On The Calibration Of The 3/2 Model

Hilmar Gudmundsson[a,*], David Vyncke[a]

[a] *Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Krijgslaan 281-S9, Ghent, BELGIUM*

Abstract

We consider the problem of calibrating the 3/2 stochastic volatility model to option data. In comparison to the characteristic function of the Heston model, the characteristic function of the 3/2 model can be up to 50 times slower to evaluate. This makes the standard least squares calibration with finite-difference gradients unreasonably slow. To address this problem we derive the analytic gradient of the characteristic function in compact form. We then propose a computational method for the analytic gradient formula which caches intermediate results across the partial derivatives, in addition to the strike dimension and the maturity dimension. Compared to the fastest method of calibrating the 3/2 model which we could find in the literature, the method proposed in this paper is more than 10 times faster. We also discuss the issue of apparent non-convexity in the least squares calibration of the 3/2 model for market data. To tackle it, we propose a regularized calibration where the regularization point is obtained using "risk neutral" MCMC estimation of the model. We find that this approach is particularly well suited for the calibration problem as it generates naturally a consistent damping matrix for the parameter estimates, in addition to being very fast.

Keywords: Pricing, Model Calibration, 3/2 Model, MCMC Estimation, Optimization

1. Introduction

Since the Black-Scholes model's rise to prominence in the field of option pricing, numerous extensions to it have been proposed in an attempt to address its well known limitations. One category of such extensions involves generalizing the Black-Scholes by making the volatility of the underlying asset(s) a stochastic process. While the assumption of stochastic volatility has been shown to enable option pricing models to reproduce many of the empirical features of financial

[*] Corresponding author

Email addresses: hilmar.gudmundsson@ugent.be (Hilmar Gudmundsson), david.vyncke@ugent.be (David Vyncke)

markets which the Black Scholes model misses, the added model complexity brings about its own challenges. In particular, they generally lack a closed-form solution. This becomes particularly significant during the calibration of the model to market data, where a potentially large set of benchmark options need to be priced by the model up to several thousand times, making the computational tractability of the model a key issue.

While stochastic volatility models generally do not yield themselves to a simple solution in the same way as the Black Scholes model, several special cases do admit a density for which we know the characteristic function in closed form. This essentially reduces the computation of an option price to a Fourier inversion, which in most cases is far cheaper computationally than solving the model through either a Monte Carlo simulation or a finite difference scheme.

The best known models with this property are those which correspond to an (exponentially) affine characteristic function, one example being the square root volatility model of Heston (1993) and its many (affine) extensions (see e.g. Bates (1996); Sepp (2008); Wong et al. (2009)). However, several studies done on the S&P 500 index seem to suggest that the dynamics of the underlying process are not affine, see e.g. Poteshman (1998). In particular, Bakshi et al. (2006) use the VIX index to estimate several specifications of the volatility process, including

$$dv_t = \left(\alpha_0 + \alpha_1 v_t + \alpha_2 v_t^2 + \frac{\alpha_3}{v_t}\right) dt + \beta_0 v_t^{\beta_1} dW_t \tag{1.1}$$

with $0 \leq t \leq T$ for a finite horizon T. Their findings suggest that both α_2 and α_3 are significant, and that β_1 is around 1.27.

A stochastic volatility model with the potential to better match these empirical findings (Carr et al., 2007) is the so-called 3/2 model. It is given by

$$\begin{aligned} dS_t &= \mu S_t dt + \sqrt{V_t} S_t dB_t \\ dV_t &= \kappa(\eta V_t - V_t^2)dt + \sigma {V_t}^{3/2} dW_t \end{aligned} \tag{1.2}$$

where B and W are standard Brownian motions with $\operatorname{Cov}(dB_t, dW_t) = \rho dt$. Although non-affine, the characteristic function for the corresponding density is known in closed form, implying the same tractability as we have for affine models.

A comparison of the 3/2 model and the Heston model reveals several fundamental differences. The 3/2 model allows for much stronger and more rapid volatility deviations away from the zero boundary than the Heston model. In addition, while the dynamics of the Heston model predict that the implied volatility skew flattens when the instantaneous volatility increases, the dynamics of the

3/2 model predict the opposite behavior. As such, the 3/2 model fits the empirical assumptions of risk managers better than the Heston model (Drimus, 2012).[1]

A potential issue, however, is that the characteristic function is considerably more complex than that of the Heston model. In particular, it involves the confluent hypergeometric function, $M(\alpha, \beta, z)$ which is defined as the solution to

$$z\frac{d^2w}{dz^2} + (\beta - z)\frac{dw}{dz} - \alpha w = 0 \tag{1.3}$$

This nests a wide variety of special functions, as well as the exponential function, which is the most computationally expensive function in the characteristic function of the Heston model. The computational cost of evaluating it in the parameter space that corresponds to plausible values of the parameters of the 3/2 model can be up to two orders of magnitude greater than for the exponential function.[2]

For a one factor stochastic volatility specification, this puts the characteristic function approach to option pricing with the 3/2 model rather close to a finite difference scheme in terms of speed. While this might not be a problem for evaluating the price of a single option, it certainly has a big impact on the calibration of the model. For this reason, speeding up the calibration process is crucial to making the model computationally efficient enough for practitioners.

