Weierstraß-Institut
für Angewandte Analysis und Stochastik
Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 0946 – 8633

Forward-reverse EM algorithm for Markov chains

Christian Bayer[1], Hilmar Mai[1], John Schoenmakers[1]

submitted: December 19, 2013

[1] Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
E-Mail: christian.bayer@wias-berlin.de
hilmar.mai@wias-berlin.de
john.schoenmakers@wias-berlin.de

No. 1907
Berlin 2013

2010 *Mathematics Subject Classification.* 65C05, 65J20.

Key words and phrases. Forward-reverse representations, EM algorithm, Monte Carlo simulation, maximum likelihood estimation, Markov chain estimation.

Partially supported by the DFG Research Center Matheon “Mathematics for Key Technologies” in Berlin.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract. We develop an EM algorithm for estimating parameters that determine the dynamics of a discrete time Markov chain evolving through a certain measurable state space. As a key tool for the construction of the EM method we develop forward-reverse representations for Markov chains conditioned on a certain terminal state. These representations may be considered as an extension of the earlier work [1] on conditional diffusions. We present several experiments and consider the convergence of the new EM algorithm.

1. Introduction

The EM algorithm going back to the seminal paper [5] is a very general method for iterative computation of maximum likelihood estimates in the setting of incomplete data. The algorithm consists of an expectation step (E-step) followed by a maximization step (M-step) which led to the name EM algorithm. Due to its general applicability and relative simplicity it has nowadays found its way into a great number of applications. These include maximum likelihood estimates of hidden Markov models in [8], of non-linear time series models in [3] and full information item factor models in [10] to give just a very limited selection.

Despite the simplicity of the basic idea of the algorithm its implementation in more complex models can be rather challenging. The global maximization of the likelihood in the M-step has recently been addressed successfully (see e.g. [9] and [7]). On the other hand, when the expectation of the complete likelihood is not known in closed form only partial solutions have been given yet. One approach developed in [17] uses Monte Carlo approximations of the unknown expectation and was therefore named Monte Carlo EM (MCEM) algorithm. As an alternative procedure the stochastic approximation EM algorithm was suggested in [6].

In this paper we take a completely different route by using a forward-reverse algorithm (cf. [1]) to approximate the conditional expectation of the complete data likelihood. In this respect we extend the idea from [1] to a Markov chain setting, which is considered an interesting contribution on its own. Indeed, Markov chains are more general in a sense since any diffusion monitored at discrete times yields canonically a Markov chain, but not every chain can be embedded (straightforwardly) into some continuous time diffusion the other way around.

The central issue is the identification of a parametric Markov chain model $(X_n,\, n = 0, 1, \ldots)$ based on data, i.e. realizations of the model, given on a typically course grid of time points, let us say $n_1, n_2, \ldots n_N$. Let us assume that the chain runs through $\mathbb{R}^d$ and that the transition densities $p^\theta_{n,m}(x,y)$, $n \geq m$, of the chain exist (with $p^\theta_{n,n}(x,y) := \delta_x(y)$), where the unknown parameter θ has to be determined. Then the standard method of maximum likelihood estimation would suggest to evaluate

$$\arg\max_\theta \sum_{i=0}^{N-1} \ln p^\theta_{n_i,n_{i+1}}(X_{n_i}, X_{n_{i+1}}) \quad \text{with } X_{n_0} = x_0 \text{ being the initial state of the chain.} \tag{1.1}$$

The problem with this approach is that usually only the one-step transition densities $p^\theta_{n,n+1}(x,y)$ are explicitly known, while any multi-step density $p^\theta_{n,m}(x,y)$ for $m > n$ can be expressed as an $m-n-1$ fold integral of one-step densities. In particular for larger $m-n$, these multiple integrals are numerically intractable however. We therefore consider the alternative problem

$$\arg\max_\theta \sum_{i=0}^{N-1} \sum_{j=n_i}^{n_{i+1}-1} \mathbb{E} \ln p^\theta_{j,j+1}(X_j, X_{j+1}), \tag{1.2}$$

in terms of the "missing data" $X_{n_i+1}, ..., X_{n_{i+1}-1}$, $i = 0, ..., N-1$. As such, between two such consecutive time points, n_i and n_{i+1} say, the chain may be considered as a bridge process

starting in realization X_{n_i} and ending up in realization $X_{n_{i+1}}$ (under the unknown parameter θ though), and so each term in (1.2) may be considered as an expected functional of the "bridged" Markov chain starting at time n_i in (data point) X_{n_i}, conditional on reaching (data point) $X_{n_{i+1}}$ at time n_{i+1}. We will therefore develop firstly an algorithm for estimating the terms in (1.2) for a given parameter θ. This algorithm will be of forward-reverse type in the spirit of the one in [1] developed for diffusion bridges. It should be noted here that in the last years the problem of simulating diffusion bridges has attracted much attention. Without pretending to be complete, see for example, [2, 4, 13, 15, 16, 14]. Having the forward-reverse algorithm at hand, we may construct an approximate solution to (1.2) in a sequential way by the so called EM algorithm: Once a generic approximation $\widehat{\theta}^{(m)}$ is constructed after m steps, one estimates

$$\widehat{\theta}^{(m+1)} := \arg\max_{\theta} \sum_{i=0}^{N-1} \sum_{j=n_i}^{n_{i+1}-1} \widehat{\mathbb{E}} \ln p^{\theta}_{j,j+1}(X^{\widehat{\theta}^{(m)}}_j, X^{\widehat{\theta}^{(m)}}_{j+1}),$$

where $X^{\widehat{\theta}^{(m)}}$ denotes the Markov bridge process under the transition law due to parameter $\widehat{\theta}^{(m)}$ and each term

$$\widehat{\mathbb{E}} \ln p^{\theta}_{j,j+1}(X^{\widehat{\theta}^{(m)}}_j, X^{\widehat{\theta}^{(m)}}_{j+1})$$

represents a forward-reverse estimation of

$$\mathbb{E} \ln p^{\theta}_{j,j+1}(X^{\widehat{\theta}^{(m)}}_j, X^{\widehat{\theta}^{(m)}}_{j+1})$$

as a (known) function of θ.

The structure of the paper is as follows. In Section 2 we recapitulate and adapt for our purposes the concept of reversed Markov chains, initially developed in [12] using the ideas in [11] on reversed diffusions. A general stochastic representation for expected functionals of conditional Markov chains is constructed in Section 3. This representation allows for a forward reverse EM simulation algorithm that is introduced and analyzed in Section 4. The section is concluded with finite sample study on simulated data from a practically relevant Ornstein-Uhlenbeck example.

2. Recap of forward and reverse representations for Markov chains

Consider a discrete-time Markov process $(X_n, \mathcal{F}_n),\ n = 0, 1, 2, ...$, on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with phase space $(S, \mathcal{S})$, henceforth called Markov chain. In general we assume that S is locally compact and that $\mathcal{S}$ is the Borel σ-algebra on S. For example, $S = \mathbb{R}^d$ or a proper subset of $\mathbb{R}^d$. Let $P_n,\ n \geq 0$, denote the one-step transition probabilities defined by

$$P_n(x, B) := \mathbb{P}(X_{n+1} \in B \mid X_n = x),\ n = 0, 1, 2, ...,\ \ x \in S,\ B \in \mathcal{S}. \tag{2.1}$$

In the case of an autonomous Markov chain all the one-step transition probabilities coincide and are equal to $P := P_0 = P_1 = \cdots$.

