Working Paper Series
Department of Economics
University of Verona

Computing option values by pricing kernel with a stochatic volatility model

Silvia Centanni

WP Number: 5 April 2011

ISSN: 2036-2919 (paper), 2036-4679 (online)

Computing option values by pricing kernel with a stochatic volatility model

Silvia Centanni
Department of Economic Sciences, University of Verona
Ex Caserma Passalacqua - Palazzina 32, 37129 Verona, Italy
email: silvia.centanni@univr.it

Andrea Ongaro
Cattolica Assicurazioni
Lungadige Cangrande 16, 37126 Verona, Italy
email: andrea.ongaro@cattolicaassicurazioni.it

Abstract

To use a wider range of information available on the market, we propose a parameter estimation and option pricing procedure which involves a two step approach: in a first step real world parameters are estimated from time series data of the underlying financial asset, and in a second step the so called pricing kernel is computed from option data. For the first step we compare two likelihood based estimation procedures, namely the particle filter and the SEM algorithms. For the second step we use an adapted version of the so called asset specific pricing kernel. The results are then analyzed in a simulation study and implemented in a real dataset of the FTSE Mib Index, and compared with the classical calibration approach, which makes use of the option data only.

Key Words

Option pricing, Stochastic discount factor, Heston model, Particle Filter, Markov chain Monte Carlo, Expectation maximization.

Address for correspondence: Silvia Centanni, Department of Economic Sciences, University of Verona, Viale dell'Università 3, 37129 Verona, Italy, Tel. +39 045 84 25 460 (silvia.centanni@univr.it)

1 Introduction

Option pricing is one of the most known and widely investigated matters of quantitative finance. The classical and probably most famous approach to option pricing is the Black and Scholes (1973) model. This approach presents several advantages, mostly due to its simplicity since it presents constant parameters, and leads to the widely used closed-form formula for options evaluation. Nevertheless it is well known that it presents also many disadvantages. The strong assumptions of constant volatility and log-normal returns make the Black-Scholes model inappropriate to describe empirical financial data, because it fails to catch many stylized facts observed in option data, such as volatility smiles. Several improvements of the Black-Scholes model have been proposed in the literature. One of the most interesting developments allows the volatility term of the stock price dynamics to vary along time, often following a stochastic process itself. In particular, in the Heston (1993) model the constant volatility is replaced by a stochastic CIR-type process, and in Bates (1996) an additional jump component is considered.

A basic result of mathematical finance states that for a stochastic process S, representing the stock price, the existence of an equivalent martingale measure, i.e. a measure equivalent to the natural measure P such that the discounted price process is a martingale, is essentially equivalent to absence of arbitrage opportunities (see, for example, Harrison and Kreps (1979) and Delbaen and Schachermayer (1994)). If the price of the risky asset follows a stochastic volatility process, the market model is in general incomplete and it can be shown that there exist more than one of such measures. With this martingale approach, the pricing problem of a contingent claim is reduced to taking the expected value under the 'right' measure (among all existing martingale measures) of the discounted payoffs. Alternatively, another approach can be taken into consideration: it can be shown that for each one of all equivalent martingale measures it can be found a stochastic process Y called pricing kernel or stochastic discount factor (SDF), such that $Y \cdot S$ is a local martingale under the natural probability P (see, for example, Kardaras (2010)). The SDF is related to the Radon-Nykodim derivative of the pricing martingale measure with respect to P. Thus, choosing an EMM for pricing purposes is equivalent to choose a SDF, and contingent claim prices can be computed by taking the expected value under the natural probability P of the payoff discounted with the stochastic discount factor.

From an applicative point of view, once a model for the dynamics of the underlying asset has been chosen, the next step is to compute option prices by some numerical procedure, which in general involves the estimation of the parameters of the model, based on data available on financial markets. The classical option pricing technique relies on risk-neutral parameters estimation, which is based on currently observed option prices. Risk-neutral calibration of the parameters using option prices available on financial markets is addressed using an optimization procedure, that is, looking for the set of parameters which allows for the best fit to the observed option prices. As an alternative, it is possible to perform the estimation of real-world parameters (i.e. under the natural measure P) and then use the pricing kernel for pricing purposes. This alternative approach relies on the use of the time series of the underlying asset during the first step, i.e. the real-world parameters estimation, and on the use of market data such as option prices to estimate the pricing kernel in the second step.

It has to be considered that the time series of the underlying could contain useful information that would not be used under the classical approach. This information would be important for example for 'out of the sample' pricing, in particular in a situation with poor information on options similar to those to be evaluated. Furthermore, the classical approach requires to perform calibration on a daily basis, as there is not time consistency between data. Converserly, using the time series of the underlying asset allows to make statistical inference on the estimates obtained. At last, real-world estimates could be directly used for risk management purposes, which relies on actual probabilities, as well as for volatility arbitrage trading.

The main problem that arises when trying to perform such an estimation in the Heston model framework is the unobserved stochastic volatility process. For a review of the most used techniques for the treatment of stochastic volatility see, for example, Javaheri (2005). Here we will use two very flexible techniques, namely Markov chain Monte Carlo (MCMC) together with the stochastic expectation maximization(SEM) algorithm, and particle filter (PF). For a general description of MCMC see for example Andrieu et al. (2003) or Besag (2001), while a detailed reference for particle filter is Doucet et al. (2001). These techniques use a different set of information to obtain the estimates at each time step t. While MCMC uses all the information available at final time T to get the estimate of the entire volatility path, PF uses only information up to time t to estimate the volatility in t, i.e. performs 'online' estimation.

As far as the pricing kernel is concerned, different types of shapes are provided in the literature. The most flexible characterization is probably the semi non-parametric approach of Gallant and Tauchen (2009), which has been applied in different contexts. In the current work a parametric pricing kernel, the Asset Specific Pricing Kernel (APSK) firstly proposed in Lüders and Franke (2004) will be used, because of its tractability and interesting features. Düring (2009) provides a closed form solution for call options, based on an Heston-type behaviour of the underlying asset and the same ASPK form of Lüders and Franke (2004). However, because of the use he makes of the pricing kernel, he consideres a driftless information process. Conversely, in our context we want to estimate the pricing kernel which brings to market consistent option prices given an estimated real world drift. Furthermore, it has to be considered that this work assumes a different framework. While in Düring (2009) different elasticity parameters are used for simulation, here the parameters will be estimated using option data.

The paper is organised as follows. In Section 2 the stochastic volatility, and particularly the Heston model is presented. In section 3 the classical calibration of the parameters, which make use of the fast Fourier transform method, is applied to option pricing. Section 4 is devoted to the description of our estimation and pricing procedure. Finally, in Section 5, different empirical results are presented, based on simulation studies and on an application to a real data set.

2 Stochastic volatility and the Heston model

We model the dynamics of a financial asset with a stochastic process (on a suitable filtered probability space), which is defined to be the solution of the following stochastic differential equations (SDE)

$$dS(t) = \mu(S_t, v_t, t)dt + \sigma(S_t, v_t, t)dB_t \tag{2.1}$$

$$dv(t) = \alpha(S_t, v_t, t)dt + \beta(S_t, v_t, t)dZ_t, \tag{2.2}$$

where μ, σ, α, β are functions such that the solution of (2.1) and (2.2) exists and is unique, and where B and Z are correlated Brownian motions with correlation coefficient ρ.

The Heston model (Heston, 1993) is one of the most popular stochastic volatility models for the description of the dynamics of stock prices. The stochastic differential equations which characterize the process are defined through a particular choice of the functions μ, σ, α, β, that is

$$dS_t = \mu S_t dt + \sqrt{v_t} S_t dB_t \tag{2.3}$$

$$dv_t = \kappa \left[\theta - v_t\right] dt + \xi \sqrt{v_t} dZ_t, \tag{2.4}$$

with $\mu \in \mathbb{R}$ and $\kappa, \theta, \xi > 0$. For simplicity we assume a constant interest rate r, therefore, the price at time t of a discount bond with maturity $t+\tau$ is given by $P(t, t+\tau) = e^{-r\tau}$. The SDE in (2.4) can be recognized to be the familiar square-root process used by Cox, Ingersoll, and Ross (1985). The variance drifts toward a long-run mean of θ, with mean-reversion speed determined by κ. When mean reversion is positive, the variance has a steady-state distribution (see Cox et al. (1985)) with mean θ. The parameter ξ controls the volatility of volatility. When a is zero, the volatility is deterministic, and continuously compounded spot returns have a normal distribution, otherwise κ increases the kurtosis of spot returns. This is an important aspect, since skewness and kurtosis effects are observed in practice. The assumptions in Section 2 are not enough to price contingent claims because there is no assumption that gives the so called 'price of volatility risk'. As suggested in Bakshi et al. (1997), one could parametrize equations (2.3) and (2.4) directly under a risk neutral measure and risk premia can be internalized in the stochastic structure; therefore the price of volatility risk is implicitly reflected in the parameters in equation (2.4). Indeed in this case, for example, the implied variance θ may not equal the variance of spot returns given by the 'true' process; furthermore the drift parameter μ is replaced by the riskless rate r. According to this approach, the price of a european call option can be obtained by

$$C = e^{-qT} S \Pi_1 - e^{-rT} K \Pi_2, \tag{2.5}$$

where S is the current value of the underlying, K is the strike price, r is the risk free rate , q is the dividend yield and T is time to maturity. Π_1 and Π_2 are defined as

$$\Pi_1 = \frac{1}{2} + \frac{1}{\pi} \int_0^\infty \mathrm{Re}\left[\frac{e^{-i\omega \ln(K)} \phi_T(\omega - i)}{i\omega \phi_T(-i)}\right] d\omega \tag{2.6}$$

$$\Pi_2 = \frac{1}{2} + \frac{1}{\pi} \int_0^\infty \mathrm{Re}\left[\frac{e^{-i\omega \ln(K)} \phi_T(\omega)}{i\omega}\right] d\omega, \tag{2.7}$$

where ϕ_T is the characteristic function of the logarithm of the stock price.