With this in mind, we propose a calibration method which uses the analytic gradient of the characteristic function. The compact form of the analytic gradient we derive allows us to develop a very efficient caching method for the partial derivatives, in addition to the strike and maturity dimensions. This approach turns out to be an order of magnitude faster in terms of computing the gradient of the objective function compared to a centered finite difference gradient approximation, with a similar speedup in the calibration of the model. The speedup factor is similar to what is reported in Cui et al. (2017), where the authors give a comprehensive exposition of the calibration of the Heston model, that includes the derivation of the analytic gradient of the characteristic function of the Heston model.

Lastly, as has been reported in the literature, see e.g. (Cont et al., 2004; Guillaume et al.,

[1] Moreover, we do not have to choose between them. Knowing the characteristic function for both of these models means we can combine them into one multi-factor model for which we also know the characteristic function in closed form (Grasselli, 2017). This is a consequence of the convolution property of Fourier transforms, see Appendix C for a more in-depth discussion.

[2] This assumes a standard implementation of the confluent hypergeometric function for complex valued arguments, see Abramowitz et al. (1964)

2010; Mikhailov et al., 2003), the nonlinear least squares approach to calibrating option pricing models to market data is often accompanied by issues of non-convexity and numerical instability. More precisely, the objective function can exhibit several different local minima and using a purely gradient-based optimization can lead to very different parameter values depending on the initial guess. The precise choice of benchmark options also tends to have an impact on what values we obtain from the calibration procedure, and the optimal parameter values can fluctuate considerably between consecutive trading days.[3]

To tackle these issues for the 3/2 model, we propose using a Tikhonov type of regularization formulation of the least squares problem, where the regularization point is computed using MCMC estimation of the 3/2 model, and the damping matrix is the inverse covariance matrix of the parameter estimates. Since we are only after the parameter values under the risk neutral measure, we propose using the volatility index VIX as a proxy for the instantaneous volatility which greatly simplifies the MCMC estimation procedure.

The remainder of this paper is as follows. In section 2, we derive the analytic gradient for the characteristic function of the 3/2 model and present the caching algorithm for quickly evaluating it for a set of benchmark options. In section 3 we describe a regularized version of the calibration problem, and give the MCMC estimators we use for computing the regularization point and the damping matrix. Finally, in section 4 we present the results of our numerical experiments on these methods.

2. Calibration of the 3/2 Model with an Analytic Characteristic Function Gradient

Given a set of market quoted plain vanilla options, $i = 1, \dots, I$, we characterize option i as the pair $(K, T)_i$, where K is its strike price, and T is its maturity. We observe the market price C_i^{Market} for each of these options at time t, and wish to infer the parameters of our pricing model using these observed market prices. The standard approach to solving this problem involves solving the weighted (nonlinear) least squares minimization problem,

$$\mathbf{\Theta}^* = \underset{\mathbf{\Theta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{I} w_i f_i(\mathbf{\Theta})^2 \right\} \tag{2.1}$$

[3] Of course, we could also interpret this in a way that the model should incorporate parameters that are themselves stochastic processes, but that is outside the scope of this paper.

where

$$f_i(\mathbf{\Theta}) \equiv C(\mathbf{\Theta}; (K,T)_i) - C_i^{Market}$$

Here, w_i is a weight which determines the influence of option price i on the calibration, and $C(\mathbf{\Theta}; (K,T)_i)$ is the price of option i given by the model, given the parameter values $\mathbf{\Theta}$. In our case, the vector of parameters is given by $\mathbf{\Theta} = \{\kappa, \eta, v, \sigma, \rho\}$. Note that while the spot volatility $v = V_t$ is strictly speaking not a parameter of the model, it is unobserved and therefore included as a decision variable in the optimization as well.

The set of quoted options we consider for our numerical studies are plain European options. More precisely, denoting with S_T the price of the underlying asset at time T, the price $C^{(\text{call})}$ of a European call option at time t with strike K and maturity T is given by $C^{(\text{call})} = e^{-r(T-t)} E_t^{\mathcal{Q}} \left[\{S_T - K\}^+ \right]$. Likewise, the price $C^{(\text{put})}$ of a European put option at time t with the same strike and maturity is given by $C^{(\text{put})} = e^{-r(T-t)} E_t^{\mathcal{Q}} \left[\{K - S_T\}^+ \right]$. Here, the superscript $\mathcal{Q}$ refers to the fact that the expectation is taken under the risk-neutral measure, rather than the objective measure.