Let $X^{n,x}_m$, $m \geq n$, be a trajectory of the Markov chain which is at step n in the point x, i.e., $X^{n,x}_n = x$. The multi-step transition probabilities $P_{n,m}$ are then defined by

$$P_{n,m}(x, B) := \mathbb{P}(X^{n,x}_m \in B),\ \ x \in S,\ \ B \in \mathcal{S},\ \ m \geq n.$$

Due to these definitions, $P_{n,n}(x, B) = \delta_x(B) = 1_B(x)$ (Dirac measure), $P_n = P_{n,n+1}$, and the Chapman - Kolmogorov equation has the following form:

$$P_{n,m}(x, B) = \int P_{n,k}(x, dy) P_{k,m}(y, B), \quad x \in S,\ B \in \mathcal{S},\ n \leq k \leq m. \tag{2.2}$$

Let us fix $N > 0$ and consider for $0 \le n \le N$ the function

$$u_n(x) := \int P_{n,N}(x, dy) f(y) = \mathbb{E}\, f(X_N^{n,x}), \tag{2.3}$$

where f is $\mathcal{S}$-measurable and such that the mathematical expectation in (2.3) exists; for example, f is bounded. By the Markov property we have for $0 \le n < N$:

$$\begin{aligned} u_n(x) &= \mathbb{E}\, f(X_N^{n,x}) = \mathbb{E}\, f(X_N^{n+1,X_{n+1}^{n,x}}) \\ &= \mathbb{E}\, \mathbb{E}^{\mathcal{F}_{n+1}} f(X_N^{n+1,X_{n+1}^{n,x}}) = \mathbb{E}\, \mathbb{E}^{X_{n+1}^{n,x}} f(X_N^{n+1,X_{n+1}^{n,x}}) \\ &= \mathbb{E}\, u_{n+1}(X_{n+1}^{n,x}) = \int u_{n+1}(y) P_n(x, dy). \end{aligned}$$

Thus, $u_n(x)$ satisfies the following discrete integral Cauchy problem

$$u_n(x) = \int u_{n+1}(y) P_n(x, dy), \ n < N, \tag{2.4}$$

$$u_N(x) = f(x), \tag{2.5}$$

and (2.3) is a forward probabilistic representation of its solution. In fact, the probabilistic representation (2.3) can be used for simulating the solution of (2.4)-(2.5) by Monte Carlo. For our purpose, reverse probabilistic representations we need a somewhat more general version of the above result.

Theorem 2.1. *Let P_n be the one-step transition density of a Markov chain X as in (2.1) and let the function $f : S \to \mathbb{R}$ be measurable and bounded. Let further $\varphi_n : S \times S \to \mathbb{R}$ be a measurable and bounded functions for $n = 0, 1, 2, ...$ Then, the solution of the problem*

$$w_n(x) = \int w_{n+1}(z) \varphi_n(x, z) P_n(x, dz), \quad n < N, \tag{2.6}$$

$$w_N(x) = f(x) \tag{2.7}$$

has the following probabilistic representation:

$$w_n(x) = \mathbb{E}\left[f(X_N^{n,x}) \mathcal{X}_N^{n,x,1} \right], \tag{2.8}$$

where $(X, \mathcal{X})$ is an extended Markov chain in which $\mathcal{X}$ is governed by the equations

$$\mathcal{X}_{k+1}^{n,x,\gamma} = \mathcal{X}_k^{n,x,\gamma} \varphi_k(X_k^{n,x}, X_{k+1}^{n,x}), \qquad \mathcal{X}_n^{n,x,\gamma} = \gamma,$$

where $n \le k < N$.

Proof. Note that $\mathcal{X}_k^{n,x,\gamma} = \gamma \mathcal{X}_k^{n,x,1}$. Thus, for $n < N$, (2.8) may be written as

$$\begin{aligned} w_n(x) &= \mathbb{E}\left[f(X_N^{n+1,X_{n+1}^{n,x}}) \mathcal{X}_N^{n+1,X_{n+1}^{n,x},\mathcal{X}_{n+1}^{n,x,1}} \right] \\ &= \mathbb{E}\, \mathcal{X}_{n+1}^{n,x,1} \mathbb{E}^{(X_{n+1}^{n,x}, \mathcal{X}_{n+1}^{n,x,1})} \left[f(X_N^{n+1,X_{n+1}^{n,x}}) \mathcal{X}_N^{n+1,X_{n+1}^{n,x},1} \right] \\ &= \mathbb{E}\, \left[\mathcal{X}_{n+1}^{n,x,1} w_{n+1}(X_{n+1}^{n,x}) \right] \\ &= \mathbb{E}\, \left[\varphi_n(x, X_{n+1}^{n,x}) w_{n+1}(X_{n+1}^{n,x}) \right] \\ &= \int w_{n+1}(z) \varphi_n(x, z) P_n(x, dz), \end{aligned}$$

and (2.7) is trivially fulfilled for $n = N$. □

2.1. **Reverse probabilistic representations.** We henceforth take $(S,\mathcal{S}) = \left(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d)\right)$ and assume that the transition probabilities $P_{n,m}(x,dy)$ have densities $p_{n,m}(x,y)$ with respect to the Lebesgue measure on $(S,\mathcal{S})$. We note however that without any problem one may consider more general state spaces equipped with some reference measure, and transition probabilities absolutely continuous to with respect to it. The representation (2.3) can thus be written in the form

$$I(f) := \mathbb{E}\, f(X_N^{n,x}) = \int p_{n,N}(x,y) f(y) dy, \quad 0 \le n \le N. \tag{2.9}$$

Let the initial value ξ of the chain X at moment n be random with density $g(x)$. Consider the functional

$$I(g,f) = \int\int g(x) p_{n,N}(x,y) f(y) dx dy = \mathbb{E} f(X_N^{n,\xi}). \tag{2.10}$$

Formally, by taking for g a δ-function we obtain (2.9) again, and by taking f to be a δ-function we obtain the integral

$$J(g) := \int g(x) p_{n,N}(x,y) dx. \tag{2.11}$$

We now propose suitable (reverse) probabilistic representations for $J(g)$, where g is an arbitrary test function (not necessarily a density). For this we are going to construct a class of reverse Markov chains that allow for a probabilistic representation for the solution of (2.11).

Let us fix a number $N \in \mathbb{N}$ and consider for $0 \le m < N$, functions $\psi_m : S \times S \to \mathbb{R}_+$ such that for each m and y the function

$$q_m(y,\cdot) := \frac{p_{N-m-1}(\cdot, y)}{\psi_m(y,\cdot)}, \quad 0 \le m < N, \tag{2.12}$$

is a density on S. For example, one could take ψ_m independent of the second argument, and then obviously

$$\psi_m(y) = \int p_{N-m-1}(z,y) dz. \tag{2.13}$$

We now introduce a "reverse" processes $(Y_m^y, \mathcal{Y}_m^y)_{0\le m\le N}$ by the system

$$\begin{aligned} &\mathbb{P}(\, Y_{m+1}^y \in dz' \big|\ Y_m^y = z) = q_m(z,z')dz', \\ &\mathcal{Y}_{m+1}^y = \mathcal{Y}_m^y \psi_m(Y_m^y, Y_{m+1}^y), \\ &Y_0^y := Y_0^{0,y} := y, \quad \mathcal{Y}_0^y := \mathcal{Y}_0^{0,y,1} := 1, \quad 0 \le m < N, \end{aligned} \tag{2.14}$$

hence Y^y is governed by the one-step transition probabilities $Q_m(z,dz') := q_m(z,z')dz'$ (i.e. Q_m instead of P_m).

Theorem 2.2. *For any n, $0 \le n \le N$, (2.11) has the following probabilistic representation.*

$$\int g(x) p_{n,N}(x,y) dx = \mathbb{E}\left[g(Y_{N-n}^y) \mathcal{Y}_{N-n}^y \right],$$

where g is an arbitrary test function (a "density" $p_{m,m}$ has to be interpreted as a Dirac distribution or δ-function).

Proof. From the Chapman - Kolmogorov equation (2.2) we obtain straightforwardly the Chapman-Kolmogorov equation for densities,

$$p_{n,m}(x,y) = \int p_{n,k}(x,z) p_{k,m}(z,y) dz, \quad x,y \in S,\ n \le k \le m. \tag{2.15}$$

Let us now fix n, $n < N$ (for $n = N$ the statement is trivial) also, and introduce the functions

$$v_k(y) := \int g(x)p_{n,k}(x,y)dx,\ n \le k \le N. \tag{2.16}$$

From (2.15) we get

$$\begin{aligned} v_k(y) &= \int v_{k-1}(z)p_{k-1}(z,y)dz,\ n < k \le N, \\ v_n(y) &= g(y), \end{aligned} \tag{2.17}$$

where $p_{k-1} := p_{k-1,k}$ denote the one-step densities. For $n < k \le N$ we now consider a "reversed" time variable $m = N + n - k$ and write with $\widetilde{v}_m(y) := v_{N+n-m}(y)$ and (2.12) system (2.17) in the form

$$\begin{aligned} \widetilde{v}_m(y) &= \int \widetilde{v}_{m+1}(z)\psi_{m-n}(y,z)q_{m-n}(y,z)dz, \quad n \le m < N, \\ \widetilde{v}_N(y) &= g(y). \end{aligned} \tag{2.18}$$

