Calibrating the exponential Ornstein–Uhlenbeck multiscale stochastic volatility model

CYRILLE DUBARRY* and RANDAL DOUC
Telecom SudParis/CITI, Evry, France

(Received 16 September 2010; in final form 8 October 2012)

This paper demonstrates a tractable and efficient way of calibrating a multiscale exponential Ornstein–Uhlenbeck stochastic volatility model including a correlation between the asset return and its volatility. As opposed to many contributions where this correlation is assumed to be null, this framework allows one to describe the leverage effect widely observed in equity markets. The resulting model is non-exponential and driven by a degenerate noise, thus requiring a high level of care in designing the estimation algorithm. The way this difficulty is overcome provides guidelines concerning the development of an estimation algorithm in a non-standard framework. The authors propose using a block-type expectation maximization algorithm along with particle smoothing. This method results in an accurate calibration process able to identify up to three timescale factors. Furthermore, we introduce an intuitive heuristic which can be used to choose the number of factors.

Keywords: Multiscale stochastic volatility model; Inference; Particle smoothing; Maximum split data estimate; Expectation-maximization algorithm

JEL Classification: C, C1, C13, C15

1. Introduction

The well-known Black–Scholes model is the first step in understanding financial pricing. However, it fails to include some statistical properties of observed financial data. This issue is partially due to the constant volatility used in the Black–Scholes model. As a consequence, more sophisticated models have been introduced to make the volatility non-constant: ARCH/GARCH models for discrete time introduced in Engle (1982) and Bollerslev (1986), and stochastic volatility models for continuous time (Hull and White 1987). In the latest model class, the exponential Ornstein–Uhlenbeck multiscale stochastic volatility model (ExpOU) is of real interest to take some financial data properties into account (see Masoliver and Perello 2006, Buchbinder and Chistilin 2007) and has been successfully used in Eisler et al. (2007) to infer the hidden volatility process in the particular case of one timescale. This model is presented below as a stochastic differential equation (SDE) system:

\[
\begin{align*}
\frac{dS_t}{S_t} &= \left[ \kappa dt + \sigma^S_t dB^S_t \right], \\
\frac{d\xi_t}{\xi_t} &= \frac{1}{2} \left( \begin{array}{cc} 1 & \xi_t \\ \xi_t & 0 \end{array} \right) dt + \frac{1}{2} \left( \begin{array}{c} 1 \\ b \end{array} \right) dB^2_t, \\
\end{align*}
\]

where \( S \) is the price of a financial asset, \( \kappa \) is a drift factor (which is not dealt with in this paper), \( \sigma^S \) is the volatility of \( S \) driven by a \( p \)-dimensional vector \( \xi \) of Ornstein–Uhlenbeck processes of parameters \( a = (a_1, \ldots, a_p)^T \), \( \mu = (\mu_1, \ldots, \mu_p)^T \) and \( b = (b_1, \ldots, b_p)^T \), and \( \langle \cdot, \cdot \rangle \) denotes the standard scalar product of two \( p \)-dimensional vectors. When coming to the application to real data, \( \kappa \) will be discarded using the detrended series defined in Kim et al. (1998). Without loss of generality, we will restrict our attention to the case where the components of \( a \) are distinct. Moreover, we assume that the volatility process has reached its stationary regime. The process \( (B^S, B^2) \) is a two-dimensional brownian motion with correlation \( \rho \), i.e. \( d(B^S, B^2) = \rho dt \) such that marginally, \( B^S \) and \( B^2 \) are standard brownian motions. \( 1_p \) denotes a \( p \)-dimensional vector of ones. Typically, each component \( \xi_i \) of \( \xi \) may be associated with the timescale \( 1/a_i \). The same brownian motion \( B^2 \) is driving all the components of \( \xi \). We could have chosen to include a more general correlation matrix,

*Corresponding author. Email: cyrille.dubarry@it-sudparis.eu
but in this case the model may become unidentifiable (see Fouque et al. 2008). This choice for the correlation matrix models the fact that market participants operate at different timescales not independently (intraday market makers, hedge funds or pension funds operate with quite different schedules).

A challenging issue with this model remains the estimation of its parameters \((a, \mu, b, \rho)\) given the observation of the process \(S\) over a fixed period of time. This is a tricky task since the volatility noise is degenerate and correlated to that of the asset. It implies that the complete log-likelihood is not even defined so that classical algorithms cannot be used within this framework. It thus fundamentally differs from the work of Eisler et al. (2007), which aims at reconstructing the hidden volatility in the particular case \(p=1\) and \(\rho=0\) for a fixed set of parameters. The calibration of a similar model has been performed in Fouque et al. (2008) in the particular case of \(p \in \{1, 2\}\) using a Markov chain Monte Carlo (MCMC) method. The main differences between Fouque et al. (2008) and the model introduced above is that in Fouque et al. (2008), the brownian motions driving \(S\) and \(\xi\) are independent and the components of \(\xi\) are driven by independent brownian motions, whereas in this paper we keep the same brownian motion for all the components of \(\xi\) (making it degenerate) correlated to that driving \(S\). As opposed to many contributions where this correlation between the asset return and its volatility is assumed to be null, this model allows one to describe the leverage effect widely observed in equity markets. In fact, early studies (e.g. Black 1976, Christie 1982) stated that a leverage effect widely observed in equity markets. In fact, the model is not affected by any permutation of the volatility of the volatility (\(b\)).

In this paper, we focus on a block-type variant of the expectation-maximization (EM) algorithm (introduced in Dempster et al. 1977) to perform the calibration. The expectation step is approximated using sequential Monte Carlo methods giving an estimate of posterior distributions in nonlinear and non-gaussian state-space models (Gordon et al. 1993, Del Moral 2004). An overview of these methods can be found in Doucet et al. (2001).

This technique implies working on a discretization of (1) which is done in section 2. In section 3, we develop a calibration method through a block-type expectation-maximization algorithm relying on a quantitative identifiability result (stated and proved in appendix A). The expectation step involved in this method is then handled in section 4. We finally apply the algorithm to simulated and real data (CAC 40 and Dow Jones indices over 10 years) in section 5 and give a criterion to select the number of factors. The proposed algorithm works well with up to three timescale factors \((p \in \{1, 2, 3\})\).

2. The discretized model

In order to discretize (1), we apply an Euler scheme associated to a fixed time step \(\delta\) and discretization grid \(\{t_k\}_{k \geq 0}\) with \(t_k = k\delta\):

\[
\begin{align*}
\tilde{S}_{t_{k+1}} - \tilde{S}_{t_k} &= \tilde{S}_{t_k} [\tilde{\xi}_k \delta + \tilde{\sigma}_k \sqrt{\delta} V_{t_{k+1}}], \\
\tilde{\sigma}_k^2 &= \exp \left( \frac{1}{2} \left( 1, \tilde{\xi}_k \right) \right), \\
\tilde{\xi}_{k+1} - \tilde{\xi}_k &= \text{diag}(\alpha)(\mu - \tilde{\xi}_k) \delta + b \sqrt{\delta} W_{t_{k+1}},
\end{align*}
\]

(2)

where \(\{(W_k, V_k)\}_{k \geq 1}\) is a sequence of i.i.d. two-dimensional gaussian vectors such that

\[
\begin{bmatrix}
W_1 \\
V_1
\end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} 0 \\ 1 \rho \end{bmatrix} \right).
\]

By setting

\[
Y_{k+1} = \frac{1}{\sqrt{\delta}} \begin{bmatrix} \tilde{S}_{t_{k+1}} - \tilde{S}_{t_k} \\ \tilde{\sigma}_k \end{bmatrix} \text{ and } X_k = \tilde{\xi}_k - \mu,
\]

equation (2) can be rewritten according to the following discretized model.