Since the probability density of S_T is not known in closed form for the 3/2 model, a straightforward calculation of the expectation of the payoff is not possible. On the other hand, we do know the characteristic function of the density of $\ln(S_T)$. More precisely, the characteristic function, $\phi(u) \equiv E\left[e^{iuX_T}\right]$, where $X_T \equiv \ln(S_T)$, is known in closed form. With the parameter set $\mathbf{\Theta}$ explicitly accounted for, and letting $x = \ln(S_t)$, it is given by

$$\phi(\mathbf{\Theta}; u) = e^{iux} \frac{\Gamma(\beta - \alpha)}{\Gamma(\beta)} \zeta^\alpha M(\alpha; \beta; -\zeta) \tag{2.2}$$

where $M(\alpha; \beta, -\zeta)$ is the confluent hypergeometric function of the first kind,

$$M(\alpha; \beta; -\zeta) \equiv \sum_{n=0}^{\infty} \frac{(\alpha)_n}{(\beta)_n} \frac{(-\zeta)^n}{n!} \tag{2.3}$$

with

$$(x)_n \equiv \begin{cases} \prod_{i=0}^{n-1} (x+i) & \text{if } n > 0 \\ 1 & \text{if } n = 0 \end{cases}$$

and α, β, and ζ are given by

$$
\begin{aligned}
\zeta &\equiv \frac{2\kappa\eta}{v\sigma^2\left(e^{\kappa\eta(T-t)}-1\right)}\\
\alpha &\equiv -\left(\frac{1}{2}-\frac{p}{\sigma^2}\right)+\sqrt{\left(\frac{1}{2}-\frac{p}{\sigma^2}\right)^2+\frac{2q}{\sigma^2}}\\
\beta &\equiv 2\left[\alpha+1-\frac{p}{\sigma^2}\right]\\
p &\equiv -\kappa+i\sigma\rho u\\
q &\equiv \frac{iu}{2}+\frac{u^2}{2}
\end{aligned}
\tag{2.4}
$$

Armed with the characteristic function, the next step is to apply an inverse Fourier transform to calculate the option price. The details of this step can vary depending on which technique we use, see e.g. Aichinger et al. (2013). We have selected the following formula by Attari (2004):

$$
\begin{aligned}
C(S_t,T,K) &= S_t-\frac{1}{2}e^{-r(T-t)}K\\
&\quad-\frac{e^{-r(T-t)}K}{\pi}\int_0^\infty \frac{\left(\mathrm{Re}(\phi(u))+\frac{\mathrm{Im}(\phi(u))}{u}\right)\cos(uk)+\left(\mathrm{Im}(\phi(u))-\frac{\mathrm{Re}(\phi(u))}{u}\right)\sin(uk)}{1+u^2}du
\end{aligned}
\tag{2.5}
$$

where $k = \ln(K)$. Comparing the speed of different Fourier based methods for pricing vanilla options, Kilin (2011) concludes that using adaptive quadrature, with intermediary caching with the formula above, vastly outperforms the FFT method of Carr et al. (1999), as well as the fractional FFT method of Chourdakis (2005). We present an extension of this idea in the following section, which takes advantage of the special structure of the characteristic function of the 3/2 model.

2.1. The Analytic Gradient of the Characteristic Function

As mentioned in the introduction, the computation of the characteristic function of the 3/2 model is very expensive, which makes a gradient-based approach to finding the solution to (2.1) the only practical option. More specifically, methods which compute

$$
\mathbf{\Theta}_{j+1}=\mathbf{\Theta}_j-\gamma_j H^{-1}\nabla_\Theta\left(\sum_{i=1}^{I}w_i f_i(\mathbf{\Theta}_j)^2\right)
$$

iteratively until one or more of the user provided stopping criteria are satisfied. Here γ_j is the step length, and H^{-1} is either the inverse Hessian of the objective function, or an approximation to the inverse Hessian. Given the existence of a feasible minimum, this iterative process should terminate for some j^* such that $\mathbf{\Theta}^*=\mathbf{\Theta}_{j^*}$. Calculating the full Hessian is very time consuming, so instead we utilize a Quasi-Newton method for solving (2.1), which is the standard approach to this problem. Quasi-Newton methods construct an approximation H^{-1} to the inverse Hessian by means of the gradient. This means that the gradient computation for (2.1) is the main computational work of finding $\mathbf{\Theta}^*$. How γ and H^{-1} are calculated depends on the specific Quasi-Newton method. For our calibration tests in section 4, we performed the minimization using the Levenberg-Marquardt algorithm (Marquardt, 1963) which is a typical approach to solving nonlinear least squares problems such as the calibration problem of (2.1).