Let us write (2.18) in a slightly different form,

$$\begin{aligned} \widetilde{v}_m(y) &= \int \widetilde{v}_{m+1}(z)\psi^{(n)}_m(y,z)q^{(n)}_m(y,z)dz, \quad n \le m < N, \\ \widetilde{v}_N(y) &= g(y) \end{aligned}$$

with $\psi^{(n)}_m := \psi_{m-n}$ and $q^{(n)}_m := q_{m-n}$. Via Theorem 2.1 we next obtain a probabilistic representation of the form (2.8) for the solution of problem (2.18), hence (2.11) or $J(g)$. Indeed, by taking in Theorem 2.1 instead of X a Markov chain $\left(Y^{(n),y}_m\right)_{n\le m\le N}$, where $Y^{(n),y}$ is governed by the one-step transition probabilities $Q^{(n)}_m(z,dz') := q^{(n)}_m(z,z')dz'$, $n \le m < N$, with initial condition $Y^{(n),y}_n = y$, and constructing $\left(\mathcal{Y}^{(n),y}_m\right)_{n\le m\le N}$ according to

$$\mathcal{Y}^{(n),y}_{m+1} = \mathcal{Y}^{(n),y}_m \psi^{(n)}_m(Y^{(n),y}_m, Y^{(n),y}_{m+1}), \qquad \mathcal{Y}^{(n),y}_n = 1, \qquad n \le m < N, \tag{2.19}$$

it follows by Theorem 2.1 that

$$J(g) = \widetilde{v}_n(y) = v_N(y) = \mathbb{E}\left[g(Y^{(n),y}_N)\mathcal{Y}^{(n),y}_N\right]. \tag{2.20}$$

It remains to see that

$$\mathbb{E}\left[g(Y^{(n),y}_N)\mathcal{Y}^{(n),y}_N\right] = \mathbb{E}\left[g(Y^{y}_{N-n})\mathcal{Y}^{y}_{N-n}\right]$$

which follows from the fact that initial values and the one step transition probabilities of the processes

$$\left(Y^{(n),y}_{n+i}, \mathcal{Y}^{(n),y}_{n+i}\right)_{i=0,\dots,N-n} \quad \text{and} \quad \left(Y^{y}_{i}, \mathcal{Y}^{y}_{i}\right)_{i=0,\dots,N-n}$$

coincide. □

It should be stressed that, in contrast to a corresponding theorem in [12], Theorem 2.2 provides a family of probabilistic representations indexed by $n = 1, \dots, N$, that involves only one common reverse process Y^y. In Theorem 2.2 N was fixed but, when different N are in play, we will denote them by $Y^{y;N}$. It turns out that this extension of the related result in [12] is crucial for deriving probabilistic representations for conditional Markov chains below (cf. [1]).

3. Simulation of conditional diffusions via forward-reverse representations

In this section we describe for a Markov Chain (2.1) an efficient procedure for estimating the final distributions of a chain $X = (X_n)_{n=0,\dots,N}$ conditioned, or pinned, on a terminal state X_N. More specifically, for some given (unconditional) diffusion process X we aim at simulation of the functional

$$\mathbb{E}\Big[\, g(X_{m_1}, \dots, X_{m_r}) \Big|\; X_N \in A,\, X_0 = x \Big], \tag{3.1}$$

where $0 \le m_1 < m_2 < \cdots < m_r < N$ (hence $r < N$), A is some set that may consist of only one point, and g is an arbitrarily given suitable test function, and $x \in \mathbb{R}^d$ is a given state. The procedure proposed below is in fact a discrete-time version of the method developed in [1] for continuous-time processes given by an Ito SDE. Thus, let us consider the problem (3.1) for fixed $x, y \in \mathbb{R}^d$ (i.e. $A = \{y\}$). We firstly state the following central theorem.

Theorem 3.1. *Given a grid $\mathcal{D}_l := \{0 \le n^* < n_1 < \cdots < n_l =: N\}$, it holds that*

$$\mathbb{E}\Big[f(Y^{y;n_l}_{n_l-n_0}, Y^{y;n_l}_{n_l-n_1}, \dots, Y^{y;n_l}_{n_l-n_{l-1}}) \boldsymbol{\mathcal{Y}}^{y;n_l}_{n_l-n_0} \Big] = \int_{\mathbb{R}^{d\times L}} f(y_0, y_1, \dots, y_{l-1}) \prod_{i=1}^{l} p_{n_{i-1},n_i}(y_{i-1}, y_i) dy_{i-1}$$

with $y_l := y$ and $n_0 := n^*$.

Proof. Without loss of generality, we assume in this proof that the grid satisfies $n_i - n_{i-1} = 1$, $i = 1, \dots, l$. Indeed, extend $f : \mathbb{R}^{d\times l} \to \mathbb{R}$ to a function $\tilde{f} : \mathbb{R}^{d\times(N-n^*)} \to \mathbb{R}$ such that

$$\tilde{f}\Big(Y^{y;N}_{N-n^*}, Y^{y;N}_{N-n^*-1}, \dots, Y^{y;N}_{2}, Y^{y;N}_{1}\Big) = f\Big(Y^{y;N}_{n_l-n_0}, Y^{y;N}_{n_l-n_1}, \dots, Y^{y;N}_{n_l-n_{l-1}}\Big).$$

Then, re-expressing the transition densities p_{n_{i-1},n_i} in terms of the one-step transition densities p_i using Chapman-Kolmogorov, we see that the statement of the theorem is equivalent to

$$\mathbb{E}\Big[\tilde{f}\Big(Y^{y;N}_{N-n^*}, Y^{y;N}_{N-n^*-1}, \dots, Y^{y;N}_{1}\Big) \boldsymbol{\mathcal{Y}}^{y;N}_{N-n^*} \Big] = \int_{\mathbb{R}^{d\times(N-n^*)}} \tilde{f}\,(y_{n^*}, \dots, y_{N-1}) \prod_{i=n^*+1}^{N} p_{i-1}(y_{i-1}, y_i) dy_{i-1} \tag{3.2}$$

with $y_N \equiv y$. In fact, we shall prove that

$$\mathbb{E}\Big[f_p\Big(Y^{y;N}_{p}, \dots, Y^{y;N}_{1}\Big) \boldsymbol{\mathcal{Y}}^{y;N}_{p} \Big] = \int f_p(y_{N-p}, \dots, y_{N-1}) \prod_{i=N-p+1}^{N} p_{i-1}(y_{i-1}, y_i) dy_{i-1} \tag{3.3}$$

for any $1 \le p \le N - n^*$ for any (e.g., bounded measurable) function $f_p : \mathbb{R}^{d\times p} \to \mathbb{R}$. (3.3) gives the formula from the statement of the theorem for $p = N - n^*$ with f_{N-n^*} being the function $\tilde{f}$ from above. We prove (3.3) by induction on p. For $p = 1$, this boils down to Theorem 2.2 with $n = N - 1$.

For the step from $p-1$ to p, we note that by definition

$$\boldsymbol{\mathcal{Y}}^{y;N}_{p} = \boldsymbol{\mathcal{Y}}^{y;N}_{p-1} \psi_{p-1}\Big(Y^{y;N}_{p-1}, Y^{y;N}_{p}\Big),$$

with $\psi_{p-1}(y,\cdot) q_{p-1}(y,\cdot) = p_{N-(p-1)-1}(\cdot, y) = p_{N-p}(\cdot, y)$ by (2.12). Hence, we have

$$\begin{aligned}\mathbb{E}\Big[f_p(Y^{y;N}_{p}, Y^{y;N}_{p-1}, \dots, Y^{y;N}_{1}) \boldsymbol{\mathcal{Y}}^{y;N}_{p} \Big] &= \mathbb{E}\Big[f_p(Y^{y;N}_{p}, Y^{y;N}_{p-1}, \dots, Y^{y;N}_{1}) \boldsymbol{\mathcal{Y}}^{y;N}_{p-1} \psi_{p-1}(Y^{y;N}_{p-1}, Y^{y;N}_{p}) \Big] \\ &= \mathbb{E}\Big[g\Big(Y^{y;N}_{p-1}, \dots, Y^{y;N}_{1}\Big) \boldsymbol{\mathcal{Y}}^{y;N}_{p-1} \Big],\end{aligned}$$

with

$$g(z_{p-1},\dots,z_1) \equiv \mathbb{E}\Big[f_p\left(Y_p^{y;N}, Y_{p-1}^{y;N},\dots,Y_1^{y;N}\right)\psi_{p-1}(Y_{p-1}^{y;N}, Y_p^{y;N})\Big|\, Y_{p-1}^{y;N}=z_{p-1},\dots,Y_1^{y;N}=z_1\Big]$$
$$= \int f_p(z,z_{p-1},\dots,z_1)p_{N-p}(z,z_{p-1})dz.$$