**Definition 2.1** – The discretized model:

\[
\begin{align*}
X_{k+1} &= \text{diag}(\alpha)X_k + \sigma W_{k+1}, \\
Y_{k+1} &= \beta e^{(1, X_k)^{1/2} V_{k+1}},
\end{align*}
\]

with \(\alpha = 1_p - \delta a, \sigma = \sqrt{\delta b} \) and \(\beta = \exp\left( \frac{1}{2} (1, \mu) \right)\). We assume that for \(i \in \{1, \ldots, p\}\), \(0 \leq \alpha_i < 1, \sigma_i > 0\). In addition, since the model is not affected by any permutation of the components of \(\alpha\), we restrict our attention to the case where \(\alpha_i < \alpha_{i+1}\) for \(i \in \{1, \ldots, p-1\}\).

The previously described model is parametrized by the \((2p + 2)\)-dimensional parameter vector \(\theta \in \Theta_p\), where

\[
\Theta_p \equiv \left\{ (\alpha, \sigma, \beta, \rho) \in [0, 1)^{p} \times (\mathbb{R}_+^*)^p \times \mathbb{R}_+^* \times (-1, 1); \alpha_1 < \cdots < \alpha_p \right\},
\]

and inference on this parameter will only be based on the observations \(Y_{k_i}\). At this stage, it is worthwhile to note that it is not possible to estimate separately the components of \(\mu\) since this vector appears in the discretized model only through the quantity \((1, \mu)\), which will be estimated by the parameter \(\beta\). In the following, we will assume that the distribution of \(X_0\) is the stationary distribution of the Markov chain \(X\):

\[
X_0 \sim \mathcal{N}(0, Y_{\bar{\alpha}, \sigma}),
\]

(4)

where the \(p \times p\) covariance matrix satisfies \(Y_{\bar{\alpha}, \sigma} = \text{diag}(\sigma) A_{\bar{\alpha}} \text{diag}(\sigma)\), with \(A_{\bar{\alpha}} = [(1 - \alpha_i)\alpha_j^{-1}]_{1 \leq i, j \leq p}\). Using the Cholesky decomposition of \(A_{\bar{\alpha}}\), \(Y_{\bar{\alpha}, \sigma}\) can be written...
\( Y_{x,\sigma} = \text{diag}(\sigma)L_x L_x^\top \text{diag}(\sigma) \), where \( L_x \) is the lower triangular matrix defined by

\[
\forall i, j \in \{1, \ldots, p\}, \quad (L_x)_{ij} = \begin{cases} 0, & j > i, \\ \left( \prod_{k=1}^{j-1} \frac{\alpha_i - \alpha_k}{\alpha_i \alpha_k} \right) \frac{1 - \alpha_i^2}{1 - \alpha_i \alpha_j}, & j \leq i. \end{cases}
\]

This decomposition allows one to simulate \( X_0 \) easily and shows that its distribution is not degenerate.

3. Inference

In order to infer the parameter \( \theta \) using a fixed set of observations \( Y_{1:T} \), a natural idea in hidden models would be to use an expectation-maximization (EM) algorithm. This algorithm relies on a recursive sequence (\( \hat{\theta}_n \)) in the parameter space \( \Theta_p \) satisfying

\[
\hat{\theta}_{n+1} = \arg\max_{\theta \in \Theta_p} E_{\hat{\theta}_n}[\log p_{\theta}(Y_{1:T}, U)|Y_{1:T}],
\]

where \( \log p_{\theta}(Y_{1:T}, U) \) is the log-likelihood of the joint distribution of \( (Y_{1:T}, U) \) for any hidden random variable \( U \). The choice of \( U \) is usually driven by the fact that: (i) the log-likelihood is easily computable; and (ii) the expectation and maximization steps in the EM algorithm can be separated. \( X_{0:T} \) is not a good candidate for \( U \): it is a degenerate Markov chain, for which \( \rho(\theta) \) cannot be defined. However, (3) suggests introducing the quantity \( X_{\ell} \equiv (X_{\ell}, X_{\ell}) \) and, indeed, it turns out that \( U = (X_{0:T}, X_{1:T}) \) meets the previous two requirements at the cost of having to estimate the smoothing distribution of all the couples \( (X_k, X_k) \), \( 1 \leq k < \ell \leq T \). As standard smoothing algorithms for approximating such joint distributions are degenerate for huge values of \( \ell - k \), we develop a block-type EM by splitting the observations into \( B \) smaller blocks to decrease the degeneracy. As will be detailed in subsection 3.2, this approach is closely linked to the split data likelihood technique introduced in Rydén (1994).

**Remark 1:** When \( X_0 \) is distributed according to (4), the process \( \tilde{X} \) is gaussian, centred and stationary. Consequently, this process is characterized by its autocovariance function \( \gamma_k^\tilde{X} \) given for all \( k \geq 0 \) by (using equation A2)

\[
\gamma_k^\tilde{X} \equiv \text{Cov}(\tilde{X}_0, \tilde{X}_k) = \sum_{j=1}^q Y_{x,\sigma}^j Y_{y,\sigma}^j.
\]

Thus the law of the process \( Y \) can be characterized by the law of \( \tilde{X} \), \( \beta \) and \( \rho \). This remark will be used in section 5 as a criterion for selecting the number of timescales factors when calibrating real data

**3.1. The standard EM algorithm**

As mentioned above, we first need to express the loglikelihood of the joint distribution of \( (Y_{1:T}, X_0, \tilde{X}_{1:T}) \), where \( X_0 \) is distributed according to (4). This is done in the following proposition.

**Proposition 3.1:** The log-likelihood of the joint distribution of the random variable \( (Y_{1:T}, X_0, \tilde{X}_{1:T}) \) is given by

\[
\log p_{\theta}(Y_{1:T}, X_0, \tilde{X}_{1:T}) = K - T \log(\beta) - \frac{1}{2} \sum_{k=1}^{T-1} \tilde{X}_k
\]

\[
- \frac{1}{2} \log|\text{det} Y_{x,\sigma}| - \frac{1}{2} \sum_{i=0}^{p} \left( \sum_{j=1}^q (Y_{x,\sigma})_{ij} (Y_{y,\sigma})_{ij} \right)
\]

\[
- \frac{1}{2} \beta^2 \left( 1 - \rho^2 \right) \sum_{i=1}^{T} Y_i^2 e^{-\tilde{X}_i} + \frac{\rho}{\beta(1 - \rho^2)} \sum_{k=1}^{T-1} \sum_{l=1}^{k} Y_k \omega_{k-l}
\]

\[
\times \left[ e^{-\tilde{X}_{k-1}/2} - \sum_{i=1}^{k} \alpha_i e^{-\tilde{X}_{k-1}/2} Y_i^2 \right]
\]

\[
- \frac{1}{2(1 - \rho^2)} \sum_{k=1}^{T} \sum_{l=1}^{k} \omega_{k-l} \omega_{k-m}
\]

\[
\times \left[ \tilde{X}_k \tilde{X}_m + \sum_{i=1}^{p} \alpha_i^m \alpha_i^j \omega_i^k \omega_i^l \right] - 2 \sum_{i=1}^{p} \alpha_i^m \tilde{X}_k \tilde{X}_m,
\]

where \( \tilde{X}_0 = (1_p, X_0) \), \( K \) is a deterministic constant which does not depend on \( \theta \), and

\[
\omega_0 = 1/0, \quad \omega_k = - \sum_{l=0}^{k-1} \omega_{k-l} \omega_{k-m}.
\]

Proof of this proposition is postponed to appendix A.2.