Without an analytic form for the gradient, a finite difference approximation can be used, such as the simple forward difference quotient, or the symmetric difference quotient. These are given by

$$D_+ f(x) \equiv \frac{f(x+h) - f(x)}{h},\ D_\pm f(x) \equiv \frac{f(x+h) - f(x-h)}{2h}$$

respectively, with the latter being significantly more accurate than the first. There are, however, well known numerical issues that go along with using either of these for the purposes of optimization. To begin with, they are ill-conditioned and subject to cancellation errors when h gets small enough. On the other hand, putting h too large results in truncation errors. Even if h is optimally chosen, the value of D_+ ($D_\pm$) is only half as accurate (two-thirds as accurate) in terms of significant digits as $f(x)$ (see e.g. Griewank et al. (1987) for a more in-depth discussion).

Another issue is that if the optimization is constrained, which indeed it is in our case, invalid values can be generated on the boundary by the finite difference approximation due to the $x \pm h$ step. Lastly, the numerical gradient approach misses out on the interdependency between the different partial derivatives which can be used to speed up the computation of the gradient.

With that in mind, deriving and using the analytic gradient of the objective function (2.1) is well motivated in our attempt to speed up the calibration of the 3/2 model. We begin by noting that

$$\nabla_\Theta \left(\sum_{i=1}^{I} w_i f_i(\mathbf{\Theta})^2 \right) = 2 \sum_{i=1}^{I} w_i f_i(\mathbf{\Theta}) \nabla_\Theta f_i = 2 \sum_{i=1}^{I} w_i f_i(\mathbf{\Theta}) \nabla_\Theta C(\mathbf{\Theta}; (K,T)_i)$$

Furthermore, given the formula for the price $C(\mathbf{\Theta}; (K,T)_i)$ for $i = 1, \dots, I$ above, we see that

$$\begin{aligned} \nabla_\Theta C(\mathbf{\Theta}; (K,T)_i) = &-\frac{e^{-r(T-t)}K}{\pi} \times \\ &\int_0^\infty \frac{\left(\mathrm{Re}(\nabla_\Theta \phi(u)) + \frac{\mathrm{Im}(\nabla_\Theta \phi(u))}{u}\right)\cos(uk) + \left(\mathrm{Im}(\nabla_\Theta \phi(u)) - \frac{\mathrm{Re}(\nabla_\Theta \phi(u))}{u}\right)\sin(uk)}{1+u^2} du \end{aligned} \quad (2.6)$$

Therefore, to calculate the gradient of the objective function in (2.1) with respect to the parameter vector $\mathbf{\Theta}$, we need the gradient of the characteristic function ϕ, which is presented below through Theorem 2.1 and its corollary.

Theorem 2.1. *Define*

$$\tilde{\phi}(\omega; u) \equiv e^{iux}\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\zeta^{\alpha} M\left(\alpha; \beta; -\zeta\right)$$

where α, β and z are univariate functions of ω, with

$$\alpha' \equiv \frac{d\alpha}{d\omega}, \beta' \equiv \frac{d\beta}{d\omega}, \text{ and } \zeta' \equiv \frac{d\zeta}{d\omega}$$

then we have that

$$\begin{aligned}\Omega_\omega \equiv \frac{\partial\tilde{\phi}}{\partial\omega} = \tilde{\phi}(\omega; u)\Bigg[&(\beta'-\alpha')\psi(\beta-\alpha) - \beta'\psi(\beta) \\ &+ \left(\alpha' ln(\zeta) + \alpha\frac{\zeta'}{-\zeta}\right) + \frac{\beta G(\alpha;\beta;-\zeta) - \alpha\zeta' M(\alpha+1;\beta+1;-\zeta)}{\beta M(\alpha;\beta;-\zeta)}\Bigg]\end{aligned} \quad (2.7)$$

where ψ is the digamma function, and

$$G(\alpha;\beta;-\zeta) \equiv \sum_{n=1}^{\infty}\left(\alpha' H_n^{\alpha} - \beta' H_n^{\beta}\right)\frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^n}{n!}$$

$$H_n^{\alpha} \equiv \sum_{k=0}^{n-1}\frac{1}{\alpha+k}, \; H_n^{\beta} \equiv \sum_{k=0}^{n-1}\frac{1}{\beta+k},$$

Proof. See Appendix. □

Note that if α and β are constant with respect to ω, as is the case when we differentiate the characteristic function with respect to either η or v, we end up with a significantly simpler form.