Applying the induction hypothesis for $f_{p-1}=g$, we obtain

$$\mathbb{E}\left[f_p(Y_p^{y;N}, Y_{p-1}^{y;N},\dots,Y_1^{y;N})\mathcal{Y}_p^{y;N}\right] = \mathbb{E}\left[g\left(Y_{p-1}^{y;N},\dots,Y_1^{y;N}\right)\mathcal{Y}_{p-1}^{y;N}\right]$$
$$= \int g(y_{N-p+1},\dots,y_{N-1}) \prod_{i=N-p+2}^{N} p_{i-1}(y_{i-1},y_i)dy_{i-1}$$
$$= \int f_p(y_{N-p},\dots,y_{N-1}) \prod_{i=N-p+1}^{N} p_{i-1}(y_{i-1},y_i)dy_{i-1}.$$

□

Following the lines of [1], we now consider an extended integer sequence

$$0 < m_1 < \cdots < m_k = n^* = n_0 < n_1 < \cdots < n_l = N,$$

and a kernel K_ϵ of the form

$$K_\epsilon(u) := \epsilon^{-d}K(u/\epsilon), \quad y\in\mathbb{R}^d,$$

with K being integrable on $\mathbb{R}^d$ and $\int_{\mathbb{R}^d} K(u)du = 1$. Formally K_ϵ converges to the delta function δ_0 on $\mathbb{R}^d$ (in distribution sense) as $\epsilon\downarrow 0$. We then have the following stochastic representation for (3.1) with $n_i = m_{k+i}$, $i=0,...,l=r-k+1$.

Theorem 3.2. *Let the chain $(Y,\mathcal{Y}) := \left(Y^{y;N},\mathcal{Y}^{y;N}\right)$ be given by (2.14), and the modified integer sequence $(\widehat{n}_\cdot)$ be defined by*

$$\widehat{n}_i := n_l - n_{l-i}, \quad i=1,\dots,l. \tag{3.4}$$

It then holds

$$\mathbb{E}\Big[\, g(X_{m_1},\dots,X_{m_r})\Big|\; X_{m_0}=x,\; X_N=y\Big]$$
$$= \mathbb{E}\Big[\, g(X_{m_1},\dots,X_{m_{k-1}},X_{n^*}^{0,x},X_{n_1},\dots,X_{n_{l-1}})\Big|\; X_{m_0}=x,\; X_N=y\Big]$$
$$= \lim_{\epsilon\to 0}\frac{\mathbb{E}\left[g\left(X_{m_1}^{m_0,x},\dots,X_{m_{k-1}}^{m_0,x},X_{n^*}^{m_0,x},Y_{\widehat{n}_{l-1}}^{y;N},\dots,Y_{\widehat{n}_1}^{y;N}\right)K_\epsilon\left(Y_{\widehat{n}_l}^{y;N}-X_{n^*}^{m_0,x}\right)\mathcal{Y}_{\widehat{n}_l}^{y;N}\right]}{\mathbb{E}\left[K_\epsilon\left(Y_{\widehat{n}_l}^{y;N}-X_{n^*}^{m_0,x}\right)\mathcal{Y}_{\widehat{n}_l}^{y;N}\right]}. \tag{3.5}$$

Proof. The proof is completely analogue to the corresponding one in [1]. As a rough sketch, apply Theorem 3.1 to

$$f(X_{m_1}^{0,x},\dots,X_{n^*}^{0,x},y_0,y_1,\dots,y_{l-1}) := g(X_{m_1}^{0,x},\dots,X_{n^*}^{0,x},y_1,\dots,y_{l-1})K_\epsilon(y_0-X_{n^*}^{0,x}),$$

conditional on $X_{m_1}^{0,x},\dots,X_{n^*}^{0,x}$, send $\epsilon\to 0$, and divide the result by

$$p_{0,N}(x,y) = \lim_{\epsilon\to 0}\mathbb{E}\left[K_\epsilon\left(Y_{\widehat{n}_l}-X_{n^*}\right)\mathcal{Y}_{\widehat{n}_l}\right]..$$

□

3.0.1. *Forward-Reverse estimator.* Given Theorem 3.2 the corresponding forward-reverse Monte Carlo estimator for (3.5) suggests itself: Sample i.i.d. copies $X^{0,x,(1)}, ..., X^{0,x,(M)}$ of the process $X^{0,x}$ and, independently, i.i.d. copies $\left(Y^{y;N,(1)}, \mathcal{Y}^{y;N,(\widetilde{M})}\right), ..., \left(Y^{y;N,(1)}, \mathcal{Y}^{y;N,(\widetilde{M})}\right)$ of the process $\left(Y^{y;N}, \mathcal{Y}^{y;N}\right)$. Take for K a second order kernel, take for simplicity $M = \widetilde{M}$, and choose a bandwith $\epsilon_M \sim M^{-1/d}$ if $d \leq 4$, or $\epsilon_M \sim M^{-2/(4+d)}$ if $d \geq 4$. By next replacing the expectations in the numerator and denominator of (3.5) by their respective Monte Carlo estimates involving double sums, one ends up with an estimator with Root-Mean-Square error $O(M^{-1/2})$ in the case $d \leq 4$ and $O(M^{-4/(4+d)})$ in the case $d > 4$ (cf. [1] for details).

4. Implementation of the forward-reverse-EM algorithm

Before presenting two concrete numerical examples, we will first discuss general aspects of the implementation of the forward-reverse EM algorithm. For this purpose, let us, for simplicity, assume that the Markov chains X and $(Y, \mathcal{Y})$ are time-homogeneous, i.e., that $p \equiv p_k$ and $q \equiv q_k$ do not depend on time k. We assume that we observe the Markov process X at times $0 = i_0 < \cdots < i_r = N$, i.e., our data consist of the values $X_{i_k} = x_{i_k}$, $k = 0, \ldots, r$. For later use, we introduce the shortcut-notation $\mathbf{x} := (x_{i_j})_{j=0}^r$.

The law of X depends on an s-dimensional parameter $\theta \in \mathbb{R}^s$, which we are trying to estimate, i.e., $p = p^\theta$. To this end, let

$$\ell(\theta; x_0, \ldots, x_N) := \sum_{i=1}^N \log p^\theta(x_{i-1}, x_i)$$

denote the log-likelihood function for the estimation problem assuming full observation. We make the structural assumption that there are (explicitly given) functions $g_i : \mathbb{R}^s \to \mathbb{R}$ and $h_i : \mathbb{R}^{N+1} \to \mathbb{R}$, $h_i^j : \mathbb{R}^{i_j - i_{j-1}+1} \to \mathbb{R}$ such that

$$\ell(\theta; x_0, \ldots, x_N) = \sum_{i=1}^m g_i(\theta) h_i(x_0, \ldots, x_N) = \sum_{i=1}^m g_i(\theta) \sum_{j=1}^r h_i^j(x_{i_{j-1}}, \ldots, x_{i_j}). \tag{4.1}$$

The structural assumption (4.1) allows us to effectively evaluate the conditional expectation of the log-likelihood ℓ_c for different parameters θ, without having to re-compute the conditional expectations. More precisely, recall that for a given guess $\widetilde{\theta}$ the E step of the EM algorithm consists in calculating the function

$$\theta \mapsto Q(\theta; \widetilde{\theta}, \mathbf{x}) := \mathbb{E}_{\widetilde{\theta}}\Big[\ell_c(\theta; X_0, \ldots, X_N)) | X_{i_j} = x_{i_j},\ j = 0, \ldots, r \Big], \tag{4.2}$$

with $\mathbb{E}_{\widetilde{\theta}}$ denoting (conditional) expectation under the parameter $\widetilde{\theta}$. Inserting the structural assumption (4.1), we immediately obtain

$$Q(\theta; \widetilde{\theta}, \mathbf{x}) = \sum_{i=1}^m g_i(\theta) \mathbb{E}_{\widetilde{\theta}}\Big[h_i(X_0, \ldots, X_N) | X_{i_j} = x_{i_j},\ j = 0, \ldots, r \Big] = \sum_{i=1}^m g_i(\theta) z_i^{\widetilde{\theta}}$$

with $z_i^{\widetilde{\theta}} := \mathbb{E}_{\widetilde{\theta}}\Big[h_i(X_0, \ldots, X_N) | X_{i_j} = x_{i_j},\ j = 0, \ldots, r \Big]$, $i = 1, \ldots, m$. Note that the definition of $z_i^{\widetilde{\theta}}$ does not depend on the free parameter θ. Thus, only one (expensive) round of calculations of conditional expectations is needed for a given $\widetilde{\theta}$, producing a cheap-to-evaluate function in θ, which can then be fed into any maximization algorithm.

