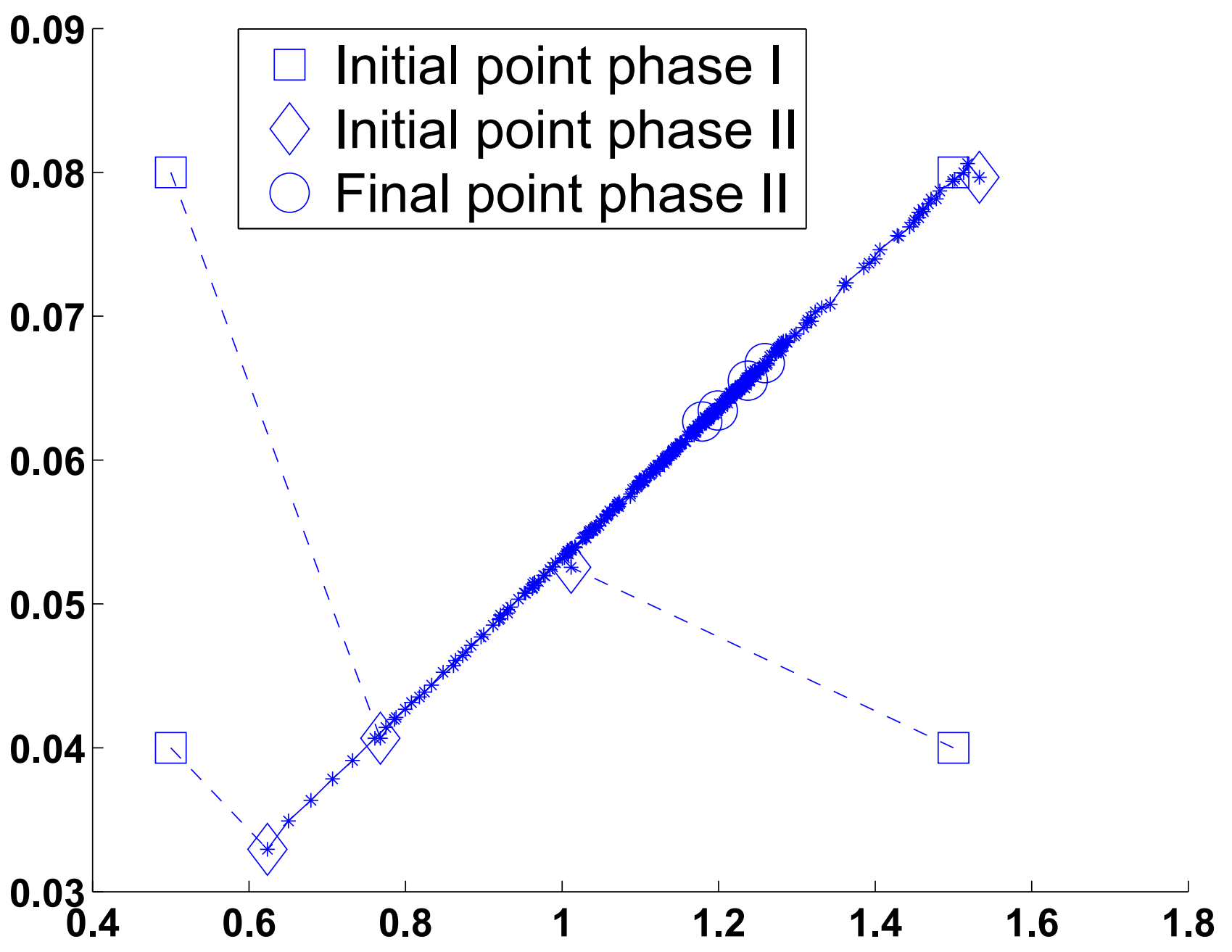

Figure 4: *FREM estimation (phase I and phase II) for the birth-death process.*

i	$\square = \theta_{I,i}^{(0)}$	$\Diamond = \theta_{II,i}^{(0)}$	$\bigcirc = \hat{\theta}_{II,i}^{(p^*)}$
1	(0.5, 0.04)	(6.24e-01, 3.29e-02)	(1.24e+00, 6.55e-02)
2	(0.5, 0.08)	(7.68e-01, 4.07e-02)	(1.29e+00, 6.67e-02)
3	(1.5, 0.04)	(1.01e+00, 5.25e-02)	(1.18e+00 6.27e-02)
4	(1.5, 0.08)	(1.53e+00, 7.97e-02)	(1.20e+00, 6.34e-02)

Table 1: *Values computed by the FREM Algorithm.*

Conclusions

We extended the forward-reverse technique developed by Bayer and Schoenmakers in [2] for stochastic reaction networks. We apply this technique in the statistical problem of inferring the set of coefficient of the propensity functions of stochastic reaction networks. We present an efficient two-phase algorithm in which the first phase is deterministic and it is intended to provide a starting point for the second phase which is the Monte Carlo EM Algorithm.

Acknowledgements

A. Moraes, R. Tempone and P. Vilanova are members of the KAUST SRI Center for Uncertainty Quantification in Computational Science and Engineering at King Abdullah University of Science and Technology (KAUST).

References

[1] C. Bayer, A. Moraes, R. Tempone, and P. Vilanova. The forward-reverse algorithm for stochastic reaction networks with applications to statistical inference. *submitted*, 2014.

[2] C. Bayer and J. Schoenmakers. Simulation of forward-reverse stochastic representations for conditional diffusions. *Annals of Applied Probability*, 24(5):1994–2032, October 2014.

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

[4] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. Wiley-Interscience, 2 edition, 3 2008.

[5] C. Robert and G. Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer, 2nd edition, 2005.

[6] M. Watanabe and K. Yamaguchi. *The EM Algorithm and Related Statistical Models*. Marcel Dekker Inc, 10 2003.