Corollary 2.2. *The gradient of the characteristic function $\phi(\mathbf{\Theta}; u)$ with respect to $\mathbf{\Theta} = [\kappa, \eta, v, \sigma, \rho]$ is given by*

$$\nabla_{\Theta}\phi = [\Omega_\kappa, \Omega_\eta, \Omega_v, \Omega_\sigma, \Omega_\rho]$$

where $\Omega_\kappa, \ldots, \Omega_\rho$ are defined in theorem 2.1, and the intermediate derivatives of α, β and ζ with respect to the parameters are given by

$$\frac{\partial\alpha}{\partial\sigma} = \frac{2\kappa}{\sigma^3} - \frac{i\rho u}{\sigma^2} + \frac{\left(\frac{-2\kappa}{\sigma^3} + \frac{i\rho u}{\sigma^2}\right)\left(\frac{1}{2} - \frac{p}{\sigma^2}\right) - \frac{2q}{\sigma^3}}{\sqrt{\left(\frac{1}{2} - \frac{p}{\sigma^2}\right)^2 + \frac{2q}{\sigma^2}}}$$

$$\frac{\partial\alpha}{\partial\kappa} = -\frac{1}{\sigma^2} + \frac{\frac{1}{2\sigma^2} - \frac{p}{\sigma^4}}{\sqrt{\left(\frac{1}{2} - \frac{p}{\sigma^2}\right)^2 + \frac{2q}{\sigma^2}}}$$

$$\frac{\partial \alpha}{\partial \rho} = \frac{iu}{\sigma} + \frac{\frac{-iu}{2\sigma} + \frac{iup}{\sigma^3}}{\sqrt{\left(\frac{1}{2} - \frac{p}{\sigma^2}\right)^2 + \frac{2q}{\sigma^2}}}$$

$$\frac{\partial \beta}{\partial \sigma} = 2\left(\frac{\partial \alpha}{\partial \sigma} - \frac{2\kappa}{\sigma^3} + \frac{i\rho u}{\sigma^2}\right)$$

$$\frac{\partial \beta}{\partial \kappa} = 2\left(\frac{\partial \alpha}{\partial \kappa} + \frac{1}{\sigma^2}\right)$$

$$\frac{\partial \beta}{\partial \rho} = 2\left(\frac{\partial \alpha}{\partial \rho} - \frac{iu}{\sigma}\right)$$

$$\frac{\partial \zeta}{\partial \sigma} = -\frac{2\zeta}{\sigma}$$

$$\frac{\partial \zeta}{\partial v} = -\frac{\zeta}{v}$$

$$\frac{\partial \zeta}{\partial \kappa} = -\left(\eta(T-t) - \frac{1}{\kappa}\right)\zeta - \frac{(T-t)v\sigma^2}{2\kappa}\zeta^2$$

$$\frac{\partial \zeta}{\partial \eta} = -\left(\kappa(T-t) - \frac{1}{\eta}\right)\zeta - \frac{(T-t)v\sigma^2}{2\eta}\zeta^2$$

with all other partial derivatives equal to zero.

2.2. Optimal Gradient Computation

In computational terms, the most naive approach to solving (2.1) is to apply an optimization algorithm with finite-difference gradient approximations, and where $C(\mathbf{\Theta}; (K,T)_i)$ is evaluated individually for each option i without any computational caching. In contrast, the Fast Fourier Transform method of Carr et al. (1999) computes the option price for an entire grid of strikes for a given maturity simultaneously. The integrand of the original inversion formulation proposed in Heston (1993) precludes the use of the FFT since a singularity is present at $u = 0$. To address this, Carr et al. (1999) proposed the damped formulation

$$\psi_T(u) = \frac{e^{-r(T-t)}\phi(u - i(\gamma+1))}{\gamma^2 + \gamma - u^2 + i(2\gamma+1)u}$$

where T is the maturity of the option set, and γ is a damping coefficient, which needs to be chosen by hand.[4] Given $\psi_T(\cdot)$, the option price is given by

$$C(\boldsymbol{\Theta}; K_u, T) \approx \frac{e^{-\gamma k_u}}{\pi} \sum_{j=1}^{N} \delta_j e^{-i\lambda\mu(j-1)(u-1)} e^{ibv_j} \psi_T(v_j)\mu \tag{2.8}$$

where $k_u = -b + \lambda(u-1)$, $v_j = \mu(j-1)$, $b = \frac{1}{2}N\lambda$, $K_u = e^{k_u}$, and $\delta_j = 0.5$ for $j = 1, N$, and $\delta_j = 1$ otherwise.

In words, λ is the distance between consecutive log strike points, μ is the distance between consecutive integration nodes, and $-b$ and b are the lower and upper limits on the log strike range, respectively. Using the FFT to compute the inversion, we set $\lambda\mu = \frac{2\pi}{N}$. This implies a tradeoff between the resolution of the integration grid, and the resolution of the strike price grid.