By plugging (7) into (5) with \( U = (X_0, \tilde{X}_{1:T}) \), the recursive sequence (\( \hat{\theta}_n \)) satisfies

\[
\hat{\theta}_{n+1} = \arg\max_{\theta \in \Theta_p} E_{\hat{\theta}_n}[\theta^\top Y_{1:T}]\]

where, for all \( 1 \leq r < s \leq T \),

\[
\epsilon_{\theta,\phi}(r,s) = - (s - r + 1) \log(\beta) - \frac{1}{2} \log|\text{det} Y_{x,\sigma}|
\]

\[
- \frac{1}{2} \sum_{i=1}^{p} \left( (Y_{x,\sigma})_{ij} \right) E_{\hat{\theta}_n}[Y_i^2 Y_{i-1}^2 Y_{r,s}]
\]

\[
- \frac{1}{2} \beta^2 \left( 1 - \rho^2 \right) \sum_{k=1}^{T} Y_i^2 E_{\hat{\theta}_n}[e^{-\tilde{X}_i} Y_{r,s}]
\]

\[
+ \frac{\rho}{\beta(1 - \rho^2)} \sum_{k=1}^{T-1} \sum_{l=1}^{k} Y_k \omega_{k-l} \left[ E_{\hat{\theta}_n}[e^{-\tilde{X}_{k-1}/2} \tilde{X}_l] Y_{r,s} \right]
\]

\[
- \sum_{i=1}^{p} \alpha_i E_{\hat{\theta}_n}[e^{-\tilde{X}_{k-1}/2} Y_i^2 Y_{r,s}]
\]

\[
- \frac{1}{2(1 - \rho^2)} \sum_{k=1}^{T} \sum_{l=1}^{k} \omega_{k-l} \omega_{k-m} \left[ E_{\hat{\theta}_n}[\tilde{X}_k \tilde{X}_m] Y_{r,s} \right]
\]

\[
+ \sum_{i=1}^{p} \alpha_i^m \alpha_i^j E_{\hat{\theta}_n}[Y_i^m Y_{i-1}^j] Y_{r,s},
\]

\[
- 2 \sum_{i=1}^{p} \alpha_i^m \tilde{X}_k \tilde{X}_m,
\]

(10)
Consequently, the expectation step of the EM algorithm (developed in detail in section 4) is separated from the maximization step and consists of estimating the following expectations:

\[
\begin{align*}
E_{\theta_{n+1}}[X_k^iX_0^j | Y_{1:T}] & \quad \forall (i,j) \in \{1, \ldots, p\}^2, \\
E_{\theta_{n+1}}[e^{-kX_{i-1}} | Y_{1:T}] & \quad \forall k \geq 1, \\
E_{\theta_{n+1}}[e^{-kX_{i-1}}X_0 | Y_{1:T}] & \quad \forall k \geq \ell \geq 1, \\
E_{\theta_{n+1}}[X_{i+1}^i | Y_{1:T}] & \quad \forall i \in \{1, \ldots, p\}, \\
E_{\theta_{n+1}}[X_{i}^i | Y_{1:T}] & \quad \forall i \in \{1, \ldots, p\}.
\end{align*}
\]

The maximization step is then performed with some deterministic optimization process (e.g., conjugate gradient method) over the whole parameter space \(\Theta\).

The estimation of these expectations thus requires obtaining an approximation of the smoothing distribution of all the couples \((\tilde{X}_k, \tilde{X}_j), 1 \leq k < \ell \leq T\), which is quite a difficult task when \(T\) is large. As noted above, we overcome this difficulty by proposing a block-type EM algorithm that will be described in the following subsection.

### 3.2. The block EM algorithm

The maximum split data likelihood estimate (MSDLE) introduced by Rydén (1994) consists in splitting the observations into blocks of fixed size \(\eta\) viewing these groups as independent and then maximizing the resulting likelihood. In other words, the estimator is obtained by maximizing the contrast function \(\sum_{u=0}^{B-1} \log p_{\theta_{n+1}}(Y_{\eta u+1:u+1})\) where \(B = T/\eta\) instead of the log-likelihood \(\log p_{\theta}(Y_{1:T})\). This allows one to derive asymptotic properties of the MSDLE from the M-estimator theory, provided that the size of the block \(\eta\) is large enough to guarantee that the asymptotic contrast function is maximized at the true value of the parameter. According to the identifiability result obtained in theorem A.1, taking \(\eta \geq 2p\) ensures that condition C4 in Rydén (1994) is satisfied and thus, using a generalized version of theorems 1 and 2 in Rydén (1994) to continuous state spaces (see Lai and Tung 2003), the MSDLE is consistent and asymptotically normal.

This estimator being intractable for the ExpOU model, we use the EM algorithm to maximize the contrast function by defining the intermediary random variables \(Z_{0:B-1}\) such that for all \(u \in \{0, \ldots, B-1\}\), \(Y_{\eta u+1:u+1}\) and \(Z_u\) have the same distribution but \((Z_u)_{u \in \{0, \ldots, B-1\}}\) are independent. Exploiting these two properties, the log-likelihood of \(Z\) is exactly the proposed contrast function of \(Y\):

\[
\log p_{\theta}(Z_{0:B-1}) = \sum_{u=0}^{B-1} \log p_{\theta}(Z_u) = \sum_{u=0}^{B-1} \log p_{\theta}(Y_{\eta u+1:u+1}).
\]

For any hidden random variable \((V_u)_{u \in \{0, \ldots, B-1\}}\), the EM algorithm suggests recursively computing the sequence \((\hat{\theta}_n)\) defined by

\[
\hat{\theta}_{n+1} = \text{argmax}_\theta \log p_{\theta}(Z_{0:B-1}, V_{0:B-1} | Z_{0:B-1}).
\]

We choose \(V\) such that for all \(u \in \{0, \ldots, B-1\}\), \((Y_{\eta u+1:u+1}, X_{\eta u}, \tilde{X}_{\eta u+1:u+1})\) and \((Z_u, V_u)\) have the same distribution but \((Z_u, V_u)_{u \in \{0, \ldots, B-1\}}\) are independent. Then we have

\[
\begin{align*}
E_{\hat{\theta}_n}[\log p_{\theta}(Z_{0:B-1}, V_{0:B-1} | Z_{0:B-1})] &= \sum_{u=0}^{B-1} E_{\hat{\theta}_n}[\log p_{\theta}(Z_u, V_u) | Z_{0:B-1}] \\
&= \sum_{u=0}^{B-1} E_{\hat{\theta}_n}[\log p_{\theta}(Z_u, V_u) | Z_u] \\
&= \sum_{u=0}^{B-1} E_{\hat{\theta}_n}[\log p_{\theta}(Y_{\eta u+1:u+1}, X_{\eta u+1:u+1}, \tilde{X}_{\eta u+1:u+1} | Y_{\eta u+1:u+1})].
\end{align*}
\]

Consequently, we can replace the quantity defined in (10) with

\[
L_{u,\theta}^{B,T}(Y_{1:T}) = \sum_{u=0}^{B-1} E_{\hat{\theta}_n}[\log p_{\theta}(Y_{\eta u+1:u+1}, X_{\eta u+1:u+1}, \tilde{X}_{\eta u+1:u+1} | Y_{\eta u+1:u+1})].
\]

This splitting technique points out a new issue: the choice of \(\eta\) which is directly linked to the estimator efficiency. As already mentioned, the identifiability result obtained in theorem A.1 shows that \(\eta\) should be necessarily greater than 2p, and small enough to avoid degeneracy of the smoothing algorithm.

### 4. Expectation step

In section 3 we have seen that inference about the parameter \(\theta\) based on a fixed set of observations \(Y_{1:T}\) requires evaluating expectations of the form given in (11). To that purpose, we aim to approximate the smoothing distribution of \((X_0, \tilde{X}_{1:T})\) conditionally to \(Y_{1:T}\).

For the block EM algorithm, the smoothing distributions to approximate are for all \(u \in \{0, \ldots, B-1\}\), \((X_{u:T+B-1}, \tilde{X}_{u:T+B-1} | \tilde{X}_{1:T+B-1})\) knowing \(Y_{u:T+B-1} | \tilde{X}_{1:T+B-1}\). We only need to focus on \((X_0, \tilde{X}_{1:T+B-1})\) conditionally to \(Y_{1:T+B-1}\). For clarity reasons, we will keep \(T\) as the time horizon.