3 Pricing using risk neutral parameter calibration

Once formulas (2.5)–(2.7) have been obtained, as suggested in Bakshi et al. (1997), one could calibrate the parameters of the model by obtaining the values implied by option prices available on the market. When the price of a non-traded option is needed, one simply applies formulas (2.5)–(2.7) to the corresponding payoff, using the calibrated values for the parameters. For the computation of the theoretical option prices we adopt the fast Fourier transform (FFT) technique proposed in Carr and Madan (1998), using the MATLAB code provided in Gilli and Schumann (2010). Risk-neutral calibration of the parameters using option prices available on financial markets is addressed using an optimization procedure, that is, looking for the set of parameters which allows for the best fit to the observed option prices. To perform the calibration, it is therefore necessary to choose the set of options to be used within the optimization and to define the pricing error that will be minimized; then the quality of the estimation depends on the optimization algorithm, which must be able to perform a global optimization, avoiding to get stuck in local minima. Different measures of pricing error can be taken to calibrate the parameters. For example, we can compute the sum of the square differences between the observed option prices and the theoretical option prices, which are calculated using the model with the current parameter values. An alternative way to measure the discrepancy between observed and theoretical option prices is the relative error (Gilli and Schumann, 2010) and

$$\min \sum_{i=1}^{N} \frac{|C_i^{\text{model}} - C_i^{\text{market}}|}{C_i^{\text{market}}} \tag{3.1}$$

can be taken as the objective function to be minimized. The choice of the algorithm which is used to minimize the pricing error is crucial for the calibration results. Here we use the Genetic Algorithm (GA) contained in MATLAB, which can be viewed as a variant of the so called Differential Evolution (DE) algorithm proposed in Storn and Price (1997). Although it differs slightly from the algorithm which is used in our empirical study, we recall here the DE algorithm, since it offers the advantage of higher simplicity. DE takes an initial population of n_p individuals, where each individual is a vector of length p that represent a possible solution to the minimization problem. At each generation k, the algorithm creates a new candidate solution for each existing one. If the value of the objective function evaluated at the new candidate is less than the value computed in the original solution the new candidate is accepted, otherwise the old solution is kept. The creation of new candidates is based upon a two step procedure. First of all, for each existing solution a mutated vector is constructed by taking the weighted difference between two other solutions and adding it to a third solution. Then an element-wise crossover takes place between the original solutions and the mutated vectors. The general structure based on mutation and crossover is kept in the GA contained in MATLAB, which is characterized by a higher speed due to efficient computation.

4 Pricing using the stochastic discount factor

Apart from risk-neutral valuation, an alternative way to approach option pricing is obtained observing that the fundamental asset pricing equation states that in an arbitrage free market the price of a contingent claim D_t is given by the expected discounted payoff $H(S_T)$ at maturity T, where the expectation is taken under some equivalent martingale measure Q. This last measure Q is defined by its Radon-Nikodym derivative process with respect to the phisical probability, and we can write

$$D_t = E^Q\left(e^{-r(T-t)}H(S_t)|\mathcal{F}_t\right) = E\left(H(S_T)\phi_{t,T}|\mathcal{F}_t\right), \tag{4.1}$$

where $\phi_{t,T}$ is the pricing kernel, and where $\mathcal{F}_t$ represents the information up to time t. Once a form for $\phi_{t,T}$ has been chosen, to compute the price of a non-traded option a two step trocedure can be implemented. In the first step real word parameters of the dynamics of the underlying asset are estimated from historical data. Then, in the second step, the pricing kernel is estimated using available option prices, and the expected value in (4.1) can be computed with the corresponding payoff.

4.1 Parameter estimation with historical data

The main issue in real world parameters estimation is represented by the presence of stochastic volatility, since the only available information for the estimation is the set of discrete observations of the stock price. Here we choose to apply two different likelihood based techniques, namely a stochastic version of the expectation-maximization (SEM) algorithm, which makes use of Markov chain Monte Carlo techniques, and particle filters.

Markov chain Monte Carlo techniques are designed for sampling from a desired probability distribution π when it is not possible to sample from it directly. This class of algorithms is based on the construction of a Markov chain with π as its limit distribution. Supposing that the parameters are known, the aim is to obtain samples from the conditional distribution of the the volatility process v_t given the observed data. The SEM algorithm is an iterative likelihood based procedure for parameter estimation, which starts from the current guess of the value of the parameters, and alternates a simulation step (which makes use of the MCMC) to obtain samples from the conditional probability of the unobserved volatility process, given the wholw set of observations, an expectation step to find the complete likelihood, and a maximization step of the likelihood function to find the new parameter values.

Particle filter, also known as *sequential importance sampling*, is a technique for the evaluation of the conditional distribution of an unobservable process, given the observation up to the evaluation time, by obtaining a sample from the desired distribution and using the sampled values as the support of a discrete approximating distribution. The main difference with the MCMC algorithm, where the complete information between time 0 an time T is used to get an estimate on the entire volatility path v, is that only the information between time 0 and time t is used to estimate v_t. For this reason it can be viewed as an 'online' filter and can be implemented in such a way that new data arriving in the market can be used to improve the current estimation. In the general particle filter framework, the parameters of the model are considered as known,but in our case we need a joint estimation of the volatility path and the model parameters. This problem has been addressed for example in Aihara et al. (2008), who use a modification of the particle filter based upon the *augmented state* technique. Here, adapting the algorithm presented in Spall (1998) and applied to particle filtering in Yang et al. (2008), we apply the simultaneous perturbation stochastic approximation (SPSA) to get the joint estimation of the volatility and parameters.

4.1.1 Particle Filters

Given a a state space model of the form

$$v_t \sim p(v_t|v_{t-1};\Psi), \tag{4.2}$$

$$y_t \sim p(y_t|v_t;\Psi) \tag{4.3}$$

where v_t is unobserved state vector at time t, y_t is an observation at time t and Ψ is the set of unknown parameters, at each time step t we want to get an estimate of the conditional distribution $p(v_t|y_{0:t};\Psi)$. To do this, a set of samples, called *particles* are drawn from a proposal distribution and then a weight is associated to each of them. The obtained particles, together with their weight, constitutes a discrete approximation of the desired conditional distribution. In our case the vector $y_{0:t}$ represents the observed daily logreturns until time t, while v_t is the unobserved volatility term in at time t. At time $t = 0$ a set of N particles is initialized, each with initial weight $1/N$. Then for each time step t between 1 and T, v_t is simulated sequentially from some proposal distribution $\pi(v_t|y_{0:t})$. Then importance weights w_t are computed (and then normalized) as

$$w_t = w_{t-1}\frac{p(y_t|v_t;\Psi)p(v_t|v_{t-1};\Psi)}{\pi(v_t|v_{0:t-1},y_{0:t};\Psi)}\ , \tag{4.4}$$

since the state evolution is a Markov process and the proposal distribution factorise as

$$\pi(v_{0:t}|y_{0:t};\Psi) = \pi(v_0;\Psi)\prod_{k=1}^{t}\pi(v_k|v_{0:k-1},y_{0:k};\Psi).$$

In the SPSA applied to particle filter, the dynamic state v is evaluated by particle filtering while the (static) parameters are estimated by a recursive maximum likelihood (ML) optimization which allows an update of the parameter values with the arrival of new informations. Given a set of observations $y_{0:t}$, the ML estimation requires to perform a maximization with respect to the parameter vector. The objective function to be maximized is

$$\begin{aligned} p(y_{0:t},\Psi) &= p(y_t|y_{0:t-1};\Psi)p(y_{0:t-1};\Psi) \\ &= p(y_0;\Psi)\prod_{k=1}^{t}p(y_k|y_{0:k-1};\Psi) \end{aligned} \tag{4.5}$$

where

$$\begin{aligned} p(y_k|y_{0:k-1};\Psi) &= \int p(y_k|v_k;\Psi)p(v_k|y_{0:k-1};\Psi)dv_k \\ p(y_0;\Psi) &= \int p(y_0|v_0;\Psi)p(v_0;\Psi)dv_0\ . \end{aligned}$$

For computational reasons, Yang et al. (2008) choose the predicted likelihood as cost function, i.e.

$$f(\Psi) = p(y_t|y_{0:t-1};\Psi) = \int p(y_t|v_t;\Psi)p(v_t|y_{0:t-1};\Psi)dv_t \tag{4.6}$$

This choice allows for fast computation, which is an useful feature of particle filter technique.

The problem of maximizing the objective function can be translated, under regularity conditions, into finding the zeros of the gradient $\nabla f(\Psi)$. In SPSA it is not necessary to be able to perform a direct measurement or calculation of the gradient, since at each recursion to estimate Ψ^* we proceed computing

$$\Psi_t = \Psi_{t-1} + a_t \hat{\nabla} f(\Psi_{t-1}) \tag{4.7}$$

where $\hat{\nabla} f(\Psi_{t-1})$ is the numerical approximate evaluation of the gradient at Ψ_{t-1} and a_t is a sequence of decreasing step-size such that $a_t \to 0$, and $\sum_{t=1}^{\infty} a_t = \infty$, and where $\hat{\nabla} f(\Psi_{t-1})$ is a gradient approximation. This approximation is achieved by a simultaneous random variation of all elements of Ψ to obtain two evaluations of the objective function, that is, the gradient approximation

$$\hat{\nabla} f(\Psi_{t-1}) = (\hat{\nabla} f_1(\Psi_{t-1}), \ldots, \hat{\nabla} f_m(\Psi_{t-1}))$$

is given by

$$\hat{\nabla} f_j(\Psi_{t-1}) = \frac{\hat{f}(\Psi_{t-1} + c_t \Delta_{t,\cdot}) - \hat{f}(\Psi_{t-1} - c_t \Delta_{t,\cdot})}{2 c_t \Delta_{t,j}} \tag{4.8}$$

where c_t denotes a sequence of positive scalars decreasing to 0 and Δ_t is a random perturbation vector.

To address the problem of degeneracy of the particles, which is often encountered with particles filters, a resampling step is inserted in the algorithm, in which a new set of particles is sampled from a distribution given by the existing set of particles with associated weights, only if the effective sample size (the number of particles having a weight which is sufficiently different from zero) goes below a given treshold, empirically selected.