The FFT method has proven useful for a variety of interesting applications in quantitative finance, see e.g. (Albanese et al., 2004; Chen et al., 2014; Fusai et al., 2016). However, a major drawback of the method in our setting is that a very large number of integration nodes are required to obtain a satisfactory level of accuracy, since the integration nodes need to be equally spaced and that prohibits the use of genuinely efficient integration schemes. Furthermore, we end up with a far greater number of option prices for each maturity than for which we could hope to have any use for most calibration purposes. This is because strike prices far in or out of the money tend to be very illiquid. In other words, most of the option prices we obtain from the FFT are likely matched to noise in the market data which we would prefer to keep out of the calibration.

A different method of tackling the inversion problem was introduced by Fang et al. (2008). It is based on a Fourier-cosine expansion and is generally referred to as the COS method. Here, the inversion integral is replaced by its cosine series expansion, with the series coefficients extracted directly from the integrand. Using this method, we have that

$$C(\boldsymbol{\Theta}; K, T) \approx e^{-r(T-t)} \sum_{j=1}^{N} \delta_j \mathrm{Re}\left\{ \phi\left(\frac{j\pi}{b-a}\right) e^{-ij\pi\frac{a}{b-a}} \right\} V_j \tag{2.9}$$

where a and b are the lower and upper endpoints of the truncated interval of support for the density function, and V_j depends on the payoff of the derivative. For plain vanilla options, V_j can

[4] The choice of this parameter has a significant impact on the accuracy of the option prices produced by the model. Furthermore, the optimal value of this parameter depends on the option maturity, so ideally a different damping parameter value is required for each maturity present in the set of benchmark options when calibrating the model.

be obtained in closed form. Define

$$\chi_j(c,d) \equiv \frac{1}{1+\left(\frac{j\pi}{b-a}\right)^2}\left[\cos\left(j\pi\frac{d-a}{b-a}\right)e^d - \cos\left(j\pi\frac{c-a}{b-a}\right)e^c \right. \\ \left. + \frac{j\pi}{b-a}\sin\left(j\pi\frac{d-a}{b-a}\right)e^d - \frac{j\pi}{b-a}\sin\left(j\pi\frac{c-a}{b-a}\right)e^c\right] \quad (2.10)$$

and

$$\psi_j(c,d) \equiv \begin{cases} \left[\sin\left(j\pi\frac{d-a}{b-a}\right) - \sin\left(j\pi\frac{c-a}{b-a}\right)\right]\frac{b-a}{j\pi} & \text{if } j \neq 0, \\ (d-c), & \text{if } j = 0 \end{cases}$$

The coefficients for a call and a put option for a given strike are given, respectively, by

$$V_j^{\text{call}} = \frac{2}{b-a}K(\chi_j(0,b) - \psi_j(0,b))$$

and

$$V_j^{\text{put}} = \frac{2}{b-a}K(-\chi_j(a,0) + \psi_j(a,0))$$

The COS method has been reported to be around 20-40 times faster (Fang et al., 2008) than the FFT method[5]. Its efficiency depends in part on the maturity, with longer maturities requiring a lower N than shorter maturities. Another factor is the choice of the truncation, i.e. the interval $[a,b]$. In Fang et al. (2008), the following is proposed for the Heston model for maturities T between 0.1 and 10.0:

$$[a,b] \equiv \left[c_1 - L\sqrt{c_2+\sqrt{c_4}}, c_1 + L\sqrt{c_2+\sqrt{c_4}}\right]$$

where c_n is the n-th cumulant of $\ln(S_T/K)$, and L is a control parameter which in Fang et al. (2008) is set equal to 10.

A somewhat simpler method, which is of similar computational efficiency as the COS method, is proposed in Kilin (2011). By noting that the characteristic function ϕ does not depend on the strike price K, we see that once we have computed the inversion of ϕ for a given option i we can reuse the computed values of ϕ to quickly compute (2.5) for the remainder of the options that have the same maturity T as i but different strike prices. In addition, we can couple this approach with an efficient integration scheme, like Gaussian quadrature.

[5] We find this difference to be smaller with a more streamlined implementation of the FFT. We would like to thank an anonymous referee for providing us with the ideas behind making the FFT faster.

To be more specific, a direct Fourier inversion of the characteristic function in (2.5) means calculating a discrete approximation to the integral in (2.5), which we can write as

$$\sum_{j=1}^{J} b_j \left\{ \frac{\left(\mathrm{Re}(\phi(u_j)) + \frac{\mathrm{Im}(\phi(u_j))}{u_j}\right)\cos(u_j k) + \left(\mathrm{Im}(\phi(u_j)) - \frac{\mathrm{Re}(\phi(u_j))}{u_j}\right)\sin(u_j k)}{1 + u_j^2} \right\} \tag{2.11}$$

where $b_1, \ldots, b_J$ are the integration weights, and are determined by the integration scheme, along with the discretization $u_1, \ldots, u_J$ of the integrand domain.