For any given $\widetilde{\theta}$, the calculation of the numbers $z_1^{\widetilde{\theta}}, \dots, z_m^{\widetilde{\theta}}$ requires running the forward-reverse algorithm for conditional expectations. More precisely, using the Markov property we decompose

$$z_i^{\widetilde{\theta}} := \mathbb{E}_{\widetilde{\theta}}\Big[h_i(X_0, \dots, X_N) | \, X_{i_j} = x_{i_j}, \ j = 0, \dots, r \Big] \\ = \sum_{j=1}^{r} \mathbb{E}_{\widetilde{\theta}}\Big[h_i^j(X_{i_{j-1}}, \dots, X_{i_j}) \Big| \, X_{i_{j-1}} = x_{i_{j-1}}, X_{i_j} = x_{i_j} \Big].$$

All these conditional expectations are of the Markov-bridge type for which the forward-reverse algorithm is designed. Hence, for each iteration of the EM algorithm, we apply the forward-reverse algorithm r times, one for the time-intervals $i_{j-1}, \dots, i_j$, $j = 1, \dots, r$, evaluating all the functionals $h_1^j, \dots, h_m^j$ at one go.

4.1. **Choosing the reverse process.** Recall the defining equation for the one-step transition density q of the reverse process given in (2.12). For simplicity, we shall again assume that the forward and the reverse processes are time-homogeneous, implying that (2.12) can be re-expressed as

$$q(y, z) = \frac{p(z, y)}{\psi(y, z)}.$$

Notice that in this equation only p is given a-priori, i.e., the user is free to choose any re-normalization ψ provided that for any $y \in \mathbb{R}^d$ the resulting function $z \mapsto q(y, z)$ is non-negative and integrates to 1. In particular, we can turn the equation around, choose *any* transition density q and *define*

$$\psi(y, z) := \frac{p(z, y)}{q(y, z)}.$$

Note, however, that for the resulting forward-reverse process square integrability of the process $\mathcal{Y}$ is desirable. More precisely, only square integrability of the (numerator of the) complete estimator corresponding to (3.5) (see 3.0.1) is required, but it seems far-fetched to hope for any cancelations giving square integrable estimators when $\mathcal{Y}$ itself is not square integrable. From a practical point of view, it therefore seems reasonable to aim for functions ψ satisfying

$$\psi \approx 1,$$

i.e., to try to find functions ψ which are bounded from above by a number slightly smaller than 1 and bounded from below by a number slightly smaller than 1. Indeed, note that $\mathcal{Y}$ is obtained by multiplying terms of the form $\psi(Y_n, Y_{n+1})$ along the whole trajectory of the reverse process Y. Hence, if ψ is bounded by a large constant, $\mathcal{Y}$ could easily take extremely large values, to the extent that buffer-overflow might occur in the numerical implementation – think of multiplying 100 numbers of order 100. On the other hand, if ψ is considerably smaller than 1, $\mathcal{Y}$ might take very small values, which can cause problem in particular taking into account the division by the forward-reverse estimator for the transition density in the denominator of the estimator 3.0.1.

Heuristically, the following procedure seems promising.

- If $y \mapsto \int_{\mathbb{R}^d} p(z, y) dz$ can be computed in closed form (or so fast that one can think of a closed formula), then choose

$$\psi(y) := \psi(y, z) = \int_{\mathbb{R}^d} p(z, y) dz.$$

- Otherwise, assume that we can find a non-negative (measurable) function $\widetilde{p}(z,y)$ with closed form expression for $\int_{\mathbb{R}^d} \widetilde{p}(z,y)dz$ such that $p(z,y) \approx \widetilde{p}(z,y)$. Then define

$$q(y,z) := \frac{\widetilde{p}(z,y)}{\int_{\mathbb{R}^d} \widetilde{p}(z,y)dz},$$

which is a density in z. By construction, we have

$$\psi(y,z) = \frac{p(z,y)}{q(y,z)} = \int_{\mathbb{R}^d} \widetilde{p}(z,y)dz\, \frac{p(z,y)}{\widetilde{p}(z,y)},$$

implying that we are (almost) back in the first situation.

Remark 4.1. Even if we can, indeed, explicitly compute $\psi(y,z) = \int_{\mathbb{R}^d} p(z,y)dz$, there is no guarantee that $\mathcal{Y}$ has (non-exploding) finite second moments. However, in practice, this case seems to be much easier to control and analyze.

4.2. **Complexity of the forward-reverse algorithm.** We end this general discussion of the forward-reverse EM algorithm by a refined analysis of the complexity of the forward-reverse algorithm for conditional expectations as compared to [1].

Theorem 4.2. *Assume that the transition densities p and q have full support in $\mathbb{R}^d$.*[1] *Moreover, assume that the kernel K is supported in a ball of radius $R > 0$. Then the forward-reverse algorithm for N forward and reverse trajectory based on a bandwidth proportional to $N^{-1/d}$ can be implemented in such a way that its cost is $O(N \log N)$ as $N \to \infty$.*

Proof. In order to increase the clarity of the argument, we re-write the double sum in the forward-reverse algorithm 3.0.1 to a simpler form, which highlights the computational issues. Indeed, we are trying to compute a double sum of the form

$$\sum_{i=1}^{N} \sum_{j=1}^{N} F_{i,j} K_\epsilon \left(X^i_{n^*} - Y^j_{\hat{n}_l} \right), \tag{4.3}$$

where $F_{i,j}$ obviously depends on the whole ith sample of the forward process X and on the whole jth sample of the reverse process $(Y, \mathcal{Y})$.

We may assume that the end points $X^i_{n^*}$ and $Y^j_{\hat{n}_l}$ of the N samples of the forward and reverse trajectories are contained in a compact set $[-L, L]^d$. (Indeed, the necessary re-scaling operation can obviously be done with $O(N)$ operations.) In fact, for ease of notation we shall assume that the points are actually contained in $[0,1]^d$. We sub-divide $[0,1]^d$ in boxes with side-length $S\epsilon$, where $S > R$ is chosen such that $1/(S\epsilon) \in \mathbb{N}$. Note that there are $K := (S\epsilon)^{-d}$ boxes which we order lexicographically and associate with the numbers $1, \dots, K$ accordingly.

In the next step, we shall order the points $X^i_{n^*}$ and $Y^j_{\hat{n}_l}$ into these boxes. First, let us define a function $f_1 : [0,1]^d \to \{1, \dots, 1/(S\epsilon)\}^d$ by setting

$$f_1(x) := (\lceil x_1/(S\epsilon) \rceil, \dots, \lceil x_d/(S\epsilon) \rceil),$$

with $\lceil \cdot \rceil$ denoting the smallest integer larger or equal than a number. Moreover, define $f_2 : \{1, \dots, 1/(S\epsilon)\}^d \to \{1, \dots, K\}$ by

$$f_2(i_1, \dots, i_d) := (i_1 - 1)(S\epsilon)^{-d+1} + (i_2 - 1)(S\epsilon)^{-d+2} + \cdots + (i_d - 1) + 1.$$

[1] Obviously, this assumption can be weakened.

Obviously, a point $x \in [0,1]^d$ is contained in the box number k if and only if $f_2(f_1(x)) = k$.[2] Now we apply a sorting algorithm like quick-sort to both sets of points $\left(X^1_{n^*}, \dots, X^N_{n^*}\right)$ and $\left(Y^1_{\hat{n}_l}, \dots, Y^N_{\hat{n}_l}\right)$ using the ordering relation defined on $[0,1]^d \times [0,1]^d$ by

$$x < y : \Longleftrightarrow \; f_2(f_1(x)) < f_2(f_1(y)).$$

Sorting both sets incurs a computational cost of $O(N \log N)$, so that we can now assume that the vectors $X^i_{n^*}$ and $Y^i_{\hat{n}_l}$ are ordered.