As far as the practical implementation of the algorithm in the case of Heston model is concerned, all the above quantities have to be detailed. First of all, as in Aihara et al. (2008), we observe that the stochastic differential equation for $y_t := \ln(S_t/S_0)$ is

$$dy_t = (\mu - \frac{1}{2} v_t) dt + \sqrt{v_t} dB_t \ , \tag{4.9}$$

and the SDE for the volatility process is as in (2.4). Setting

$$\tilde{Z}_t = \frac{1}{\sqrt{1-\rho^2}} (Z_t - \rho B_t), \tag{4.10}$$

it follows that $\tilde{Z}_t$ is independent of B_t. Moreover, from (4.10) we obtain

$$dZ_t = \sqrt{1-\rho^2}d\tilde{Z}_t + \rho dB_t = \sqrt{1-\rho^2}d\tilde{Z}_t + \frac{\rho}{\sqrt{v_t}}(dy_t - (\mu - \frac{1}{2}v_t)dt) \ ,$$

hence we have

$$dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t}\sqrt{1-\rho^2}d\tilde{Z}_t + \xi\rho(dy_t - (\mu - \frac{1}{2}v_t)dt) \tag{4.11}$$

Now, considering the Euler discretization of (4.11)

$$\begin{aligned} v_t = v_{t-1} + \kappa(\theta - v_{t-1})\Delta t - \xi\rho(\mu - \frac{1}{2}v_{t-1})\Delta t \\ + \xi\sqrt{v_{t-1}}\sqrt{1-\rho^2}\Delta\tilde{Z}_t + \xi\rho(y_t - y_{t-1}), \end{aligned} \tag{4.12}$$

we find that the required probabilities to implement the particle filter with SPSA are given by:

- The likelihood of the observations is given by

$$\begin{aligned} p(y_t|v_{0:t}, y_{0:t-1}; \Psi) &= p(y_t|v_t, v_{t-1}, y_{t-1}; \Psi) \\ &= \mathcal{N}(y_{t-1} + (\mu - \frac{1}{2}v_t)\Delta t, \sqrt{v_{t-1}}\sqrt{\Delta t}) \ . \end{aligned} \tag{4.13}$$

- For the probability $p(v_t|v_{0:t-1}, y_{0:t-1}; \Psi)$, we have

$$p(v_t|v_{0:t-1}, y_{0:t-1}; \Psi) = p(v_t|v_{t-1}; \Psi) = \mathcal{N}(\tilde{m}(v_{t-1}), \tilde{\sigma}(v_{t-1})) \tag{4.14}$$

  where

$$\tilde{m}(v_{t-1}) = (1 + \frac{1}{2}\xi\rho\Delta t)^{-1}\left\{v_{t-1} + \kappa(\theta - v_{t-1})\Delta t + \frac{\xi\rho}{2}v_{t-1}\Delta t\right\}$$

  and

$$\tilde{\sigma}(v_{t-1}) = (1 + \frac{1}{2}\xi\rho\Delta t)^{-1}\xi\sqrt{v_{t-1}}\sqrt{\Delta t}.$$

  To get this,

$$dy_t = (\mu - \frac{1}{2}v_t)dt + \sqrt{v_t}dB_t$$

  has been substituted into (4.11).

- It remains to choose the proposal distribution $\pi(v_t|v_{t-1}, y_t; \Psi)$. Following Aihara et al. (2008) we set

$$\pi(v_t|v_{t-1}, y_t; \Psi) = p(v_t|v_{t-1}, y_t; \Psi) = \mathcal{N}(m(v_{t-1}, y_t), \sigma(v_{t-1})) \tag{4.15}$$

where

$$\begin{aligned} m(v_{t-1}, y_t) = {} & v_{t-1} + \kappa(\theta - v_{t-1})\Delta t \\ & - \xi\rho(\mu - \frac{1}{2}v_{t-1})\Delta t + \xi\rho(y_t - y_{t-1}) \end{aligned}$$

and

$$\sigma(v_{t-1}) = \xi\sqrt{v_{t-1}}\sqrt{1-\rho^2}\sqrt{\Delta t}\ .$$

This choice for π allows to benefit from the additional information contained in y_t when sampling v_t. Furthermore, as shown in Doucet et al. (2001), it is optimal in the sense that it minimizes the variance of the importance weights. However in our context this choice is not obvious, since the perturbation of the parameters could get the probability too close to zero when evaluating importance weights. An alternative is to choose $\pi(v_t|v_{t-1}, y_t; \Psi) = p(v_t|v_{t-1}; \Psi)$, so that the incremental weight reduces to $p(y_t|v_t, v_{t-1}, y_{t-1}; \Psi)$. We implemented the algorithm using both these specifications and, as we will show in Section 5, results obtained using the algorithm with the optimal choice for π, has a better performance in our applications.

Now, turning to the choice of the gain sequences c_t and a_t and of perturbation vector, the choice is to set

$$c_t = \frac{c}{(t+1)^\gamma},$$

where c and γ are appropriately set constants. A general rule is to set c at a level approximately equal to the standard deviation of the measurement noise in $f(\Psi)$. As suggested in Regarding the parameter γ, a practically effective value for it is 0.101, although the asymptotically optimal value is 1/6. Furthermore, we have

$$a_t = \frac{a}{(A+t+1)^\alpha},$$

where a, A and α are constants. The practically effective value for α which is proposed in Spall (1998) is 0.602, while the asymptotically optimal value is 1.0. Once these values are set, the choice of the values for the parameters a and A is an empirical task, based upon the desidered magnitude of the parameters perturbation and the number of available steps.

The conditions that the random perturbation vector Δ_t must satisfy are provided in Spall (1992). Here we follow the simple choice in Spall (1998), i.e. to sample each component of Δ_t from a Bernoulli ±1 distribution with probability $\frac{1}{2}$ for each ±1 outcome.

4.1.2 The MCMC algorithm and the Stochastic Expectation-Maximization

The Markov chain Monte Carlo algorithm is a simulation method that supplies a (non-necessarily independent) sample from a target distribution $\pi(dx)$, by simulating a trajectory from a Markov chain having π as its stationary distribution. To construct such a (homogeneous) Markov chain, we need to specify the transition probabilities from one state x to the next state x' . This can be done by choosing an aperiodic and irreducible transition kernel $P(x, dx')$. In order for π to be the stationary distribution of the chain, the transition kernel must satisfy the detailed balance condition:

$$\int\int \pi(dx)P(x, dx') = \int\int \pi(dx')P(x', dx)$$

(see Besag (2001) for a classical review on the subject and for further references). For instance, for a chain $(X_i)_{i\in\mathbb{N}}$ having a finite state space, $P(x, dx')$ is the transition probability $P(X_{i+1} = x'|X_i = x)$ from state x to state x' , and the detailed balance condition reduces to

$$\pi(X_i = x)P(X_{i+1} = x'|X_i = x) = \pi(X_i = x')P(X_{i+1} = x|X_i = x').$$

After a sufficiently long run (called burn-in) of the chain, the subsequent realized values of the simulated trajectory can be viewed as an approximate (dependent) sample from π . For the construction of the chain the MetropolisHastings algorithm proceeds sweeping around all the variables $x_1, \ldots, x_n$, visiting subsets of them in turn, either randomly or systematically. When a subset x_M of $\{x_1, \ldots, x_n\}$ is visited, the variables in it are updated. New values for these variables are proposed from an essentially arbitrary distribution $q_M(x'_M; x)$ and the proposed values are accepted with probability $min\{1, A(x, x')\}$, where

$$A(x, x') = \frac{\pi(x'_M|x^C_M)q_M(x_M; x')}{\pi(x_M|x^C_M)q_M(x'_M; x)}$$

and x^C_M denotes the non-visited variables; otherwise the existing values are retained. In practical implementations, suppose that the distribution π is a conditional distribution of a vector of some unobserved variable v, given an observed vector y, that it is characterized by proper densities $\pi(v|y)$, that the current state of the chain is v, and that we have defined one or more move types, indexed with (m), from the current state to the following one. Then, to update the chain, the MCMC algorithm can be structured as follows:

1. Choose to perform a move of type (m) with probability $p(m)$.
2. Generate a random vector u from a specified proposal density $q(u|m, v)$.
3. Accept $v' = u$ as the new state of the chain with probability $min\{1, A_m(v, v')\}$, where the acceptance ratio $A_m(v, v')$ is given by

$$\begin{aligned} A_m(v, v') &= \frac{\pi(v'|y)}{\pi(v|y)} \frac{p(m')q(u'|m', v')}{p(m)q(u|m, v)} \\ &= \frac{P(y|v')}{P(y|v)} \cdot \frac{P(v')}{P(v)} \cdot \frac{R(v', v)}{R(v, v')}, \end{aligned} \tag{4.16}$$

where $R(v, v') = p(m)q(u|m, v)$ and the proposal moves must satisfy $R(v, v') > 0 \Rightarrow R(v', v) > 0$. Borrowing from a Bayesian terminology, we call the ratios in (4.16) the likelihood ratio, the prior ratio and the proposal ratio respectively. In our model, the vector v of unobserved variable is given by the values of the volatility process at time $t = 1, 2, \dots, T$, and the observations are given by the available time series of daily logprices. In order to specify the algorithm, we need to detail the proposal moves and the probabilities characterizing the ratios in (4.16).