In a similar fashion, the numerical inversion of the analytic gradient of the characteristic function can be written as

$$\sum_{j=1}^{J} b_j \left\{ \frac{\left(\mathrm{Re}(\nabla_\Theta \phi(u_j)) + \frac{\mathrm{Im}(\nabla_\Theta \phi(u_j))}{u_j}\right)\cos(u_j k) + \left(\mathrm{Im}(\nabla_\Theta \phi(u_j)) - \frac{\mathrm{Re}(\nabla_\Theta \phi(u_j))}{u_j}\right)\sin(u_j k)}{1 + u_j^2} \right\} \tag{2.12}$$

We first give an outline of the technique of computing the prices of a set of vanilla options through caching the characteristic function calculations for each maturity without the use of an analytic gradient, as described in Kilin (2011). We refer to this method hereafter as strike caching (SC). Let the set of different maturities present in the set of options be denoted by $\boldsymbol{T} = \{T_s\}_{s=1}^{S}$, and for a given maturity T_s in this set, let $\boldsymbol{K}^s = \{K_r^s\}_{r=1}^{R_s}$ be the set of different strike prices present in the set of benchmark options with maturity T_s.

The method works for a given maturity T_s by calculating and caching the grid $\boldsymbol{A} = [A_1, \ldots, A_J]$ where $A_j = \phi(u_j)$ for $j = 1, \ldots, J$. This grid is then used for the computation of (2.11) for each strike K_r^s, which reduces the number of calls to ϕ by a factor equal to the number of strikes for each maturity. To make this work, we need to use the same discretization grid for every option with maturity T_s.

If we use adaptive quadrature to carry out the inversion, the discretization grid $\boldsymbol{U} = [u_1, \ldots, u_J]$, and the corresponding integration weights $\boldsymbol{B} = [b_1, \ldots, b_J]$, can be obtained simply through keeping track of the node-weight pairs generated by the quadrature from the calculation of the first option of each maturity. Alternatively, we can create the discretization in advance by standard interpolation methods. The application of Algorithm 1 to a finite difference gradient based minimization of (2.1) is straightforward. Let $\boldsymbol{e} = [1, \ldots, 1]$ be a vector of the same dimension as $\boldsymbol{\Theta}$, and let h be a sufficiently small, positive number. For a central finite difference we simply evaluate $SC(\cdot)$ for the parameter vectors $\boldsymbol{\Theta} + h\boldsymbol{e}$ and $\boldsymbol{\Theta} - h\boldsymbol{e}$, and then divide the difference of the two return values with $2h$, to obtain the gradient of (2.1) at $\boldsymbol{\Theta}$.

The strike caching approach, of course, works just as well for efficiently computing the analytic gradient of the characteristic function. To gain further performance improvements, however, we

Algorithm 1 Price Computation with Strike Caching

```
procedure SC($\mathbf{\Theta}, \{(K,T)_i\}_{i=1}^{I}$)
    for $s = 1, \ldots, S$ do
        Compute $\boldsymbol{U}$ and $\boldsymbol{B}$
        for $j = 1, \ldots, J$ do
            $A_j \leftarrow \phi_{T_s}(\mathbf{\Theta}; u_j)$
        end for
        for $r = 1, \ldots, R^s$ do
            Compute (2.11) with $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{U}$
            $C_i \leftarrow C(\mathbf{\Theta}; K_r^s, T_s)$ using (2.5)
        end for
    end for
    return $\sum_{i=1}^{I} w_i \left(C_i - C_i^{Market}\right)^2$
end procedure
```

extend the caching concept, which so far has been limited to the strike price dimension, to include the maturity dimension as well.

While the characteristic function does not depend on the option strike price, which makes the strike caching technique straightforward, it does indeed depend on the option maturity. However, if we take a closer look at ϕ we see that only ζ depends on the maturity. So, calculating $\phi(\mathbf{\Theta}; u_j)$ simultaneously for the entire set $\boldsymbol{T}$ while keeping u_j fixed enables us to significantly reduce the overall computational load, since α and β remain the same for each $T_s \in \boldsymbol{T}$.

Coupling this approach with the computation of the analytic gradient we presented in the previous section yields an even greater speedup of the calibration procedure. As is apparent from Theorem 2.1 and its corollary, the partial derivatives of the gradient of the characteristic function of the 3/2 model have a great deal of overlap. This means we can use the results from the computation of one partial derivative to help us calculate the remaining ones.

The application of the strike-maturity caching technique to the calibration problem with the analytic gradient of ϕ will be referred to hereafter as the Gradient-Maturity-Strike Caching method (GMSC).