Notice that $K_\epsilon(x-y) \neq 0$ if and only if x and y are situated in neighboring boxes, i.e., if $|f_1(x) - f_1(y)|_\infty \leq 1$, where we define $|\alpha|_\infty := \max_{i=1,\dots,d} |\alpha_i|$ for multi-indices α. Moreover, there are 3^d such neighboring boxes, whose indices can be easily identified, in the sense that there is a simple set-valued function f_3 which maps an index k to the set of all the indices $f_3(k)$ of the 3^d neighboring boxes.[3] Moreover, for any $k \in \{1, \dots, K\}$ let $X^{i(k)}_{n^*}$ be the first element of the ordered sequence of $X^i_{n^*}$ lying in the box k. Likewise, let $Y^{j(k)}_{\hat{n}_l}$ be the first element in the ordered sequence $Y^j_{\hat{n}_l}$ lying in the box with index k. Note that identifying these $2K$ indices $i(1), \dots, i(K)$ and $j(1), \dots, j(K)$ can be achieved at computational costs of order $O(K \log N) = O(N \log N)$.

After all these preparations, we can finally express the double sum (4.3) as

$$\sum_{i=1}^{N} \sum_{j=1}^{N} F_{i,j} K_\epsilon \left(X^i_{n^*} - Y^j_{\hat{n}_l}\right) = \sum_{k=1}^{K} \sum_{r \in f_3(k)} \sum_{i=i(k)}^{i(k+1)-1} \sum_{j=j(r)}^{j(r+1)-1} F_{i,j} K_\epsilon \left(X^i_{n^*} - Y^j_{\hat{n}_l}\right). \tag{4.4}$$

Regarding the computational complexity of the right hand side, note that

$$\begin{aligned} K &= O(N), \\ |f_3(k)| &\leq 3^d, \\ i(k+1) - i(k) &\sim (S\epsilon)^d N \sim S^d N^{-1} N = O(1) \text{ on average}, \\ j(r+1) - j(r) &\sim (S\epsilon)^d N \sim S^d N^{-1} N = O(1) \text{ on average}. \end{aligned}$$

Hence, after all the pre-computations of total cost $O(N \log N)$ the final summation (4.4) incurs a computational cost of order $O(N)$. □

Remark 4.3. As becomes apparent in the proof of Theorem 4.2, the constant in front of the asymptotic complexity bound does depend exponentially on the dimension d.

4.3. **A discrete Cox-Ingersoll-Ross example.** Consider the Markov chain given by

$$X_{n+1} = X_n + \lambda(\theta - X_n)\Delta t + \sigma |X_n|^\gamma \Delta W_{n+1}, \tag{4.5}$$

where Δt is fixed and ΔW_n are independent random variables distributed according to $\mathcal{N}(0, \Delta t)$. Moreover, we assume that $0 \leq \gamma$ is fixed and known. The other parameters σ, λ and θ are not considered known and need to be estimated. We are mainly interested in the case $\gamma = 1/2$, which corresponds to some kind of Euler discretization of the Cox-Ingersoll-Ross model from finance.

[2] To make this construction fully rigorous, we would have to make the boxes half-open and exclude the boundary of $[0,1]^d$.

[3] Strictly speaking, only those boxes which are not neighbors of the boundary of $[0,1]^d$ have 3^d neighbors.

For the forward-reverse algorithm, we next need to specify the reverse chain. In this case, we propose to take the following reverse chain:

$$Y_{n+1} = Y_n - \lambda\,(\theta - Y_n)\,\Delta t + \sigma\,|Y_n|^\gamma\,\Delta\widetilde{W}_{n+1}. \tag{4.6}$$

In order to get the dynamics of $\mathcal{Y}$, we need to derive the normalization function ψ between the one-step transition densities p of the forward and q of the reverse processes. (We suppress the indices as we are in a time-homogeneous situation.) For (4.5) together with (4.6) the one-step transition densities are normal densities in the forward variables,

$$p(x,y) = \frac{1}{\sqrt{2\pi\Delta t}\sigma\,|x|^\gamma}\exp\left(-\frac{(y - x - \lambda(\theta - x)\Delta t)^2}{2\sigma^2\,|x|^{2\gamma}\,\Delta t}\right),$$
$$q(y,z) = \frac{1}{\sqrt{2\pi\Delta t}\sigma\,|y|^\gamma}\exp\left(-\frac{(z - y + \lambda(\theta - y)\Delta t)^2}{2\sigma^2\,|y|^{2\gamma}\,\Delta t}\right).$$

Hence, we get

$$\psi(y,z) = \frac{p(z,y)}{q(y,z)} = \left|\frac{y}{z}\right|^\gamma \exp\left(-\frac{1}{2\sigma^2\Delta t}\left[\frac{(y - z - \lambda(\theta - z)\Delta t)^2}{|z|^{2\gamma}} - \frac{(z - y + \lambda(\theta - y)\Delta t)^2}{|y|^{2\gamma}}\right]\right). \tag{4.7}$$

Up to constant terms (in the un-known parameters σ, λ and θ), the log-likelihood function of a sequence of observations $\mathbf{x} = (x_0, \ldots, x_N)$ of the full path of the process X is given by

$$\begin{aligned}
\ell_c\,(\sigma,\lambda,\theta;\mathbf{x}) &= \log\left(\prod_{i=1}^N p(x_{i-1},x_i)\right) \\
&= -N\log\sigma - \frac{1}{2\sigma^2\Delta t}\sum_{i=1}^N \frac{(x_i - (1-\lambda\Delta t)x_{i-1} - \lambda\theta\Delta t)^2}{|x_{i-1}|^{2\gamma}} \\
&= -N\log\sigma - \frac{1}{2\sigma^2\Delta t}\sum_{i=1}^N\Bigg[\frac{x_i^2}{|x_{i-1}|^{2\gamma}} - 2(1-\lambda\Delta t)\frac{x_i x_{i-1}}{|x_{i-1}|^{2\gamma}} \\
&\quad - 2\lambda\theta\Delta t\frac{x_i}{|x_{i-1}|^{2\gamma}} + (1-\lambda\Delta t)^2\frac{x_{i-1}^2}{|x_{i-1}|^{2\gamma}} \\
&\quad + 2\lambda\theta\Delta t(1-\lambda\Delta t)\frac{x_{i-1}}{|x_{i-1}|^{2\gamma}} + \lambda^2\theta^2\Delta t^2\frac{1}{|x_{i-1}|^{2\gamma}}\Bigg].
\end{aligned}$$

Again, assume that we actually observe $x_{i_0}, \ldots, x_{i_r}$ with $i_0 = 0 < \cdots < i_r = N$, while the remaining points x_j, $j \notin \{i_0, \ldots, i_r\}$, are assumed to be unobserved. Define random variables $Z_0 := N$ and

$$Z_1 := \sum_{i=1}^N \frac{X_i^2}{|X_{i-1}|^{2\gamma}}, \qquad Z_2 := \sum_{i=1}^N \frac{X_i}{|X_{i-1}|^{2\gamma}},$$
$$Z_3 := \sum_{i=1}^N \frac{X_{i-1}X_i}{|X_{i-1}|^{2\gamma}}, \qquad Z_4 := \sum_{i=1}^N \frac{1}{|X_{i-1}|^{2\gamma}},$$
$$Z_5 := \sum_{i=1}^N \frac{X_{i-1}}{|X_{i-1}|^{2\gamma}}, \qquad Z_6 := \sum_{i=1}^N \frac{X_{i-1}^2}{|X_{i-1}|^{2\gamma}}.$$

Hence, we have with $\mathbf{X} = (X_0, \ldots, X_N)$

$$\begin{aligned}
\ell_c\,(\sigma,\lambda,\theta;\mathbf{X}) = -Z_0\log\sigma - \frac{1}{2\sigma^2\Delta t}\big[&Z_1 - 2\lambda\theta\Delta t Z_2 - 2(1-\lambda\Delta t)Z_3 \\
&+ \lambda^2\theta^2\Delta t^2 Z_4 + 2\lambda\theta\Delta t(1-\lambda\Delta t)Z_5(1-\lambda\Delta t)^2 Z_6\big].
\end{aligned}$$