First of all, to determine the probability $P(y|v)$ we follow Forbes et al. (2002), which propose an Euler discretization of the Heston stochastic differential equations (using the log-price y), namely

$$y_t = y_{t-1} + \left(\mu - \frac{1}{2}v_{t-1}\right)\Delta t + \sqrt{v_{t-1}}\sqrt{\Delta t}\epsilon_{1t} \tag{4.17}$$

$$v_t = v_{t-1} + \kappa(\theta - v_{t-1})\Delta t + \xi\sqrt{v_{t-1}}\sqrt{\Delta t}\epsilon_{2t} \tag{4.18}$$

with

$$\epsilon_{1t}, \epsilon_{2t} \sim N(0, 1), \qquad \text{corr}(\epsilon_{1t}, \epsilon_{2t}) = \rho, \qquad t = 2, 3, \dots, T.$$

Defining

$$\tilde{\epsilon}_{1t} = \frac{1}{\sqrt{1 - \rho^2}}(\epsilon_{1t} - \rho\epsilon_{2t}), \tag{4.19}$$

then $\tilde{\epsilon}_{1t}$ is independent from ϵ_{2t}, and $\epsilon_{1t} = \tilde{\epsilon}_{1t}\sqrt{1 - \rho^2} + \rho\epsilon_{2t}$. Now, substituting ϵ_{2t} in (4.18), we obtain

$$\epsilon_{1t} = \tilde{\epsilon}_{1t}\sqrt{1 - \rho^2} + \rho\left(\frac{v_t - v_{t-1} - \kappa(\theta - v_{t-1})\Delta t}{\xi\sqrt{v_{t-1}}\sqrt{\Delta t}}\right)$$

Finally, substitute this into 4.17 and, after some algebra, we obtain the following likelihood for the path $y_t, \quad t = 0, \ldots, T$:

$$\begin{aligned} P(y|v,\Psi) = (2\pi)^{-(n-1)/2}[(1-\rho^2)\Delta t]^{-(n-1)/2} \cdot \prod_{t=2}^{T} \frac{1}{\sqrt{v_{t-1}}} \\ \exp\left\{ -1/[2(1-\rho^2)\Delta t] \sum_{t=2}^{T} \left(\frac{y_t - \mu_{y_t,v_t}}{\sqrt{v_{t-1}}} \right)^2 \right\} \end{aligned} \tag{4.20}$$

where

$$\mu_{y_t,v_t} = y_{t-1} + [\mu - \frac{1}{2}v_{t-1}]\Delta t + \frac{\rho}{\xi}\left(v_t - [\theta\kappa\Delta t + (1-\kappa\Delta t)v_{t-1}]\right). \tag{4.21}$$

For the prior ratio, according to the Euler discretization scheme we have selected, the probability $P(v)$ is

$$\begin{aligned} P(v) = P(v_1)\prod_{t=2}^{T} P(v_t|v_{t-1}) = \\ = P(v_1)\prod_{t=2}^{T} \frac{1}{\sqrt{2\pi\sigma_{v_t}^2}} \exp^{-\frac{(v_t-\mu_{v_t})^2}{2\sigma_{v_t}^2}}, \end{aligned} \tag{4.22}$$

where

$$\mu_{v_t} = v_{t-1} + \kappa(\theta - v_{t-1})\Delta t \tag{4.23}$$

and

$$\sigma_{v_t} = \xi\sqrt{v_{t-1}}\sqrt{\Delta t}\ . \tag{4.24}$$

We choose for $P(v_0)$ a gamma distribution with given parameters.

As far as the proposal probability $R(v, v')$ is concerned, we adopted one set of moves, in which the proposed new path of the volatility is obtained changing the current path in a randomly selected time interval:

- Draw an integer $a \in (1, \ldots, T)$ (starting time);

- Draw an integer $c \in (1, \ldots, T/10)$ (final time);

- Change the volatility path from the starting time to time b, where $b = \max(a+c, T)$, drawing from $P(v_t|v_{t-1})$, $t = a, \ldots, b$.

With this move type we obtain a proposal probability given by

$$R(v, v') = P(a)P(b)\prod_{t=a}^{b} P(v_t|v_{t-1}).$$

Having detailed the MCMC algorithm, now we turn to the use of the SEM algorithm in order to estimate the parameters of the model. The EM algorithm (Dempster et al., 1977) is an iterative scheme devised for the numerical evaluation of maximum likelihood estimates when some of the observations are missing. Given a density function $f(\cdot;\Psi)$ depending on a vector Ψ of unknown parameters, and a data set y drawn from this distribution, the problem is to find the value of Ψ which maximizes the loglikelihood function $L(\Phi;\ \cdot\ ) = \log f(\ \cdot\ ;\Phi)$. The EM algorithm helps when the maximization of the loglikelihood function is analytically unfeasible. The key idea is based on the assumption that there exists a complete data density function $f(z;\Phi) = f(v,y;\Phi) = f(v|y;\Phi)f(y;\Phi)$ where Z is the set of all data (missing or not), composed by the actual observations y and by the missing (additional) variables v. Then, considering the complete data loglikelihood function $L(\Phi;y,v)$ and assuming that the current parameter estimate is Φ_{i-1}, the EM algorithm first finds the expected value of the complete data loglikelihood with respect to the unobserved variables Y, under the current value of the parameter Φ_{i-1}, and then maximizes this quantity to obtain the new estimate Φ_i of the parameters. The algorithm can be implemented in two steps:

E step compute $Q(\Phi,\Phi_{i-1}) := E(\log f(y,v;\Phi)|y;\Phi_{i-1})$;

M step find Φ that maximizes $Q(\Phi,\Phi_{i-1})$ and let $\Phi_i) = \Phi$.

Dempster et al. (1977) show, under some regularity conditions, that each iteration increases the loglikelihood and that the algorithm converges to a stationary point of the likelihood function, although perhaps not a maximum.

In many situations, the application of the EM algorithm is limited by practical problems. Apart from not guaranteeing convergence to a global maximum, in some cases the algorithm has a very slow rate of convergence, especially near to a stationary point, whereas in others the conditional expectation required at the E step cannot be calculated explicitly. The stochastic version of the EM algorithm that we consider here, namely the SEM algorithm (Celeux and Diebolt, 1985), were introduced to overcome these limitations. This algorithm differ from the EM algorithm by the introduction at each iteration of a simulation step in which a sample is drawn from the conditional distribution of the unobserved variables given the data. Although this was originally motivated by different reasons, the algorithms can be considered as random perturbations of the deterministic discrete time dynamic system generated by the EM algorithm. The simulation step has been justified on the basis of the so called stochastic imputation principle that suggests the completion of incomplete samples by drawing values, for the unobserved variables, from their con- ditional distribution given the data. The completed samples are then used to find the current fit of the parameters. Practically, the SEM algorithm can be implemented as follows:

S step draw a sample v_i from the density $P(v|y;\Phi_{i-1})$;

E step compute the completed data likelihood $Q_S(\Phi,\Phi_{i-1}) = logP(y,v_i;\Phi_{i-1})$;

M step find Φ that maximizes $Q_S(\Phi,\Phi_{i-1})$ and let $\Phi_i = \Phi$.

By running the algorithm, a random sequence $(\Phi_i)_{i\in\mathbb{N}}$ that is not converging pointwise, is obtained. This sequence turns out to be a homogeneous Markov chain which, whenever the chain is irreducible and ergodic, converges to a unique stationary distribution. Although this distribution is not a posterior distribution on the parameters, as one might think from a Bayesian perspective, but the result of a stochastic perturbation of the EM algorithm, it can be used for inference on Φ . To obtain a point estimate of Φ , various methods have been proposed, the most used considering the mean over a sufficiently large number of iterations.

To pratically implement the algorithm in our model, first of all an initial set of parameters $\Psi^{(0)}$ has to be fixed. and then for each step $h = 1,\dots,n$, where n is the number of iterations, perform:

S step - apply the MCMC algorithm described above, with the current values of the parameters Φ_{i-1}, to select a sample v_i from the distribution $P(v|y;\Phi_{i-1})$;

E step - compute the complete likelihood

$$\begin{aligned}Q_s(\Psi,\Phi_{i-1}) =& \ln\left(\prod_{t=2}^{T} P(y_t|v,\Psi)P(v_t|\Psi)\right) =\\ &= \Bigg\{ -\frac{1}{2}\sum_{t=2}^{T}(\ln[(1-\rho^2)\Delta t] + \ln v_{t-1}) - \sum_{t=2}^{T}\frac{(y_t-\mu_{y_t,v_t})^2}{2(1-\rho)^2\Delta t v_{t-1}} -\\ &- \frac{1}{2}\sum_{t=2}^{T}\ln(2\pi\sigma_{v_t}^2) - \sum_{t=2}^{T}\frac{(v_t-\mu_{v_t})^2}{2\sigma_{v_t}^2}\Bigg\}\ ,\end{aligned} \tag{4.25}$$

with μ_{y_t,v_t} as in (4.21), μ_{v_t} as in (4.23) and σ_{v_t} as in (4.24);

M step - maximize the complete likelihood for the value of the parameters. To do this we use the genetic algorithm function contained in MATLAB.

4.2 The pricing kernel

The pricing kernel, also known as *stochastic discount factor*, is strictly linked to consumption-based models, see for example Cochrane (2001), emphasizing equilibrium considerations. Here, as in Yang (2004), we focus on a no-arbitrage approach: once the real world parameters for the stock dynamics have been estimated, the option prices available on financial markets are be used to estimate the parameters of a specified form for the pricing kernel.

In this context, the price of a European call option with strike price K and expiration date T is given by

$$C_t = E\left[\max(S_T - K, 0)\phi_{t,T} | \mathscr{F}_t\right] \qquad 0 \leq t \leq T$$

where the filtration $\mathscr{F}_t$ represents the available market data at time t and S_t is the process which describes the underlying asset dynamics. As in Düring (2009), we chose the following shape for the pricing kernel:

$$\phi_{t,T} = \frac{\sum_{i=1}^{N} \alpha_i S_T^{\delta_i}}{E\left[\sum_{i=1}^{N} \alpha_i S_T^{\delta_i} | \mathscr{F}_t\right]} \qquad 0 \leq t \leq T \tag{4.26}$$

with $\alpha_i, \delta_i, \in \mathbb{R}, N \in \mathbb{N}$. The price of a call option is then given by

$$\begin{aligned} C\left(S_t, v_t, t\right) &= E\left[\max(S_T - K, 0)\frac{\sum_{i=1}^{N} \alpha_i S_T^{\delta_i}}{E[\sum_{i=1}^{N} \alpha_i S_T^{\delta_i} | \mathscr{F}_t]} | \mathscr{F}_t\right] \\ &= E\left[\sum_{i=1}^{N} \frac{\max(S_T - K, 0)\alpha_i S_T^{\delta_i}}{E[\sum_{i=1}^{N} \alpha_i S_T^{\delta_i} | \mathscr{F}_t]} | \mathscr{F}_t\right] \\ &= \sum_{i=1}^{N} \frac{E[\alpha_i S_T^{\delta_i} | \mathscr{F}_t]}{E[\sum_{i=1}^{N} \alpha_i S_T^{\delta_i} | \mathscr{F}_t]} E\left[\frac{\max(S_T - K, 0)\alpha_i S_T^{\delta_i}}{E[\alpha_i S_T^{\delta_i} | \mathscr{F}_t]} | \mathscr{F}_t\right]. \end{aligned}$$