Standard smoothing algorithms could be performed only if \(\tilde{X}\) were a Markov chain. Unfortunately it is not (see remark A1 in appendix A.1) and we have to replace it by a more suitable hidden variable before demonstrating two closely linked smoothing algorithms: the genealogical tree and the forward filtering backward simulation algorithms.

#### 4.1. Replacing the hidden variable

An obvious idea to handle the fact that \(\tilde{X}\) is not in general a Markov chain is to recall that \(\tilde{X}_k = (I_g, X_k)\), where \(X\) is
a Markov chain such that the smoothing distribution of $X$ could be approximated with a particle smoother. However, this Markov chain is degenerate, preventing most smoothing algorithms from being used. This issue can be overcome by a decimation in time for $X$. To that purpose, we define the $p$-dimensional random vector $\tilde{X}$ for all $k \geq 0$ by

$$\forall k \geq 0, \quad \tilde{X}_k = X_{pk},$$

and the $p$-dimensional random vector $\tilde{Y}$ for all $k \geq 1$ by

$$\forall k \geq 1, \quad \tilde{Y}_k = [Y_{p(k-1)+1}, \ldots, Y_{pk}]^T.$$ 

Then we check that smoothing algorithms can be performed on the hidden model defined by $\tilde{X}$ and $\tilde{Y}$ in the following proposition.

**Proposition 4.1:** $\tilde{X}$ is a Markov chain, is not degenerate and

$$\tilde{X}_k | \tilde{X}_{k-1} \sim \mathcal{N}(\text{diag}(\sigma)^p \tilde{X}_{k-1}, \Sigma(p) \Sigma(p)^T),$$

where, for all $q \in \{1, \ldots, p\}$, $\Sigma(q)$ is a $p \times p$ matrix defined by

$$\Sigma(q) = \begin{bmatrix} 0 & \cdots & \sigma_1 \alpha_{q1} & \cdots & \sigma_1 \alpha_{q1}^{-1} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ 0 & \cdots & \sigma_p \alpha_{qP} & \cdots & \sigma_p \alpha_{qP}^{-1} \end{bmatrix},$$

in particular, $\Sigma(p) = \text{diag}(\sigma)V_d(\alpha)^T$ and, by convention, $\Sigma(0)$ is the $p \times p$ null matrix.

Furthermore, the law of $\tilde{Y}_k$ conditionally to $\tilde{X}_{0:k}$ only depends on $(\tilde{X}_{k-1}, \tilde{X}_k)$ and

$$\tilde{Y}_k | \tilde{X}_{k-1}, \tilde{X}_k \sim \mathcal{N}\left(\rho \left[ e^{\sigma_{x,y}^{-1}(\tilde{X}_{k-1}, \tilde{X}_k)} g_{\sigma_{x,y}}(\tilde{X}_{k-1}, \tilde{X}_k) \right]_{1 \leq q \leq p}, (1 - \rho^2)^{\frac{p^2}{2}} \text{diag}\left( \left[ e^{\sigma_{x,y}^{-1}(\tilde{X}_{k-1}, \tilde{X}_k)} \right]_{1 \leq q \leq p} \right) \right),$$

where the two function families $(f_{\sigma_{x,y}})_{0 \leq q \leq p-1}$ and $(g_{\sigma_{x,y}})_{0 \leq q \leq p}$ are defined for all $q \in \{1, \ldots, p\}$, all $x \in \mathbb{R}^p$ and all $y \in \mathbb{R}^p$ by

$$f_{\sigma_{x,y}}(x, x') = (1_p, \text{diag}(\alpha)^{-1} x + \Sigma(q-1) \Sigma(p)^{-1} (x' - \text{diag}(\alpha)^{-1} x)),
$$

$$g_{\sigma_{x,y}}(x, x') = [\Sigma(p)^{-1} (x' - \text{diag}(\alpha)^{-1} x)]_{p+1-q}.$$ 

Finally, $\tilde{X}_k$ can be found back from $\tilde{X}$ for all $k \geq 1$ and $q \in \{0, \ldots, p-1\}$,

$$\tilde{X}_{p(k-1)+q} = f_{\sigma_{x,y}}(\tilde{X}_{k-1}, \tilde{X}_k).$$

**Proof:** It is direct to see that

$$\tilde{X}_k = \text{diag}(\alpha)^p \tilde{X}_{k-1} + \Sigma(p) [W_{pk}, \ldots, W_{p(k-1)+1}]^T.$$ 

This shows that $\tilde{X}$ is a Markov chain and its transition law is given by (13). $\Sigma(p)$ being invertible, $\tilde{X}$ is not degenerate.

Then it is possible to compute the missing $X_k$ through a deterministic function of $\tilde{X}$. First, we invert (16)

$$[W_{pk}, \ldots, W_{p(k-1)+1}]^T = \Sigma(p)^{-1}(\tilde{X}_k - \text{diag}(\alpha)^p \tilde{X}_{k-1}),$$

so that

$$X_{p(k-1)+q} = \text{diag}(\alpha)^p \tilde{X}_{k-1} + \Sigma(q) \Sigma(p)^{-1} (\tilde{X}_k - \text{diag}(\alpha)^p \tilde{X}_{k-1}).$$

From the two previous equations, we get

$$\tilde{X}_{p(k-1)+q} = \{1_p, X_{p(k-1)+q}\} = f_{\sigma_{x,y}}(\tilde{X}_{k-1}, \tilde{X}_k),$$

$$W_{p(k-1)+q} = g_{\sigma_{x,y}}(\tilde{X}_{k-1}, \tilde{X}_k).$$

Finally the proof is completed by writing $Y$ as follows:

$$Y_{p(k-1)+q} = \rho^2 e^{\sigma_{x,y}^{-1}(\tilde{X}_{k-1}, \tilde{X}_k)}^2 \left[ \rho g_{\sigma_{x,y}}(\tilde{X}_{k-1}, \tilde{X}_k) + \sqrt{1 - \rho^2} \tilde{Y}_{p(k-1)+q} \right].$$

Consequently, (13) and (14) can be used in any smoothing algorithm to get an approximation of the law of $\tilde{X}_{1:T/p}$ knowing $\tilde{Y}_{1:T/p}$. Then, the expectations involved in (10) and (12) are computed from $\tilde{X}_{1:T/p}$ conditionally to $\tilde{Y}_{1:T/p}$ using (15).

### 4.2. Smoothing algorithms

We consider the model illustrated in figure 1. $\tilde{X}$ is an unobserved Markov chain and at time $k$, the observation $\tilde{Y}_k$ depends on $(\tilde{X}_{k-1}, \tilde{X}_k)$ and not only on $\tilde{X}_k$ so that standard smoothing algorithms (Doucet et al. 2001) have to be adapted to this particular case. We first show that the classical bootstrap filter can be easily extended to figure 1 as it will be the base for most smoothing algorithms. Then we apply it to the genealogical tree method and focus carefully on how the forward filtering backward simulation (FFBSI) algorithm is impacted by this specific dependence structure.