Then we do the E-step. Given guesses $\sigma^n, \lambda^n, \theta^n$ for the parameters, let

$$z_i := \mathbb{E}_{\sigma^n,\lambda^n,\theta^n}\left[Z_i | X_{i_0} = x_{i_0}, \dots, X_{i_r} = x_{i_r}\right], \quad i = 0, \dots, 7,$$

and observe that

$$\begin{aligned} Q(\sigma, \lambda, \theta; \sigma^n, \lambda^n, \theta^n; x_{i_0}, \dots, x_{i_r}) &:= \mathbb{E}_{\sigma^n,\lambda^n,\theta^n}\left[\ell_c(\sigma, \lambda, \theta; \mathbf{X}) | X_{i_0} = x_{i_0}, \dots, X_{i_r} = x_{i_r}\right] \\ &= -z_0 \log \sigma - \frac{1}{2\sigma^2\Delta t}[z_1 - 2\lambda\theta\Delta t z_2 - 2(1 - \lambda\Delta t)z_3 \\ &\quad + \lambda^2\theta^2\Delta t^2 z_4 + 2\lambda\theta\Delta t(1 - \lambda\Delta t)z_5(1 - \lambda\Delta t)^2 z_6]. \end{aligned}$$

Remark 4.4. Note that in particular z_4 will only be defined when $\gamma < 1/2$, which we assume from now on. Indeed, unlike in the continuous time CIR model, there are no possible parameter regimes preventing the process to become negative with positive probability. Of course, there are other possible discretizations of the CIR model which lead to Markov chains which do not have the problem of non-integrability of the log-likelihood function. For instance, we can consider the logarithm of the CIR process (assuming the Feller condition to hold), compute its dynamics by Ito's formula and then discretise the resulting stochastic differential equation.

The first order conditions for finding the maximum of $(\sigma, \lambda, \theta) \mapsto Q(\sigma, \lambda, \theta; \sigma^n, \lambda^n, \theta^n; x_{i_0}, \dots, x_{i_r})$ are given by

$$\begin{aligned} \partial_\sigma Q &= -\frac{z_0}{\sigma} + \frac{1}{\sigma^3\Delta t}[z_1 - 2\lambda\theta\Delta t z_2 - 2(1 - \lambda\Delta t)z_3 \\ &\quad + \lambda^2\theta^2\Delta t^2 z_4 + 2\lambda\theta\Delta t(1 - \lambda\Delta t)z_5(1 - \lambda\Delta t)^2 z_6], \\ \partial_\lambda Q &= -\frac{1}{2\sigma^2\Delta t}[-2\theta\Delta t z_2 + 2\Delta t z_3 + 2\lambda\theta^2\Delta t^2 z_4 \\ &\quad + 2\theta\Delta t(1 - 2\lambda\Delta t)z_5 - 2\Delta t(1 - \lambda\Delta t)z_6], \\ \partial_\theta Q &= -\frac{\lambda}{2\sigma^2\Delta t}[-2\Delta t z_2 + 2\lambda\theta\Delta t^2 z_4 + 2\Delta t(1 - \lambda\Delta t)z_5], \end{aligned}$$

and we obtain the maximizers given by

$$\begin{aligned} \sigma^2 &= \frac{z_3^2 z_4 - 2z_2 z_3 z_5 + z_1 z_5^2 + z_3^2 z_6 - z_1 z_4 z_6}{\Delta t z_0\left(z_5^2 - z_4 z_6\right)}, \\ \lambda &= \frac{z_3 z_4 - z_2 z_5 + z_5^2 - z_4 z_6}{\Delta t\left(z_5^2 - z_4 z_6\right)}, \\ \theta &= \frac{z_3 z_5 - z_2 z_6}{z_3 z_4 - z_2 z_5 + z_5^2 - z_4 z_6}. \end{aligned}$$

4.4. **Simulation example: Ornstein-Uhlenbeck dynamics.** In this section we apply the forward-reverse EM algorithm to simulated data from a discretized Ornstein-Uhlenbeck process, that is obtained from (4.5) by setting $\sigma = 1$ and $\theta = \gamma = 0$. The corresponding Markov chain is thus given by

$$X_{n+1} = X_n + \lambda X_n \Delta t + \Delta W_{n+1}, \tag{4.8}$$

where W_n is as in Section 4.3. The drift parameter $\lambda \in \mathbb{R}$ is unknown and we will employ the forward reverse EM algorithm to estimate it from simulated data. The Ornstein-Uhlenbeck model has the advantage that the likelihood estimator is available in closed form and we can thus compare it to the results of the EM algorithm.

Δt	N	bandwidth	mean $\hat{\lambda}$	std dev $\hat{\lambda}$	likel.	std dev likel.
0.1	2000	0.0005	0.972	0.0135	-3.402	0.00290
	8000	0.000125	1.098	0.00841	-3.383	0.00062
	32000	3.125e-05	1.132	0.00476	-3.381	0.000123
	128000	7.812e-06	1.151	0.00236	-3.381	2.783e-05
	512000	1.953e-06	1.157	0.00117	-3.381	4.745e-06
	2048000	4.882e-07	1.159	0.000581	-3.381	1.005e-06
0.05	2000	0.0005	1.160	0.0141	-3.107	0.000854
	8000	0.000125	1.247	0.00872	-3.103	9.867e-05
	32000	3.125e-05	1.253	0.00468	-3.103	1.329e-05
	128000	7.812e-06	1.265	0.00225	-3.103	3.772e-06
	512000	1.953e-06	1.265	0.00111	-3.103	6.005e-07

TABLE 1. Behavior of the forward-reverse EM algorithm for a discretized Ornstein-Uhlenbeck model for different step sizes Δt, initial guess $\lambda = 0.5$ and true MLE $\hat{\lambda}_{\text{MLE}} = 1.161$ and 1.266

In each simulation run we suppose that we have known observations

$$X_0, X_{10\Delta t}, \ldots, X_{40\Delta t}$$

for varying step size Δt and use the EM methodology to approximate the likelihood function in between. We perform six iteration of the algorithm with increasing number of data points N.

In table 1 we summarize the results of two runs for the discrete Ornstein-Uhlenbeck chain. The mean and standard deviation are estimated from 1000 Monte Carlo iterations. We find that already after three steps the mean is very close to the corresponding estimate of the true MLE. This indicates a surprisingly fast convergence for this example. Note also that the approximated value of the likelihood function stabilizes extremely fast at the maximum.

Table 2 gives results for the same setup as in Table 1 but with initial guess $\lambda = 2$ such that the forward-reverse EM algorithm converges from above to the true maximum of the likelihood function. We observe that the smaller step size $\Delta t = 0.05$ results in a more accurate approximation of the likelihood and also of the true MLE. It seems that the step size has crucial influence on the convergence rate of the algorithm, since for $\Delta t = 0.05$ the likelihood stabilizes already from the second iteration.

In Figure 1 the empirical distribution of 1000 estimates for λ is plotted. The initial value was 0.5 and the true maximum of the likelihood function is at 1.161. The step size between observations was chosen to be $\Delta t = 0.1$. The histogram on the left shows the estimates after only one iteration and on the right the estimates were obtained from five iterations of the forward-reverse EM algorithm.

Figure 2 depicts the distribution of 1000 Monte Carlo samples of the likelihood values that led to the estimates in Figure 1. It is interesting to see that after one iteration of the algorithm the likelihood values are approximately bell shaped (left histogram) whereas after five iterations the distributions becomes more and more one-sided as would be expected, since the EM algorithm only increase the likelihood from step to step towards the maximum.

Figure 3 shows the convergence of the forward reverse EM algorithm when the number of iterations increases. We find that already after 4 iterations the estimate is very close to the true MLE for λ. After six iterations the algorithm has almost perfectly stabilized at the value of the true MLE $\lambda = 1.16$.