As shown in Düring (2009), this can be understood as a weighted sum of option prices:

$$C\left(S_t, v_t, t\right) = \sum_{i=1}^{N} w_i C_i\left(F_t^{(i)}, v_t, t; K\right), \tag{4.27}$$

where w_i are weights and $F_t^{(i)}$ is the 'virtual asset price', i.e. the underlying asset price that would hold if the elasticity of the ASPK were δ_i. The prices C_i are given by

$$C_i\left(F_t^{(i)}, v_t, t; K\right) = F_t^{(i)} \mathscr{I}_1 - K \mathscr{I}_2,$$

with $(k = 1, 2)$

$$\begin{aligned}
\mathscr{I}_k &= \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-i\omega\ln(K)}f_k(\omega)}{i\omega}\right]d\omega, \qquad (4.28)\\
f_k(\omega) &= e^{A(T-t,\omega;\delta_k)-A(T-t,0;\delta_k)+v_t[B(T-t,\omega;\delta_k)-B(T-t,0;\delta_k)]+i\omega\ln(S_t)},\\
A(\tau,\omega) &= \frac{\kappa\theta}{\xi^2}\left[(b+d)\tau - 2\ln(\frac{1-ge^{d\tau}}{1-g})\right] + \mu(\delta - i\omega)\tau,\\
B(\tau,\omega) &= \frac{b+d}{\xi^2}\frac{1-e^{d\tau}}{1-ge^{d\tau}},\\
g &= \frac{b+d}{b-d}, \qquad d = \sqrt{(\omega^2 - i\omega(1-2\delta_k) + \delta_k(1-\delta_k))\xi^2 + b^2},\\
b &= \kappa - \rho\xi(i\omega + \delta_k), \qquad \delta_1 = \delta_i + 1, \qquad \delta_2 = \delta_i\ .
\end{aligned}$$

The ‘virtual asset’ price $F_t^{(i)}$ is given by

$$\begin{aligned}
F_t^{(i)} &= \frac{E[S_T^{\delta_i+1}|\mathscr{F}_t]}{E[S_T^{\delta_i}|\mathscr{F}_t]}\\
&= S_t e^{A(T-t,0;\delta_i+1)-A(T-t,0;\delta_i)+v_t[B(T-t,0;\delta_i+1)-B(T-t,0;\delta_i)]}, \quad 0 \le t \le T\ . \qquad (4.29)
\end{aligned}$$

Since the model has a nonzero drift component we prove formula (4.29), since the component A differs from the formulation in Düring (2009), to which we refer fot other proofs related to the formulas above.

According to formula (4.29), the virtual asset price $F_t^{(i)}$ is given by

$$F_t^{(i)} = \frac{E[S_T^{\delta_i+1}|\mathscr{F}_t]}{E[S_T^{\delta_i}|\mathscr{F}_t]}$$

Hence it is necessary to compute $V = E[S_T^\delta|\mathscr{F}_t]$ for arbitrary δ. Assuming $V = V(S,\delta,t)$ and sufficiently smooth, by Itô’s lemma we get

$$\begin{aligned}
dV = \Big(V_t + \mu SV_S + a(v)V_v + \frac{1}{2}vS^2V_{SS} + \rho b(v)\sqrt{v}SV_{Sv} + \frac{1}{2}b^2(v)V_{vv}\Big)\,dt\\
+ \sqrt{v}SV_S dW^{(1)} + b(v)V_v dW^{(2)}
\end{aligned} \qquad (4.30)$$

To perform a pricing analysis, what follows need to be addressed in a risk-neutral environment. Hence the overall process must be a martingale. Taking the expectation on both sides we get

$$V_t + \mu SV_S + \frac{1}{2}S^2vV_{SS} + \rho b(v)\sqrt{v}SV_{Sv} + \frac{1}{2}b^2(v)V_{vv} + a(v)V_v = 0$$

This has to to be solved for $S, v > 0, 0 < t < T$, with final condition $V(S, v, T) = S^\delta$. The only difference from Düring (2009) is the component $\mu S V_S$. Similar as in Heston (1993), we guess a solution of the form $V = S^\delta P$. Now performing the transformation of variables $x = \ln S$ and $\tilde{t} = T - t$ and immediately dropping the tilde for notation simplicity, we obtain

$$\begin{aligned} P_t - \mu\delta P - \mu P_x - \frac{1}{2}v(P_{xx} - P_x) - \delta v P_x - \rho\sqrt{v}b(v)P_{xv} - \frac{1}{2}b^2(v)P_{vv} \\ - \left[a(v) + \rho\sqrt{v}b(v)\delta\right] P_v - \frac{1}{2}v\delta(\delta - 1)P = 0, \end{aligned} \tag{4.31}$$

Let's introduce the Fourier transform of P,

$$\hat{P}(\omega, v, t) = \int_{\mathbb{R}} e^{i\omega x} P(x, v, t) dx$$

under which differentiation with respect to x turns into multiplication with $-i\omega$. Hence we get

$$\begin{aligned} \hat{P}_t = \mu\delta\hat{P} - i\mu\omega\hat{P} + \frac{1}{2}b^2(v)\hat{P}_{vv} + \left[a(v) + \rho\sqrt{v}b(v)\delta - i\omega\rho\sqrt{v}b(v)\right] \hat{P}_v \\ - \left[\frac{1}{2}v\omega^2 + i\omega(\delta - \frac{1}{2})v - \frac{1}{2}\delta(\delta - 1)v\right] \hat{P} \end{aligned} \tag{4.32}$$

In the following we substitute $a(v)$ and $b(v)$ with their form according to the Heston model. This allows to solve explicitly for the characteristic function of P. As a last step, by inserting the ansatz $\hat{P} = e^{A(t,\omega;\delta)+vB(t,\omega;\delta)+i\omega x}$ with $A(0, \omega; \delta) = B(0, \omega; \delta) = 0$ into 4.32, we get the two ordinary differential equations

$$\begin{aligned} A' &= \kappa\theta B + \mu(\delta - i\omega) \\ B' &= \frac{1}{2}\xi^2 B^2 - [\kappa - \rho\xi\delta - i\omega\rho\xi] B - \frac{1}{2}\omega^2 - i\omega\left(\delta - \frac{1}{2}\right) + \frac{1}{2}\delta(\delta - 1) \end{aligned}$$

subject to $A(0) = B(0) = 0$, which can be solved yielding to the formulas above.

5 Empirical results

This section is devoted to the presentation of the empirical results regarding real world parameters estimation (by using both SEM and particle filter) and the performance of the pricing kernel technique, which is compared with the results we obtained with the classical calibration approach. This analysis is presented both in the sample and out of the sample.

5.1 Estimation results

Three different analysis of the SEM and particle filtering algorithm have been developed: at first it is applicated to simulated data, allowing to compare the estimates with the values used in the simulation of the volatility and price paths; then a wider analysis on simulated data is presented; at last, the algorithms are applied to a real dataset.

First of all we simulated a four years path using the Euler discretization of the Heston model. Then the price trajectory, which represents the observable data, is used as an input to the algorithms, which estimates jointly the parameters and the unobservable volatility path. The parameter values used to simulate the paths, as well as the volatility path, are denoted here as 'true' for comparison with the estimated values.

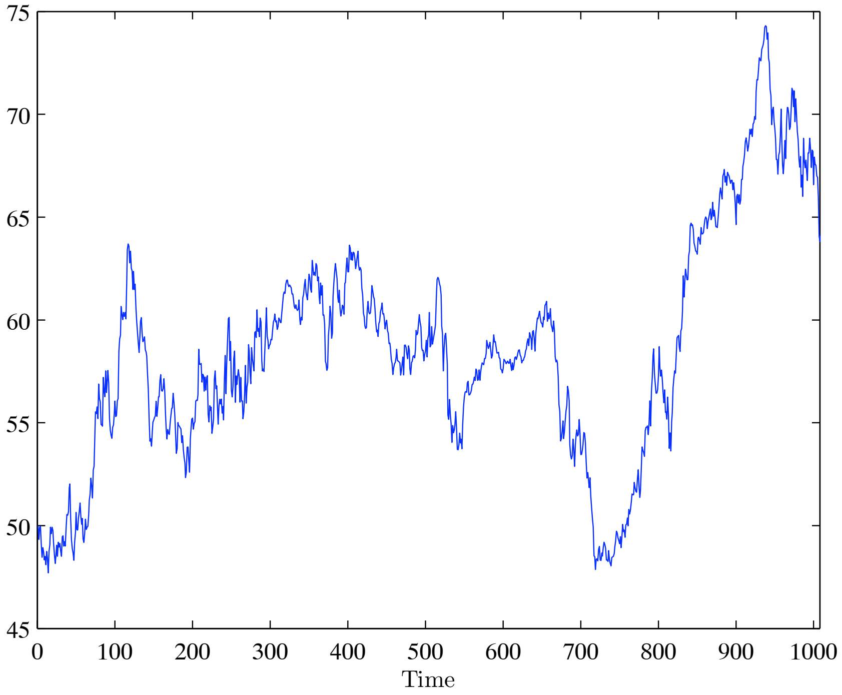

Figure 1: Four years of daily prices

Figure 1 shows the daily prices used as an input to the SEM algorithm, while the point estimates obtained are presented in table 1.

The point estimates have been obtained as an average on the last 200 steps of the SEM. The estimates at each step $n = 1, \dots, 1300$ are also shown in Figure 2, while Figure 3 shows the true and estimated volatility paths.