#### 4.2.1. Adapted bootstrap filter.

In the framework of the bootstrap filter, we iteratively approximate the filtering distribution $p(\tilde{X}_i | Y_{1:i})$ of $\tilde{X}_i$ conditionally to $Y_{1:i}$ (with convention $p(\tilde{X}_0 | Y_{1:0}) = p(\tilde{X}_0)$) by $\hat{p}_N(d\tilde{X}_i | Y_{1:i})$ using a set of $N$ weighted particles $(\omega_k, \tilde{X}_k)$ such that

$$\hat{p}_N(d\tilde{X}_i | Y_{1:i}) = \left( \sum_{i=1}^N \omega_k \right)^{-1} \sum_{i=1}^N \omega_k \delta_{\tilde{X}_k}(d\tilde{X}_i),$$

where $\delta$ is the Dirac measure. The way of drawing the particles and the updating formulas for the weights are naturally derived from the following equation. For all $k \geq 1$,

$$p(\tilde{X}_i | Y_{1:i+1}) = \frac{\int p(\tilde{X}_i | Y_{1:i}) p(\tilde{X}_{i+1} | \tilde{X}_i, \tilde{X}_{i+1}) d\tilde{X}_i}{\int p(\tilde{X}_i | Y_{1:i}) p(\tilde{X}_{i+1} | \tilde{X}_i, \tilde{X}_{i+1}) d\tilde{X}_i}.$$
Each iteration of this procedure consists of a selection step (optional) and a mutation step. The selection step resamples particles \( \xi_k^i \) by drawing independent indices \( I_k^i \) with probability proportional to the weights \( \omega_k^i \). Then, the mutation step draws particles \( \xi_{k+1}^j \) independently such that for all \( i \in \{1, \ldots, N\} \),
\[
\xi_{k+1}^i \sim p(\xi_{k+1}^j | \xi_k^i). 
\]
Finally, the weights are updated by setting \( \omega_{k+1}^i = p(\tilde{y}_{k+1}^i | \xi_{k+1}^j, \xi_{k+1}^i) \). This is summarized in algorithm 1.

**Algorithm 1:** Adapted bootstrap filter

1: sample \( (\xi_k^i)^N \) independently according to \( p(\tilde{x}_0) \)
2: \( (\omega_0^i, \ldots, \omega_0^N) \leftarrow (1/N, \ldots, 1/N) \)
3: for \( k \) from 0 to \( T/p - 1 \)
4: sample \( I_k^i \) multinomially with probability proportional to \( \omega_k^i \)
5: sample \( \xi_{k+1}^i \sim p(\xi_{k+1}^j | \xi_k^i) \)
6: \( \omega_{k+1}^i = p(\tilde{y}_{k+1}^i | \xi_{k+1}^j, \xi_{k+1}^i) \)
7: end for

### 4.2.2. Adapted genealogical tree

The genealogical tree algorithm (Gordon et al. 1993, Del Moral 2004) approximates the smoothing distribution \( p(\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) \) of \( \tilde{x}_{0:T/p} \) conditionally to \( \tilde{y}_{1:T/p} \) by \( \hat{p}_N(\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) \) using the particles \( \tilde{x}_{0:T/p} \) and the genealogy \( G_{0:T/p} \), computed by the bootstrap filter (algorithm 1) such that
\[
\hat{p}_N(d\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) = \sum_{i=1}^{N} \frac{1}{\omega_{T/p}^i} \left( \frac{d\tilde{x}_{0:T/p}}{1/N} \right)^{-1} \times \sum_{i=1}^{N} \omega_{T/p}^i \delta(\tilde{x}_{0:T/p} - \tilde{x}_{0:T/p}^i),
\]
where \( J_{k+1}^{i} \) is deterministically defined from the genealogy \( I_{k+1}^{i} \) by the following backward recursion for all \( i \in \{1, \ldots, N\} \):
\[
J_{k+1}^i = \begin{cases} 
1 & i, \\
J_k^{i+1} & \forall k \in \{0, \ldots, T/p - 1\}.
\end{cases}
\]

### 4.2.3. Adapted forward filtering backward simulation

The forward filtering backward simulation (FFBSi) algorithm (described in Doucet et al. 2000 and analysed in Douc et al. 2011) approximates the smoothing distribution \( p(\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) \) of \( \tilde{x}_{0:T/p} \) conditionally to \( \tilde{y}_{1:T/p} \) with \( \hat{p}_N(\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) \) through two simulation passes. The first one consists of computing the weighted particles \( (\omega_{0:T/p}^i, \tilde{x}_{0:T/p}^i) \) using the bootstrap filter (algorithm 1). The second one draws backward \( N \) independent paths of particle indices \( J_{0:T/p}^N \) from the set \( \{1, \ldots, N\}^{T/p+1} \) such that
\[
\hat{p}_N(d\tilde{x}_{0:T/p} | \tilde{y}_{1:T/p}) = N^{-1} \sum_{i=1}^{N} \delta(\tilde{x}_{0:T/p} - \tilde{x}_{0:T/p}^i). 
\]
The way of drawing the indices \( J_{0:T/p}^N \) is derived from the following equation. For all \( k \in \{0, \ldots, T/p - 1\} \),
\[
p(\tilde{x}_{k+1:T/p} | \tilde{y}_{k:T/p}) = p(\tilde{x}_{k+1:T/p} | \tilde{y}_{k:T/p}) \times \int p(\tilde{x}_k | \tilde{y}_k) p(\tilde{x}_{k+1:T/p} | \tilde{x}_k, \tilde{y}_{k+1:T/p}) d\tilde{x}_k.
\]
As a consequence, for \( i \in \{1, \ldots, N\} \), \( J_{0:T/p}^N \) can be independently sampled backward in time according to
\[
\mathbb{P}\{J_k^i = j \} \propto \omega_k^i.
\]
This extension of the FFBSi algorithm is summarized in algorithm 2.

**Algorithm 2:** Adapted FFBSi

1: sample \( (\omega_{0:T/p}^i, \tilde{x}_{0:T/p}^i) \) using algorithm 1
2: sample \( J_{0:T/p}^N \) multinomially with probability proportional to \( \omega_{0:T/p}^i \)
3: for \( k \) from \( T/p - 1 \) down to 0
4: for \( i \) from 1 to \( N \)
5: sample \( \tilde{x}_k^i \) with probability proportional to \( \omega_k^i \)
6: end for
7: end for

### 5. Application

In this section, we apply the previously described algorithm with \( N = 3000 \) particles to simulated and real data. We first show experimentally that the proposed technique can capture well up to three timescale factors and identify a heuristic to choose the number of factors. This criterion is based on remark 1. When coming to calibrating real data, we use the detrended returns (Kim et al. 1998) of the CAC 40 and Dow Jones indices over 10 years.

<p>| Table 1. Settings’ definition. |</p>
<table>
<thead>
<tr>
<th>Setting 1</th>
<th>Setting 2</th>
<th>Setting 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p )</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>( a_1 )</td>
<td>0.6</td>
<td>0.3</td>
</tr>
<tr>
<td>( a_2 )</td>
<td>-0.98</td>
<td>0.6</td>
</tr>
<tr>
<td>( a_3 )</td>
<td>-0.98</td>
<td></td>
</tr>
<tr>
<td>( \alpha_1 )</td>
<td>0.4</td>
<td>0.8</td>
</tr>
<tr>
<td>( \alpha_2 )</td>
<td>-0.2</td>
<td>0.6</td>
</tr>
<tr>
<td>( \alpha_3 )</td>
<td>-0.2</td>
<td></td>
</tr>
<tr>
<td>( \beta )</td>
<td>0.22</td>
<td>0.22</td>
</tr>
<tr>
<td>( \rho )</td>
<td>-0.1</td>
<td>-0.2</td>
</tr>
</tbody>
</table>

| Table 2. Block size. |
|---|---|---|
| \( \eta \) | 15 | 20 | 30 |
In order to evaluate the performance of the algorithm to identify up to three timescale factors, we have generated $T = 1500$ observations $Y_{1:T}$ for three different settings.