Δt	N	bandwidth	mean $\hat{\lambda}$	std dev $\hat{\lambda}$	likel.	std dev likel.
0.1	2000	0.0005	1.554	0.0353	-3.457	0.0134
	8000	0.000125	1.312	0.0127	-3.393	0.00221
	32000	3.125e-05	1.217	0.00544	-3.382	0.000351
	128000	7.812e-06	1.185	0.00245	-3.381	5.817e-05
	512000	1.953e-06	1.168	0.00121	-3.381	1.227e-05
0.05	2000	0.0005	1.390	0.0248	-3.108	0.00238
	8000	0.000125	1.289	0.00925	-3.103	0.000130
	32000	3.125e-05	1.261	0.00471	-3.103	1.451e-05
	128000	7.812e-06	1.266	0.00221	-3.103	2.538e-06
	512000	1.953e-06	1.266	0.00113	-3.103	5.855e-07

TABLE 2. Behavior of the forward-reverse EM algorithm for a discretized Ornstein-Uhlenbeck model for different step sizes Δt, initial guess $\lambda = 2$ and true MLE $\hat{\lambda}_{\text{MLE}} = 1.161$ and 1.266

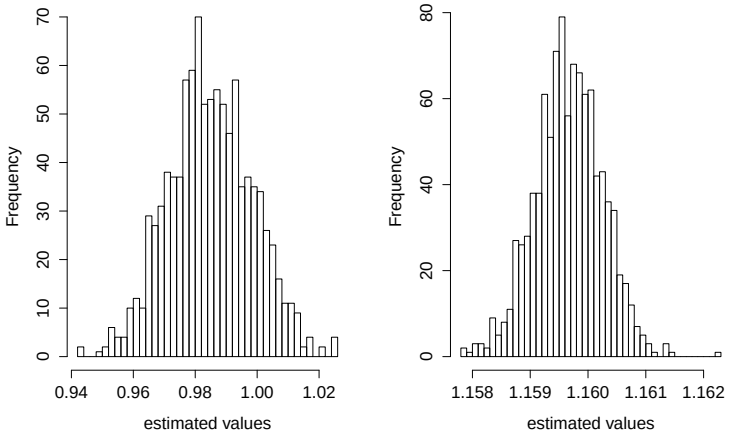

FIGURE 1. Empirical distribution of 1000 estimates after one iteration (right) and after five iteration (left) of the forward-reverse EM algorithm.

REFERENCES

[1] Christian Bayer and John Schoenmakers. Simulation of forward-reverse stochastic representations for conditional diffusions. *Ann. of Appl. Prob.* to appear, 2013.

[2] Mogens Bladt and Michael Sørensen. Simple simulation of diffusion bridges with application to likelihood inference for diffusions. Preprint, 2012.

[3] K.S. Chan and Johannes Ledolter. Monte Carlo EM estimation for time series models involving counts. *J. Am. Stat. Assoc.*, 90(429):242–252, 1995.

[4] B. Delyon and Y. Hu. Simulation of conditioned diffusion and application to parameter estimation. *Stochastic Process. Appl.*, 116(11):1660–1675, 2006.

[5] A.P. Dempster; N.M. Laird and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Discussion. *J. R. Stat. Soc., Ser. B*, 39:1–38, 1977.

[6] Bernard Delyon; Marc Lavielle and Eric Moulines. Convergence of a stochastic approximation version of the EM algorithm. *Ann. Stat.*, 27(1):94–128, 1999.

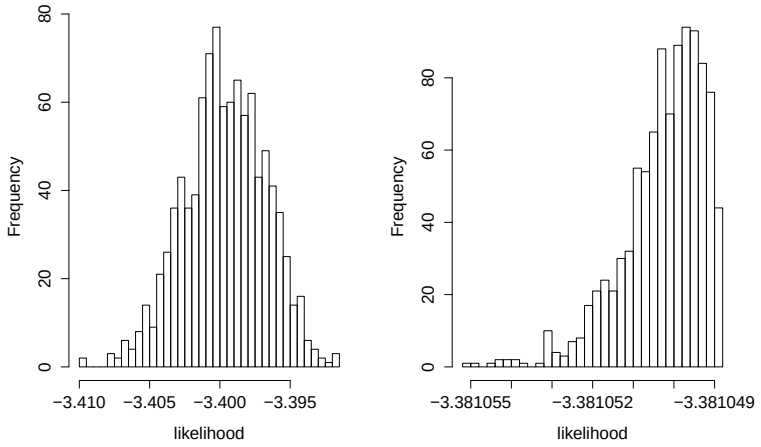

FIGURE 2. Empirical distribution of the likelihood values of 1000 Monte Carlo samples after one iteration (right) and after five iteration (left) of the forward-reverse EM algorithm.

[7] Chuanhai Liu and Donald B. Rubin. The ECME algorithm: A simple extension of EM and ECM with faster monotone convergence. *Biometrika*, 81(4):633–648, 1994.

[8] Iain L. MacDonald and Walter Zucchini. *Hidden Markov and other models for discrete-valued time series.* London: Chapman & Hall, 1997.

[9] Xiao-Li Meng and Donald B. Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

[10] Xiao-Li Meng and Stephen Schilling. Fitting full-information item factor models and an empirical investigation of bridge sampling. *J. Am. Stat. Assoc.*, 91(435):1254–1267, 1996.

[11] G. N. Milstein, J. Schoenmakers, and V. Spokoiny. Transition density estimation for stochastic differential equations via forward-reverse representations. *Bernoulli*, 10(2):281–312, 2004.

[12] G. N. Milstein, J. Schoenmakers, and V. Spokoiny. Forward and reverse representations for Markov chains. *Stochastic Process. Appl.*, 117(8):1052–1075, 2007.

[13] G. N. Milstein and M. V. Tretyakov. Evaluation of conditional Wiener integrals by numerical integration of stochastic differential equations. *J. Comput. Phys.*, 197(1):275–298, 2004.

[14] Moritz Schauer, Frank van der Meulen, and Harry van Zanten. Guided proposals for simulating multi-dimensional diffusion bridges. Preprint, 2013.

[15] Panos Stinis. Conditional path sampling for stochastic differential equations through drift relaxation. *Commun. Appl. Math. Comput. Sci.*, 6(1):63–78, 2011.

[16] Andrew M. Stuart, Jochen Voss, and Petter Wiberg. Fast communication conditional path sampling of SDEs and the Langevin MCMC method. *Commun. Math. Sci.*, 2(4):685–697, 2004.

[17] G. Wei and M. Tanner. A Monte Carlo Implementation of the EM Algorithm and the Poor Man's Data Augmentation Algorithm. *J. Am. Stat. Assoc.*, 85:699–704, 1990.

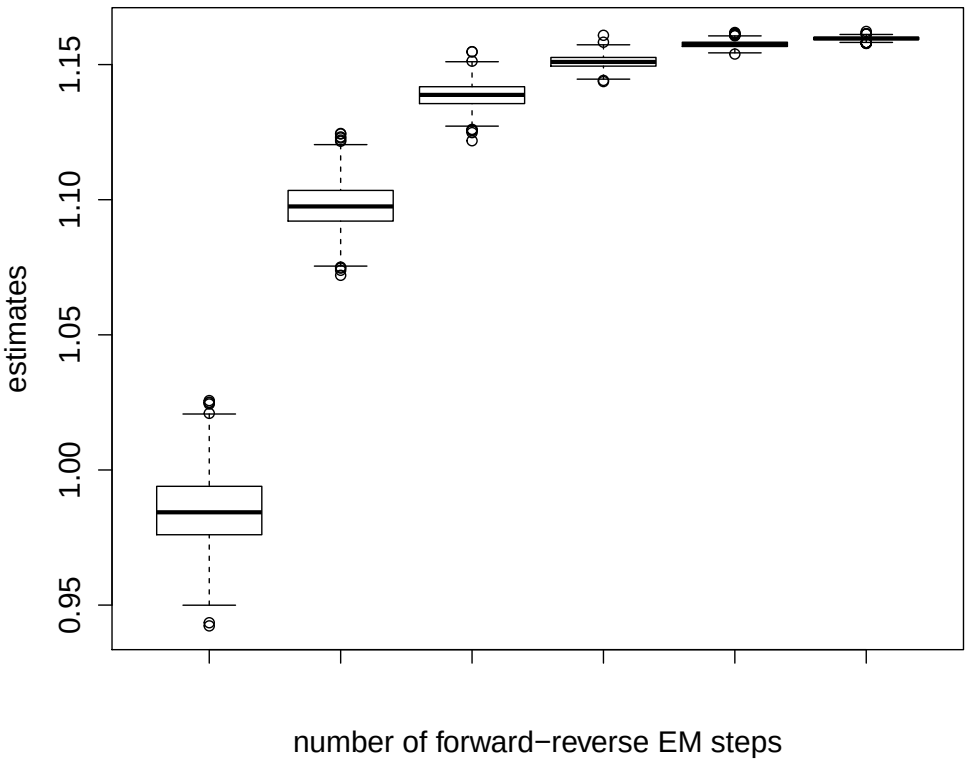

FIGURE 3. Convergence of the forward-reverse EM algorithm from one to six iterations for each 1000 estimates of λ. The value of the true MLE is $\hat{\lambda} = 1.161$.