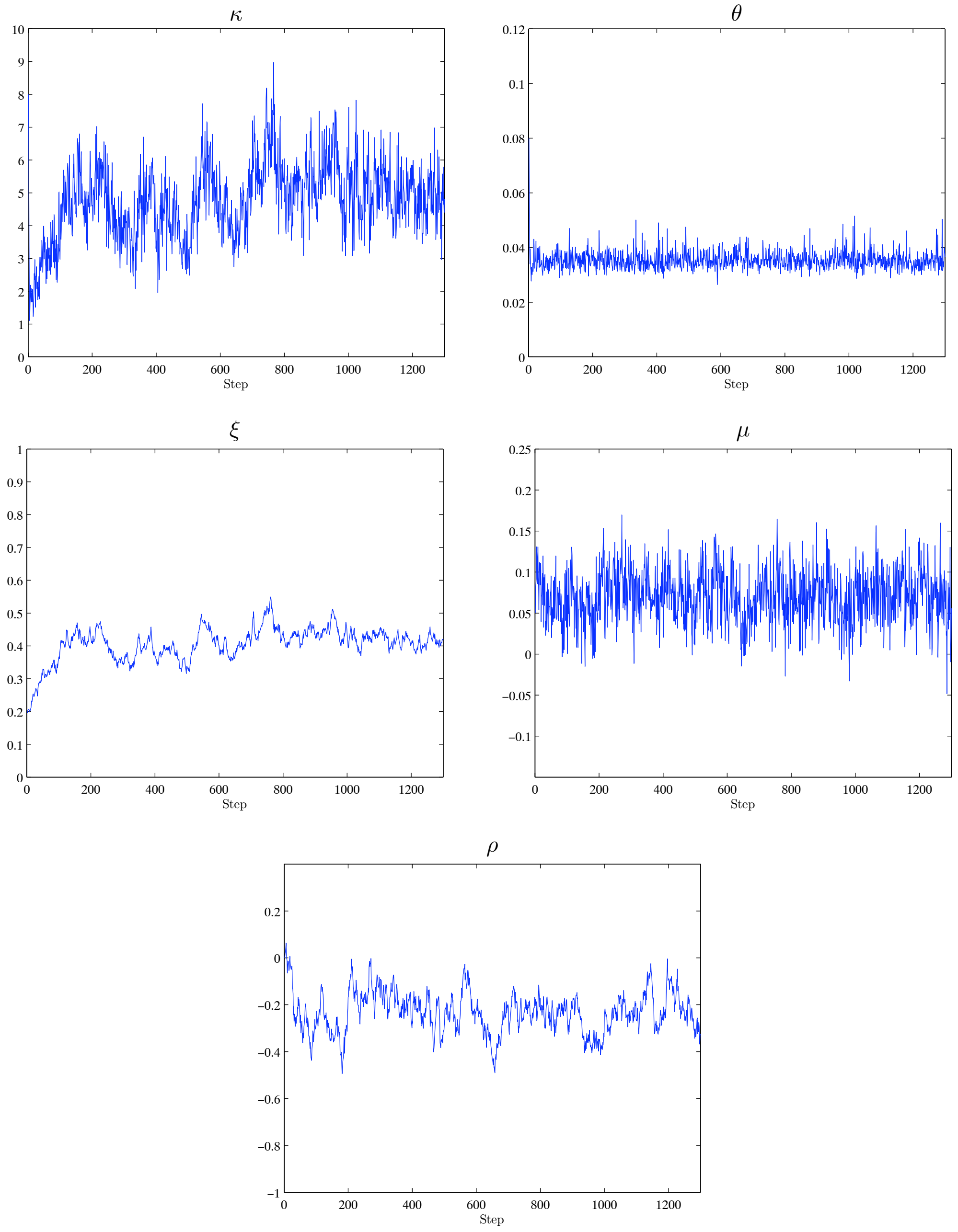

Figure 2: SEM: estimated parameters

Table 1: SEM: estimation results on simulated data

Parameter	True value	Estimated value
κ	4.000	4.920
θ	0.040	0.0351
ξ	0.450	0.415
μ	0.080	0.0714
ρ	-0.30	-0.206

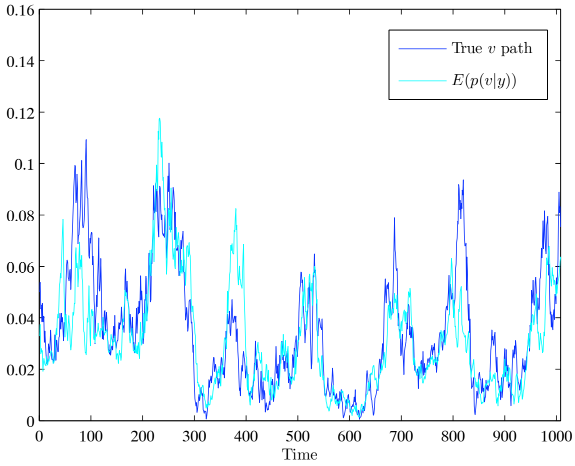

Figure 3: True and estimated v paths

Although a deeper analysis could be useful to understand and possibly reduce this effect, the point estimates obtained are close to the true values. Furthermore, as can be seen the estimated volatility path is very close to the true path.

Using the same parameter values, we repeated 50 times independently the simulation of a price path. Then for each of the simulated path we stored the point estimates obteined with the SEM. The results are shown in Figure 4, where it can be observed that the histograms include the true value, although a deeper analysis could be useful to understand and possibly reduce the variability, particularly for the mean reversion parameter κ.

As for the SEM, the particle filter is applied to simulated data.

In this case it is not possible to select the number of steps, since each daily observation

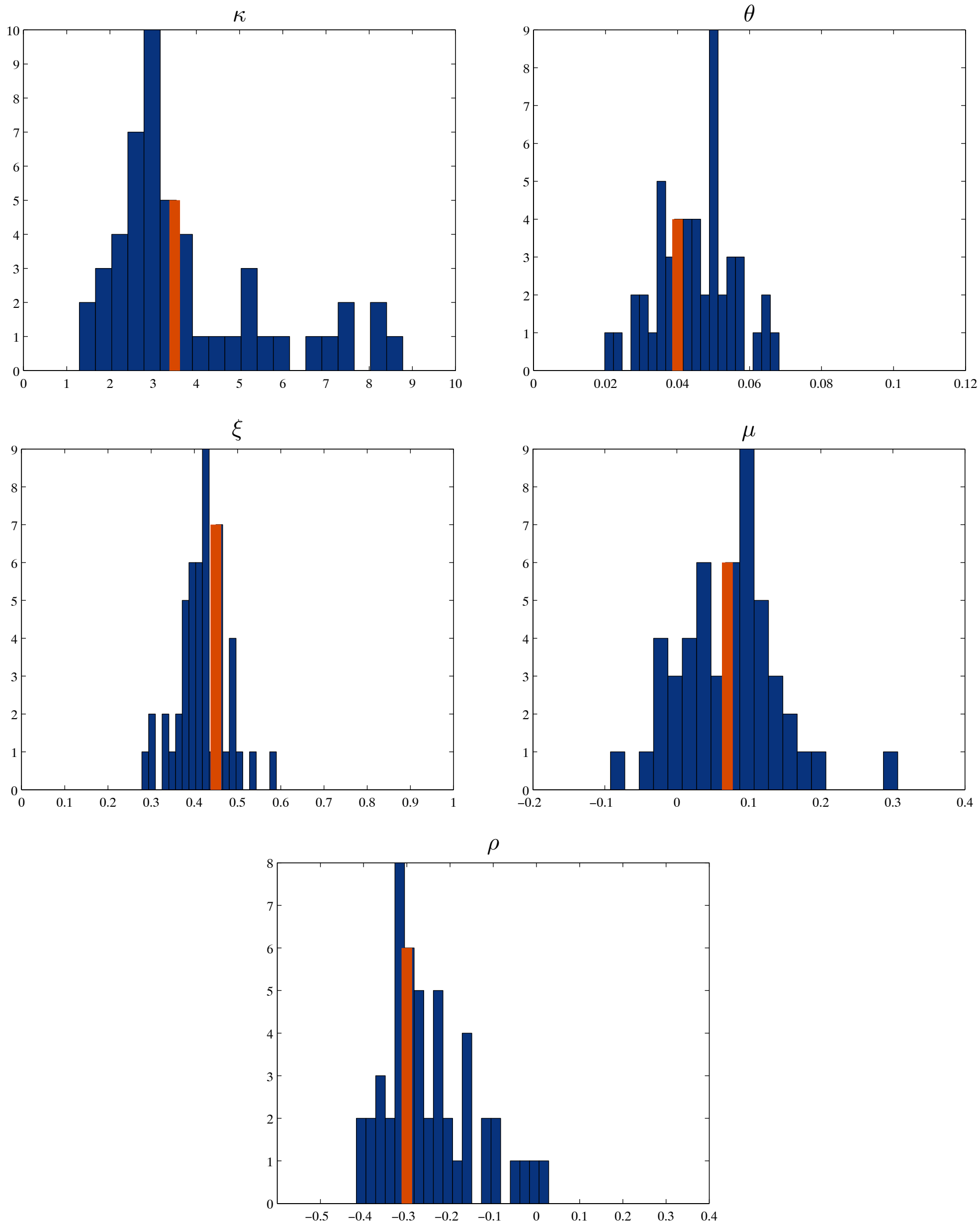

Figure 4: Results of 50 SEM estimates

is a step of the algorithm; then the point estimates are the final values of the parameters obtained at current time t. The daily price path used is shown in Figure 5. The point estimates are presented in table 2, while Figure 6 shows the estimated volatility path. Finally, Figure 7 shows the parameter estimates step by step. It can be seen that the estimates converge, notwithstanding the convergence requires more steps for the parameters κ and θ. As stated before, the algorithm we use presents some problems related to the treatment of very high values and '0/0' in importance weights evaluation. Changing the proposal distribution π results in more stable but poorer performance, hence we retain this 'more troubling' specification. Further work is necessary to address undesired values in a more effective way.

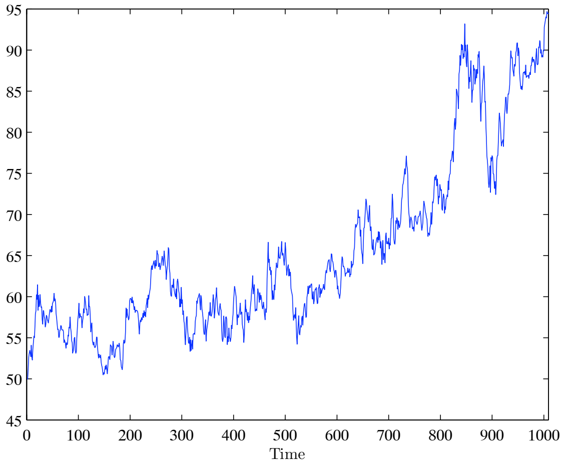

Figure 5: Four years of daily prices

Table 2: Particle filter: estimation results on simulated data

Parameter	True value	Estimated value
κ	4.000	3.945
θ	0.060	0.065
ξ	0.400	0.4069
μ	0.120	0.0627
ρ	-0.200	-0.083

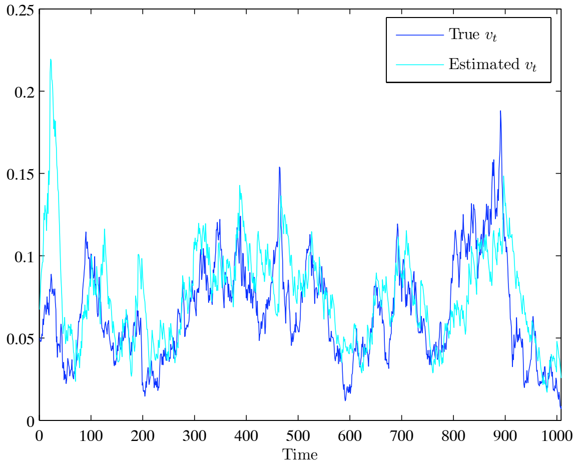

Figure 6: True and estimated v paths

Comparing the SEM and the particle filter algorithms, we found that the results obtained with SEM are much stable. It is an important feature, although it has to be considered that the two algorithms require extremely different computing time to get the estimates. Our implementation of the SEM needs about 5 hours on a computer with processor speed of 2.50GHz. Under the same conditions, the particle filter algorithm gets an estimate in less than one minute. In practical applications such a difference could be worthy. It could be also worthy to notice that while, as stated before, the SEM algorithm is much slower than particle filter, the first allows for greater flexibility. Among others, for example, it is possible to make choices regarding the trade-off between speed and precision. Conversely, our particle filter implementation does not allow for a great improvement in precision, even if one increases the number of particles.