### Table 3. Estimated parameters from simulated data (real parameter between brackets when applicable).

<table>
<thead>
<tr>
<th>$p$</th>
<th>Setting 1</th>
<th>Setting 2</th>
<th>Setting 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

| $\alpha_1$ | 0.60 (0.6) | 0.32 | 0.10 |
| $\alpha_2$ | - 0.90 | 0.59 | - |
| $\alpha_3$ | - - 0.997 | - - |
| $\sigma_1$ | 0.39 (0.4) | 0.37 | 0.21 |
| $\sigma_2$ | - 0.08 | 0.23 | - |
| $\sigma_3$ | - - 0.11 | - - |
| $\beta$ | 0.20 (0.2) | 0.20 | 0.20 |
| $\rho$ | $-0.12$ (−0.1) | $-0.13$ | $-0.11$ |

### Table 4. Differences of auto-covariance functions estimated from simulated data.

<table>
<thead>
<tr>
<th>Setting 1</th>
<th>Setting 2</th>
<th>Setting 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_{1,2}$</td>
<td>0.09</td>
<td>4.37</td>
</tr>
<tr>
<td>$d_{2,3}$</td>
<td>0.24</td>
<td>0.78</td>
</tr>
</tbody>
</table>

5.1. Simulated data

In order to evaluate the performance of the algorithm to identify up to three timescale factors, we have generated $T = 1500$ observations $Y_{1:T}$ for three different settings.
Then, for each setting, the calibration algorithm has been run for \( p \in \{1, 2, 3\} \) with 100 EM iterations, \( N = 3000 \) particles and a different block size \( \eta \) for each value of \( p \) given in table 2. As seen in subsection 3.2, choosing \( \eta \) depending on \( p \) is crucial in order to guaranty the identifiability which ensures that \( \hat{\theta}_n \) converges to the true value of the parameters. The convergence of the algorithm is shown in figures 2, 3 and 4 for their respective values of \( p \) and in figures B1, B2 and B3 in appendix B for other values of \( p \). The estimated parameters are given in table 3.

These results show that the parameter \( \theta \) is accurately estimated by our algorithm for up to three factors when knowing the value of \( p \) even though the convergence might occur after an excursion of \( \hat{\theta}_n \) far from the true value as the maximization steps do not take into account any \textit{a priori} knowledge of the parameters. Moreover, the estimation of \( \theta \) with the wrong number of factors leads to similar values of \( (\beta, \rho) \). The estimated \( (\alpha, \sigma) \) cannot be compared directly as they do not belong to the same space. Following remark 1, we compare instead the auto-covariance function \( \gamma_X \) defined in (6) which characterizes the law of \( X \). For any setting, we denote by \( \tilde{\gamma}_X(\alpha, \sigma) \) the auto-covariance function induced by \( (\alpha, \sigma) \) estimated with \( p \) timescale factors. Consequently, a possible measure of the difference between the law estimated with \( p_1 \) and \( p_2 \) factors is

\[
d_{p_1, p_2} \triangleq \left\| \gamma_{X_{\alpha_1, \sigma_1}} - \gamma_{X_{\alpha_2, \sigma_2}} \right\|_2 = \sqrt{\sum_{k=0}^{99} \left( \tilde{\gamma}_{X_{\alpha_1}}(k) - \tilde{\gamma}_{X_{\alpha_2}}(k) \right)^2}.
\]

This quantity is computed for each setting in table 4. As expected, the difference in law when going from one to two or from two to three factors is very small for setting 1, meaning that one factor is sufficient to model the data. For setting 2, the difference is important when going from one to two factors and negligible when going from two to three factors, meaning that one factor is not enough to model the data and that two factors are sufficient to do so. Finally, for setting 3, the difference in law when going from one to two or from two to three factors is significant, meaning that at least three factors are needed.

### Table 5. Estimated parameters from CAC 40 and Dow Jones.

<table>
<thead>
<tr>
<th>( p )</th>
<th>( \alpha_1 )</th>
<th>( \alpha_2 )</th>
<th>( \alpha_3 )</th>
<th>( \beta )</th>
<th>( \rho )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.96</td>
<td></td>
<td></td>
<td></td>
<td>0.53</td>
</tr>
<tr>
<td>2</td>
<td>0.98</td>
<td>0.19</td>
<td>0.18</td>
<td>0.53</td>
<td>0.41</td>
</tr>
<tr>
<td>3</td>
<td>0.98</td>
<td>0.26</td>
<td>0.15</td>
<td>0.53</td>
<td>0.54</td>
</tr>
</tbody>
</table>

### Table 6. Differences of auto-covariance functions estimated from CAC 40 and Dow Jones.

<table>
<thead>
<tr>
<th>( d_{1,2} )</th>
<th>( d_{2,3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAC 40</td>
<td>Dow Jones</td>
</tr>
<tr>
<td>1.11</td>
<td>3.45</td>
</tr>
<tr>
<td>0.36</td>
<td>0.80</td>
</tr>
</tbody>
</table>

Figure 5. Daily CAC 40 spot (left) and detrended returns (right) for the period 1 August 2000–9 August 2010.

Figure 6. Daily Dow Jones spot (left) and detrended returns (right) for the period 1 August 2000–9 August 2010.
5.2. Real data

The calibration algorithm has been applied to equity market data: the CAC 40 and Dow Jones indices over the past ten years (from 1 August 2000 to 9 August 2010). Their daily spot and returns, detrended according to Kim et al. (1998), are plotted in figures 5 and 6.

The estimated parameters with different numbers of factors are presented in table 5 and the convergence of the algorithm is displayed in appendix B, figures B4 and B5. An analysis similar to that for the simulated data can be done. First, we remark that the estimated (β, ρ) are approximately the same whichever value of p and negative value of ρ is consistent with the leverage effect mentioned in the introduction (Black 1976, Christie 1982). Then, we compute the auto-covariance differences (see definition 17) in table 6 to select the most appropriate number of factors. For both the CAC 40 and Dow Jones indices, the difference is important when going from one to two factors and negligible when going from two to three factors, meaning that one factor is not enough to model the data and that two factors are sufficient to do so. Consequently, we have captured a short timescale factor (about one day) and a long one (a few months). Identification of these two different timescales is a desired characteristic of the multiscale exponential Ornstein–Uhlenbeck model (see Chernov et al. 2003).

6. Conclusion

This paper demonstrates a tractable and efficient way of calibrating a stationary multiscale exponential Ornstein–Uhlenbeck model including a correlation between the asset and its volatility. To do so, a precise identifiability result justifies the use of the MSDLE. This estimate is then approximated by a stochastic EM algorithm involving particle smoothing. Estimation in a non-exponential model with degenerate noise requires particular care regarding the hidden variables which will be chosen differently for each step of the estimation method design (identifiability, E-step/M-step separation and particle smoothing). More generally, this provides guidelines concerning estimation algorithm design in a non-standard framework.

Experiments show that the proposed algorithm is able to identify up to three timescale factors and an intuitive heuristic allows one to select the number of factors. This criterion gives a measure of the difference in law induced when running the estimation algorithm with various numbers of factors. The choice of one, two or three factors can be driven by this heuristic.

References


Appendix A: Technical proofs

A1. Identifiability of the model

Theorem A.1: Let \( \theta^0 = (\sigma^0, \beta^0, \rho^0) \), \( i \in \{1, 2\} \), be two sets of parameters in \( \Theta_\rho \) and define two pairs of processes \( (X^0, Y^0) \), \( i \in \{1, 2\} \), such that, for all \( k \geq 0 \),

\[
X^{0,i}_{k+1} = \text{diag}(x^{(i)})X^{0,i}_k + \sigma^{0,i}W^{0,i}_{k+1}, \\
Y^{0,i}_{k+1} = \beta^{0,i}e^{(1-x^{0,i})/2}Y^{0,i}_{k+1},
\]

(A1)
where \(\{(W_k^{(1)}, V_k^{(1)})\}_{k \geq 1}\) and \(\{(W_k^{(2)}, V_k^{(2)})\}_{k \geq 1}\) are two independent sequences of i.i.d. two-dimensional gaussian vectors such that
\[
\begin{bmatrix} W_k^{(j)} \\ V_k^{(j)} \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right),
\]
and \(X_0^{(i)} \sim \mathcal{N}(0, Y_{\alpha^i, \sigma^i})\). Then the three following assertions are equivalent:

(i) \(Y_k^{(1)} \leq \varphi_k^{(2)}\), (ii) \(\theta_k^{(1)} = \theta_k^{(2)}\), (iii) \(\forall k \geq 1, Y_k^{(1)} \leq \varphi_k^{(2)}\).