Concerning volatility path estimates, our results are more than satisfying with both the algorithms. This estimate is surprisingly stable with particle filter too, even when it gets less satisfying parameter estimates.

At last, the algorithms are applied to get estimations of the Heston model parameters using data on the italian FTSE Mib Index (which was called S&P Mib before june 2009) from 24/07/2006 to 23/07/2010. The index time series is shown in Figure 8.

The point estimates we obtained with the SEM algorithm are presented in the table 3, while the estimated volatility path is shown in Figure 9.

Table 4 shows the point estimates we obtained with the particle filter, while the estimated volatility path is shown in Figure 10.

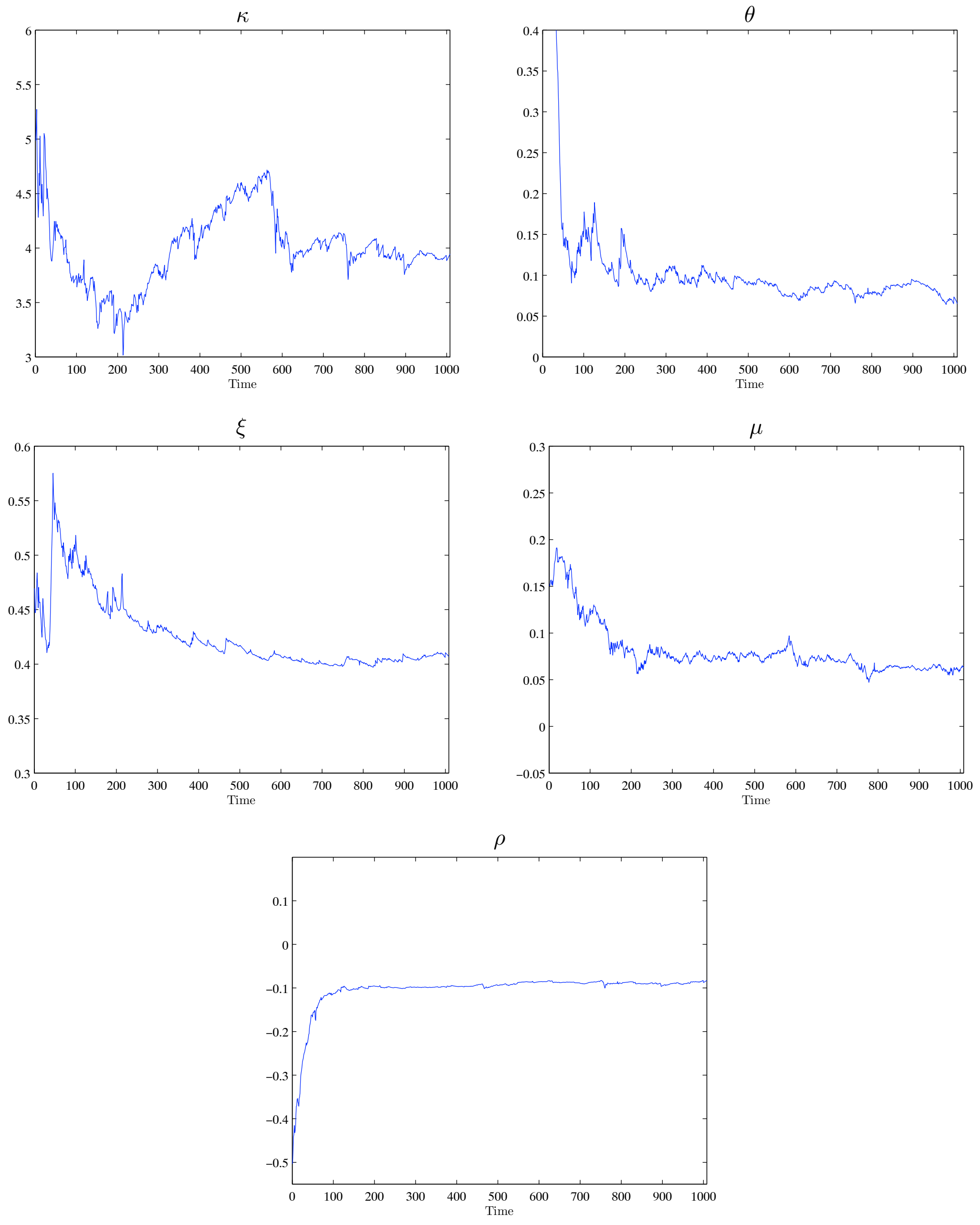

Figure 7: Particle filter: estimated parameters

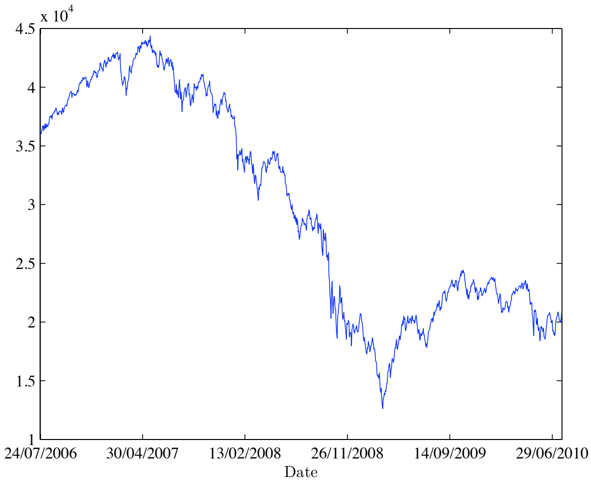

Figure 8: Value of FTSE Mib index (24/07/2006 - 23/07/2010)

Table 3: SEM: estimation results on real data

Parameter	Estimate
κ	3.842
θ	0.057
ξ	0.514
μ	0.049
ρ	-0.469

5.2 Option pricing

Here we present the results of the option pricing model we described in section 3. Tables 5 and 6 show the data we used for pricing and they are the prices of 61 call options written on the italian FTSE Mib index. These options are as quoted on 23/07/2010, with S_0 representing the initial value of the underlying, K the strike price, τ the time to maturity, r the risk-free rate and q the estimated dividend yield. As will be explained below, we used different groupings of these data to perform in the sample and out of the sample pricing. To do this, we have calibrated Heston parameters for the classical pricing technique based on Carr and Madan (1998), as well as the pricing kernel parameters which are necessary for the second step of the pricing kernel approach. We use the real

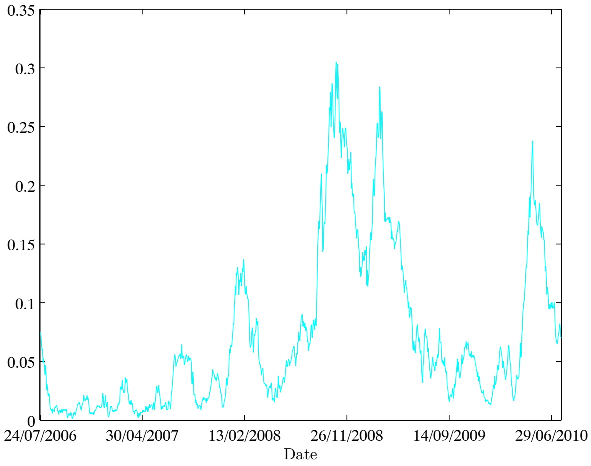

Figure 9: Estimated v for FTSE Mib (24/07/2006 - 23/07/2010)

Table 4: Particle filter: estimation results on real data

Parameter	Estimate
κ	3.052
θ	0.1245
ξ	0.6645
μ	0.0551
ρ	-0.0787

world parameter estimates obtained with the SEM technique on daily index data from 24/07/2006 to 23/07/2010 obtained before.