The proof of Theorem A.1 mainly relies on a rewriting of the model. As stated in the following proposition, the distribution of \(Y\) conditionally to \(X\) only depends on the random quantities \(X_0\) and \(\tilde{X}_k = (1_p, X_k)\) rather than on \(X\) itself.

**Proposition A.2:** The model (3) can be written as follows:
\[
\begin{align*}
\tilde{X}_k &= \{x^k, X_0\} + \sum_{\ell=1}^k k - \ell W_\ell, \\
Y_k &= \beta \tilde{X}_{1-k/2} \left\{ \rho \sum_{\ell=1}^k a_{\ell-k} \left[ \tilde{X}_\ell - \{x^\ell, X_0\} \right] + \sqrt{1 - \rho^2} \tilde{V}_k \right\},
\end{align*}
\]
(A2)

where \(\{(W_k, \tilde{V}_k)\}_{k \geq 1}\) is a sequence of i.i.d. two-dimensional independent standard gaussian vectors, \(a^k = (a_1, \ldots, a_k)^\top\) and \((a_k)_{k \geq 0}\) is defined in (8).

**Remark A1:** By straightforward algebra, one can invert the first equation of (A2), \(W_\ell = \sum_{m=1}^{\ell} a_{\ell-m} [\tilde{X}_m - \{x^m, X_0\}]\) and plug it again into (A2) so that \(\tilde{X}\) is not in general a Markov chain. Nevertheless, this is not an issue when proving the identifiability of the model.

**Proof:** By expanding (3), we obtain for \(i \in \{1, \ldots, p\}\), \(X_k^i = a_i^i X_0^i + \sum_{\ell=1}^k \sigma_i \{x^{\ell-i}, W_\ell\}\) so that
\[
\tilde{X}_k = (1_p, X_k) = \{x^k, X_0\} + \sum_{\ell=1}^k \{\sigma_i, x^{\ell-i}\} W_\ell.
\]
(A3)

A direct calculation from (A3) shows that
\[
W_k = \sum_{\ell=1}^k a_{k-\ell} [\tilde{X}_\ell - \{x^\ell, X_0\}],
\]
(A4)

where \((a_k)_{k \geq 0}\) is defined in (8). As \(\text{Corr}(W_k, V_\ell) = \rho V_{k-\ell}\), the proof is completed by rewriting the random variable \(V_k\) as follows:
\[
V_k = \rho W_k + \sqrt{1 - \rho^2} \tilde{V}_k
\]
\[
= \rho \sum_{\ell=1}^k a_{k-\ell} [\tilde{X}_\ell - \{x^\ell, X_0\}] + \sqrt{1 - \rho^2} \tilde{V}_k,
\]
where \(\{W_k, \tilde{V}_k\}_{k \geq 1}\) is a sequence of i.i.d. two-dimensional independent standard gaussian vectors.

**Lemma A.3:** Let \(\theta^0 = (\alpha^0, \sigma^0, \rho^0, \varphi^0), i \in \{1, 2\}\), be two sets of parameters in \(\Theta_p\) and define two pairs of processes \((X_k^{(i)}, Y_k^{(i)}), i \in \{1, 2\}\), as in (A1) in appendix A.1. Then, for all \(k \geq 1\),
\[
Y_k^{(1)} \leq Y_k^{(2)} \implies \begin{cases} \beta_k^{(1)} = \beta_k^{(2)}, \\ \rho_k^{(1)} (\sigma_k^{(1)} (x_k^{(1)})^m) = \rho_k^{(2)} (\sigma_k^{(2)} (x_k^{(2)})^m), \forall m \leq k - 1, \\ \delta_k^{(1)} Y_{x_k^{(1)}, \sigma_k^{(1)}} (x_k^{(1)})^m = \delta_k^{(2)} Y_{x_k^{(2)}, \sigma_k^{(2)}} (x_k^{(2)})^m, \forall m \leq k. \end{cases}
\]

**Proof:** To obtain the distribution of the \(Y\), we substitute the expression for \(\tilde{X}\) in (A2) so that, for all \(k \geq 1\),
\[
Y_k^{(i)} = \beta k \left( \sum_{\ell=1}^k \left[ \bar{\alpha}_k^{(i)} \right]^2 \right)^{1/2} \frac{1}{\sqrt{1 - \rho^2}} \tilde{V}_k^{(i)},
\]
(A5)

where \(\left\{ W_k^{(1)}, \tilde{V}_k^{(1)}, V_k^{(2)}, W_k^{(2)}, k \geq 1 \right\}\) are i.i.d. standard gaussian variables and \(X_0^{(i)} \sim \mathcal{N}(0, Y_{\alpha^{(i)}, \sigma^{(i)}})\).

Let \(k \geq 1\) and assume that \(Y_k^{(1)} = Y_k^{(2)}\). Then, for all \(s \in \mathbb{N}^*, E(Y_{x_k^{(i)}, \sigma_k^{(i)}}) = E(Y_{x_k^{(i)}, \sigma_k^{(i)}}^{(2)})\) and for all \(i \in \{1, 2\}\),
\[
E\left[ \left( Y_k^{(i)} \right)^{2m} \right] = \left( \beta k \right)^{2m} \exp \left( \frac{1}{2} \sigma_k^{(i)} \frac{1}{\bar{\alpha}_k^{(i)}} \right) \left[ Y_{x_k^{(i)}, \sigma_k^{(i)}} \right]^{2m} = \left( \beta k \right)^{2m} \exp \left( \frac{1}{2} \sigma_k^{(i)} \frac{1}{\bar{\alpha}_k^{(i)}} \right) \left[ Y_{x_k^{(i)}, \sigma_k^{(i)}} \right]^{2m},
\]
so that
\[
\rho_k^{(i)} (\sigma_k^{(1)} (x_k^{(1)})^m) = \rho_k^{(2)} (\sigma_k^{(2)} (x_k^{(2)})^m), \forall m \leq k - 1.
\]
(A6)

As a consequence, if \(\rho_k^{(1)} \neq 0\) or \(\rho_k^{(2)} \neq 0\), then \(\rho_k^{(1)} \neq 0\) and \(\rho_k^{(2)} \neq 0\) and (A6) leads to \(\rho_k^{(1)} (\sigma_k^{(1)} (x_k^{(1)})^{m+2}) = \rho_k^{(2)} (\sigma_k^{(2)} (x_k^{(2)})^{m+2})\) and \(\beta_k^{(1)} Y_{x_k^{(1)}, \sigma_k^{(1)}} (x_k^{(1)})^{m+2} = \beta_k^{(2)} Y_{x_k^{(2)}, \sigma_k^{(2)}} (x_k^{(2)})^{m+2}\).

On the other hand, if \(\rho_k^{(1)} = 0\) or \(\rho_k^{(2)} = 0\), then \(\rho_k^{(1)} = \rho_k^{(2)} = 0\) and for all \(i \in \{1, 2\}\),
\[
E\left[ \begin{bmatrix} Y_k^{(1)} \\ Y_k^{(2)} \end{bmatrix} \right] = \left( \beta k \right)^{2m} \left[ \delta_k^{(1)} Y_{x_k^{(1)}, \sigma_k^{(1)}}^{(2)} = \delta_k^{(2)} Y_{x_k^{(2)}, \sigma_k^{(2)}}^{(2)} \left( \bar{\alpha}_k^{(i)} \right)^{2m+1}, \right.
\]
which leads again to \(\beta_k^{(1)} Y_{x_k^{(1)}, \sigma_k^{(1)}}^{(2)} = \delta_k^{(2)} Y_{x_k^{(2)}, \sigma_k^{(2)}}^{(2)} \left( \bar{\alpha}_k^{(i)} \right)^{2m+1}\).
Proof of theorem A.1: (ii) ⇒ (iii) and (iii) ⇒ (i) are obvious. Let’s show (i) ⇒ (ii). If (i) holds true, then by lemma A.3 we have \( \beta^{(1)} = \beta^{(2)} \),

\[ \forall k \in \{0, \ldots, 2p - 1\}, \quad Y_{p}^{(1)} Y_{x^{(1)}, \alpha^{(1)}}(x^{(1)})^{k} = Y_{p}^{(2)} Y_{x^{(2)}, \alpha^{(2)}}(x^{(2)})^{k}, \]  

(A7)

Equation (A8) shows that if we assume \( \sigma^{(1)} = \sigma^{(2)} \), then \( \rho^{(1)} = \rho^{(2)} \). As a consequence it only remains to show that \( \alpha^{(1)} = \alpha^{(2)} \) and \( \alpha^{(1)} = \alpha^{(2)} \). To that purpose, we rewrite (A7) into the following equivalent assertion:

\[ \forall k \in \{0, \ldots, p\}, \quad Y_{x^{(1)}, \alpha^{(1)}} I_{p} = \rho^{(1)} Y_{x^{(2)}, \alpha^{(2)}} I_{p}. \]  

(A8)

By uniqueness of the eigenvalues, \( \alpha^{(1)} = \alpha^{(2)} \) and

\[ Y_{x^{(1)}, \alpha^{(1)}} I_{p} = \text{Vdm}(x^{(1)})^{-1} Z_{0} = \text{Vdm}(x^{(2)})^{-1} Z_{0} = Y_{x^{(2)}, \alpha^{(2)}} I_{p}, \]

which implies that, for all \( q \in \{1, \ldots, p\} \),

\[ \sum_{\ell=1}^{p} \frac{\sigma_{q}^{(1)} \sigma_{q}^{(1)}}{1 - \alpha_{q}^{(1)} \alpha_{q}^{(1)}} = \sum_{\ell=1}^{p} \frac{\sigma_{q}^{(2)} \sigma_{q}^{(2)}}{1 - \alpha_{q}^{(2)} \alpha_{q}^{(2)}} = 0. \]  

(A11)

This can be stated as follows. For all \( q \in \{1, \ldots, p\} \),

\[ 0 \leq \frac{\sigma_{q}^{(2)} \sigma_{q}^{(2)}}{\sigma_{q}^{(2)} \sigma_{q}^{(2)}} - 1 \sum_{\ell=1}^{p} \frac{\sigma_{q}^{(1)} \sigma_{q}^{(1)}}{1 - \alpha_{q}^{(2)} \alpha_{q}^{(2)}} \]

implying that, for all \( q \in \{1, \ldots, p\} \),

\[ \frac{\sigma_{q}^{(2)} \sigma_{q}^{(2)}}{\sigma_{q}^{(2)} \sigma_{q}^{(2)}} \geq 1. \]

Consequently, when writing (A11) for \( q = m \), we get a sum of non-negative terms being equal to 0. This shows that each term

\[ \frac{\sigma_{q}^{(1)} \sigma_{q}^{(1)}}{1 - \alpha_{q}^{(2)} \alpha_{q}^{(2)}} = 1, \quad \ell \in \{1, \ldots, p\} \]

is null, i.e.

\[ \forall \ell \in \{1, \ldots, p\}, \quad \frac{\sigma_{q}^{(2)} \sigma_{q}^{(2)}}{\sigma_{q}^{(2)} \sigma_{q}^{(2)}} = 1. \]

For \( \ell = m \) we get \( \sigma_{m}^{(1)} = \sigma_{m}^{(2)} \) and finally \( \alpha^{(1)} = \alpha^{(2)} \). Thus, (iii) is shown.

It remains to prove that \( \{Z_{0}, \ldots, Z_{p-1}\} \) is a base of \( \mathbb{R}^{p} \). To that purpose, we only need to show that this family of \( p \) vectors in \( \mathbb{R}^{p} \) is linearly independent. For all linear combinations of this family associated with the weights \( \lambda = (\lambda_{0}, \ldots, \lambda_{p-1})^{T} \), we have

\[ \sum_{k=0}^{p-1} \lambda_{k} Z_{k} = \text{Vdm}(x^{(0)}) \text{diag}(Y_{x^{(0)}, \alpha^{(0)}} I_{p}) \times \left( \sum_{k=0}^{p-1} \lambda_{k} [\alpha_{k}^{(0)}]^{k}, \ldots, \sum_{k=0}^{p-1} \lambda_{k} [\alpha_{k}^{(0)}]^{k} \right)^{T}. \]

Furthermore, the components of \( Y_{x^{(0)}, \alpha^{(0)}} I_{p} \) are all positive, and \( \alpha_{k}^{(0)} \neq \alpha_{k}^{(0)} \) for \( j \neq k \) implying that \( \text{Vdm}(x^{(0)}) \) and \( \text{diag}(Y_{x^{(0)}, \alpha^{(0)}} I_{p}) \) are invertible. The linear independence is then deduced from

\[ \sum_{k=0}^{p-1} \lambda_{k} Z_{k} = 0 \iff \forall j \in \{1, \ldots, p\}, \]

\[ \sum_{k=0}^{p-1} \lambda_{k} [\alpha_{k}^{(0)}]^{k} = 0 \iff \lambda^{T} \text{Vdm}(x^{(0)}) = 0 \iff \lambda = 0, \]

which concludes the proof. \( \square \)
A.2. Complete data likelihood

Proof of proposition 3.1: $p(\mathbf{Y}_{1:T}, \mathbf{X}_0, \tilde{\mathbf{X}}_{1:T})$ can be decomposed as follows:

$$p(\mathbf{Y}_{1:T}, \mathbf{X}_0, \tilde{\mathbf{X}}_{1:T}) = p_0(\mathbf{X}_0) \prod_{k=1}^T p_\theta(Y_k|\mathbf{X}_0, \tilde{\mathbf{X}}_{1:k}) p_\theta(\tilde{\mathbf{X}}_k|\mathbf{X}_0, \tilde{\mathbf{X}}_{1:k-1}),$$  

(A12)

with the convention $p_\theta(\tilde{\mathbf{X}}_1|\mathbf{X}_0, \tilde{\mathbf{X}}_{1:0}) = p_\theta(\tilde{\mathbf{X}}_1|\mathbf{X}_0)$. The explicit computation of (A12) is obtained from the following conditional laws. From (A2), we have for all $k \geq 1$,

$$Y_k|\mathbf{X}_0, \tilde{\mathbf{X}}_{1:k} 
\sim \mathcal{N}\left(\beta \rho \tilde{\mathbf{X}}_{k-1/2}, \omega_{k-1}\left[\tilde{\mathbf{X}}_k - \langle \mathbf{a}', \mathbf{X}_0 \rangle\right], \rho^2[1 - \rho^2]e^{\tilde{\mathbf{X}}_{k-1}}\right).$$

Appendix B: Additional graphs

Figure B1. Calibration of setting 1 for $p \in \{2, 3\}$.

Figure B2. Calibration of setting 2 for $p \in \{1, 3\}$.
Figure B4. Calibration of CAC 40 for $p \in \{1, 2, 3\}$.

Figure B3. Calibration of setting 3 for $p \in \{1, 2\}$. 

Calibrating the exponential Ornstein–Uhlenbeck multiscale
Figure B5. Calibration of Dow Jones for $p \in \{1, 2, 3\}$. 