Table 5: Data used for option pricing

Price	S_0	K	τ	r	q
3600	20604	17000	0.076712329	0.0062	0
1820	20604	19000	0.076712329	0.0062	0
1340	20604	19500	0.076712329	0.0062	0
990	20604	20000	0.076712329	0.0062	0
660	20604	20500	0.076712329	0.0062	0
400	20604	21000	0.076712329	0.0062	0
224	20604	21500	0.076712329	0.0062	0
114	20604	22000	0.076712329	0.0062	0
55	20604	22500	0.076712329	0.0062	0
25	20604	23000	0.076712329	0.0062	0
13	20604	23500	0.076712329	0.0062	0
7	20604	24000	0.076712329	0.0062	0
5	20604	24500	0.076712329	0.0062	0
3340	20604	17500	0.153424658	0.0069	0.0011
2485	20604	18500	0.153424658	0.0069	0.0011
1980	20604	19000	0.153424658	0.0069	0.0011
1620	20604	19500	0.153424658	0.0069	0.0011
1280	20604	20000	0.153424658	0.0069	0.0011
950	20604	20500	0.153424658	0.0069	0.0011
685	20604	21000	0.153424658	0.0069	0.0011
468	20604	21500	0.153424658	0.0069	0.0011
312	20604	22000	0.153424658	0.0069	0.0011

Table 6: Data used for option pricing

Price	S_0	K	τ	r	q
198	20604	22500	0.153424658	0.0069	0.0011
116	20604	23000	0.153424658	0.0069	0.0011
70	20604	23500	0.153424658	0.0069	0.0011
40	20604	24000	0.153424658	0.0069	0.0011
23	20604	24500	0.153424658	0.0069	0.0011
13	20604	25000	0.153424658	0.0069	0.0011
1120	20604	20500	0.230136986	0.0083	0.0226
645	20604	21500	0.230136986	0.0083	0.0226
440	20604	22000	0.230136986	0.0083	0.0226
300	20604	22500	0.230136986	0.0083	0.0226
190	20604	23000	0.230136986	0.0083	0.0226
87	20604	24000	0.230136986	0.0083	0.0226
4800	20604	16000	0.402739726	0.0102	0.0265
3240	20604	18000	0.402739726	0.0102	0.0265
2475	20604	19000	0.402739726	0.0102	0.0265
1785	20604	20000	0.402739726	0.0102	0.0265
1460	20604	20500	0.402739726	0.0102	0.0265
1160	20604	21000	0.402739726	0.0102	0.0265
725	20604	22000	0.402739726	0.0102	0.0265
570	20604	22500	0.402739726	0.0102	0.0265

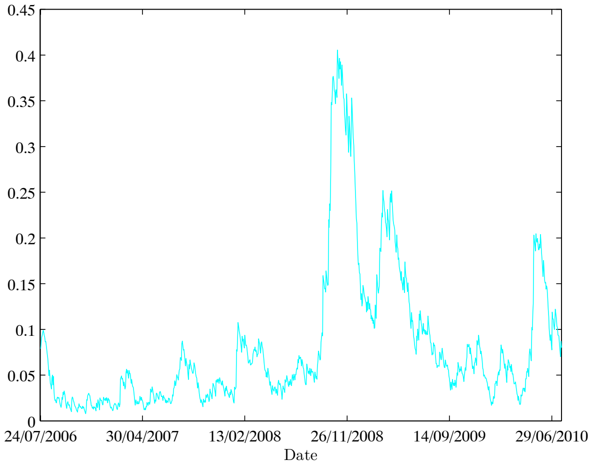

Figure 10: Estimated v path

Sample used for pricing					
Price	S_0	K	τ	r	q
434	20604	23000	0.402739726	0.0102	0.0265
182	20604	24000	0.402739726	0.0102	0.0265
138	20604	24500	0.402739726	0.0102	0.0265
98	20604	25000	0.402739726	0.0102	0.0265
36	20604	26500	0.402739726	0.0102	0.0265
2150	20604	20000	0.652054795	0.0121	0.0165
755	20604	23000	0.652054795	0.0121	0.0165
460	20604	24000	0.652054795	0.0121	0.0165
280	20604	25000	0.652054795	0.0121	0.0165
202	20604	25500	0.652054795	0.0121	0.0165
2485	20604	19500	0.901369863	0.0134	0.0404
2195	20604	20000	0.901369863	0.0134	0.0404
1930	20604	20500	0.901369863	0.0134	0.0404
1660	20604	21000	0.901369863	0.0134	0.0404
1470	20604	21500	0.901369863	0.0134	0.0404
1210	20604	22000	0.901369863	0.0134	0.0404
1020	20604	22500	0.901369863	0.0134	0.0404
122	20604	30000	1.400000000	0.0137	0.0446
75	20604	37000	2.416438356	0.0150	0.0422

The different groupings are chosen as follows.

- *Grouping A* is chosen randomly, with no particular criterion.
- *Grouping B* is designed to include carefully selected data in the sample used for calibration. We take options for all the available maturities and for the overall range of strike prices, such that no information for at the money, in the money or out of the money options is excluded.
- *Grouping C* is chosen to include poor information in the sample used for calibration. It reflects a situation in which one can rely only on little information to perform pricing, as would be the case for some OTC instruments.

To evaluate the pricing results, the following measures are proposed in Gilli and Schumann (2010).

$$\begin{aligned}
rmse &= \sqrt{\sum_{options} \frac{(\text{Market price} - \text{Model price})^2}{\text{number of options}}} \\
ape &= \frac{1}{\text{mean option price}} \sum_{options} \frac{|\text{Market price} - \text{Model price}|}{\text{number of options}} \\
aae &= \sum_{options} \frac{|\text{Market price} - \text{Model price}|}{\text{number of options}} \\
arpe &= \frac{1}{\text{number of options}} \sum_{options} \frac{|\text{Market price} - \text{Model price}|}{\text{Market price}}
\end{aligned}$$

In our calibration algorithm we minimize the relative error, hence we will use the the *arpe* value here.

First of all, to ensure the effectiveness of our benchmark, we tested the algorithm we use for classical calibration pricing against generally appreciated results. As stated before the calibration algorithm we use here is based on Gilli and Schumann (2010) and the GA algorithm. We applied it to the option set used in Schoutens et al. (2004). The option pricing technique used in Gilli and Schumann (2010) and Schoutens et al. (2004) is exactly the same and is based on Fast Fourier Transforms. As a consequence, the effectiveness of the algorithm depends on the optimization component only. For the *arpe* measure we obtained a value of 0.0088, which compared to the value of 0.0174 in Schoutens et al. (2004) shows that our calibration algorithm can be considered a good benchmark to test the pricing kernel approach. Note that in our case the relative error is the measure we minimize inside the optimization algorithm, while Schoutens et al. (2004) use the sum of squared errors as objective function.

Tables 7 and 8 show our pricing results for FTSE Mib call options.

Table 7: In the sample pricing results: *arpe*

	Calibration	Pricing kernel
grouping A	0.0753	0.1149
grouping B	0.0782	0.0860
grouping C	0.0255	0.0518

Table 8: Out of the sample pricing results: *arpe*

	Calibration	Pricing kernel
grouping A	0.1019	0.1101
grouping B	0.0845	0.1610
grouping C	0.4505	0.2861

As can be seen, while in the sample both methods give results which are not fully satisfying, except for the case of in the sample pricing of *grouping C*, this is not the case for the out of the sample performance. In fact, in the second case the additional information contained in the underlying historical path might be useful: the pricing kernel has good performances when the information contained in the sample used for calibration regards only a limited range of maturities and strike prices. On the other hand, the options used in the sample and in *grouping C* were more homogeneous (there were no highly in the money or out of the money options). This stresses the indadequacy of the classical calibration of the Heston model to fit the entire volatility surface at once. For improvements in this sense, see e.g. Fonseca et al. (2008).

Bibliography

S. I. Aihara, A. Bagchi, and S. Saha. Estimating volatility and model parameters of stochastic volatility models with jumps using particle filter. In *Proceedings of the 17th World Congress of The International Federation of Automatic Control*, 2008.

C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An Introduction to MCMC for Machine Learning. *Machine Learning*, 50:5–43, 2003.

G. Bakshi, C. Cao, and Z. Chen. Empirical Performance of Alternative Option Pricing Models. *The Journal of Finance*, 52:2003–2049, 1997.

D. Bates. Jumps and stochastic volatility: Exchange rate processes implicit in Deutschemark options. *Review of Financial Studies*, 9:69–108, 1996.

J. Besag. Markov Chain Monte Carlo for Statistical Inference. Working paper no.9, Center for Statistics and Social Sciences, University of Washington, 2001.

F. Black and M. Scholes. The Pricing of Options and Corporate Liabilities. *Journal of Political Economy*, 81:637–654, 1973.

P. Carr and D. Madan. Option Valuation using the Fast Fourier Transform. *Journal of Computational Finance*, 2:61–73, 1998.

G. Celeux and J. Diebolt. The SEM algorithm: A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly*, 2:73–82, 1985.

J. H. Cochrane. *Asset pricing.* Princeton University Press, Princeton, 2001.

J. C. Cox, J. E. Ingersoll, and S. A. Ross. A Theory of the Term Structure of Interest Rates. *Econometrica*, 53:385–408, 1985.

F. Delbaen and W. Schachermayer. A general Version of the Fundamental Theorem of Asset Pricing. *Mathematische Annalen*, 300:463–520, 1994.

A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society*, 39:1–38, 1977.

A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York, 2001.

B. Düring. Asset pricing under information with stochastic volatility. *Review of Derivatives Research*, 12:141–167, 2009.

J. D. Fonseca, M. Grasselli, and C. Tebaldi. A Multifactor Volatility Heston Model. *Quantitative Finance*, 8:591–604, 2008.

C. S. Forbes, G. M. Martin, and J. Wright. Bayesian Estimation of a Stochastic Volatility Model Using Option and Spot Prices. Working paper 2/2002, Monash University, 2002.

A. R. Gallant and G. Tauchen. Seminonparametric estimation of conditionally constrained heterogeneous process: asset pricing application. *Econometrica*, 57:1091–1120, 2009.

M. Gilli and E. Schumann. Calibrating Option Pricing Models with Heuristics. COMISEF Working Papers Series, 2010.

M. Harrison and D. Kreps. Martingales and arbitrage in multiperiod security markets. *Journal of Economic Theory*, 20:381–408, 1979.

S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6:327–343, 1993.

A. Javaheri. *Inside Volatility Arbitrage*. John Wiley & Sons, Hoboken, 2005.

C. Kardaras. Stochastic Discount Factors. Working paper, arXiv.org Quantitative Finance Papers series n.1001.1184, 2010.

E. Lüders and G. Franke. Predictability, Excess Volatility and Stock Market Crashes in Rational Expectations Models. working paper, CoFE discussion paper 04/05, 2004.

W. Schoutens, E. Simons, and J. Tistaert. A perfect calibration! Now what? *Wilmott Magazine*, March 2004.

J. C. Spall. Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation. *IEEE Transactions on Automatic Control*, 37:332–341, 1992.

J. C. Spall. Implementation of the Simultanueous Perturbation Algorithm for Stochastic Optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):817–823, 1998.

R. Storn and K. Price. Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11: 341–359, 1997.

J. Yang. *Semiparametric estimation of asset pricing kernel from time series and cross section data*. PhD thesis, Queen's University, Kingston, Ontario, 2004.

X. Yang, K. Xing, K. Shi, and Q. Pan. Joint state and parameter estimation in particle filtering and stochastic optimization. *Journal of Control Theory and Applications*, 6: 215–220, 2008.