Let $\mathbf{\Xi}_{\alpha,\beta}$ denote the Jacobian of (α, β) with respect to the set of calibration parameters, i.e.

$$\mathbf{\Xi}_{\alpha,\beta} = \begin{bmatrix} \dfrac{\partial \alpha}{\partial \omega_1} & \cdots & \dfrac{\partial \alpha}{\partial \omega_5} \\ \dfrac{\partial \beta}{\partial \omega_1} & \cdots & \dfrac{\partial \beta}{\partial \omega_5} \end{bmatrix}$$

where $\omega_i \in \{\kappa, \eta, v, \sigma, \rho\}$. Similarly, let $\mathbf{\Xi}_{\zeta}$ denote the Jacobian of $\zeta_1, \ldots, \zeta_S$ with respect to the model parameters, where ζ_s, for $s = 1, \dots, S$, is calculated using maturity $T_s \in \boldsymbol{T}$, i.e.

$$\boldsymbol{\Xi}_{\boldsymbol{\zeta}} = \begin{bmatrix} \dfrac{\partial \zeta_1}{\partial \omega_1} & \cdots & \dfrac{\partial \zeta_1}{\partial \omega_5} \\ \vdots & \ddots & \\ \dfrac{\partial \zeta_S}{\partial \omega_1} & \cdots & \dfrac{\partial \zeta_S}{\partial \omega_5} \end{bmatrix}$$

Then the GMSC method is given by Algorithm 2.

Algorithm 2 Gradient-Maturity-Strike Caching

```
 1: procedure GMSC($\boldsymbol{\Theta}, \{(K,T)_i\}_{i=1}^{I}$)
 2:     Compute $\boldsymbol{U}$ and $\boldsymbol{B}$
 3:     Compute $\boldsymbol{\zeta}$ and $\Xi_{\boldsymbol{\zeta}}$
 4:     for $j = 1, \dots, J$ do
 5:         Compute $\alpha(u_j), \beta(u_j)$, and $\Xi_{\alpha,\beta}(u_j)$
 6:         for $s = 1, \dots, S$ do
 7:             $A_{s,j} \leftarrow \phi_{T_s}(u_j)$
 8:             for $g = 1, \dots, 5$ do
 9:                 $G_{s,j,g} \leftarrow \frac{\partial \phi_{T_s}(u_j)}{\partial \omega_g}$
10:             end for
11:         end for
12:     end for
13:     for $s = 1, \dots, S$ do
14:         for $r = 1, \dots, R$ do
15:             Compute (2.11) with $\phi_{T_s}(u_j) = A_{j,s}$ for $j = 1, \dots, J$
16:             $C_i \leftarrow C(\boldsymbol{\Theta}; K_r, T_s)$ using (2.5)
17:             for $g = 1, \dots, 5$ do
18:                 Compute (2.12) with $\frac{\partial \phi_{T_s}(u_j)}{\partial \omega_g} = G_{j,s,g}$ for $j = 1, \dots, J$
19:                 $D_{i,g} \leftarrow \frac{\partial C(\boldsymbol{\Theta}; K_r, T_s)}{\partial \omega_g}$ using (2.6)
20:             end for
21:         end for
22:     end for
23:     return $\left[\sum_{i=1}^{I} w_i \left(C_i - C_i^{Market}\right) D_{i,1}, \dots, \sum_{i=1}^{I} w_i \left(C_i - C_i^{Market}\right) D_{i,5}\right]$
24: end procedure
```

The order in which the calculations are done in Algorithm 2 is designed to take advantage of cached intermediate results. In line 2 we compute the integration nodes for the transform inversion, and in line 3 we compute ζ and $\boldsymbol{\Xi}_{\boldsymbol{\zeta}}$, which are independent of the integration nodes. In line 5 we compute α,β and $\boldsymbol{\Xi}_{\boldsymbol{\alpha},\boldsymbol{\beta}}$, that are independent of the option maturity. In lines 6-10 we calculate the grid that consists of the characteristic function values for integration nodes $u_1, \dots, u_J$, as well as the partial derivatives of the characteristic function in those nodes. The characteristic function evaluation can be reused for the gradient computation, and is returned along with the gradient in our actual implementation, which speeds up the optimization procedure as well. Once we have computed the characteristic function and its gradient for the grid values $\boldsymbol{U}$, we use these stored values to calculate the integral required in (2.5) and (2.6), for the different strikes and maturities present in the set of benchmark options, as written in lines 13-22. Finally, the algorithm returns the vector of partial derivatives of the objective function with respect to the model parameters and spot volatility.

3. Regularization with Risk-Neutral MCMC Estimation

The objective function that corresponds to the nonlinear least squares formulation given by (2.1) turns out to exhibit numerically signs of being both non-convex and ill-conditioned, as we discuss in greater detail in section 4. One approach to addressing this issue is to regularize (2.1) with a penalty function. In other words, we change the problem from (2.1) to the regularized version

$$\boldsymbol{\Theta}^* = \underset{\boldsymbol{\Theta}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{I} w_i f_i(\boldsymbol{\Theta})^2 + \chi H(\check{\boldsymbol{\Theta}} - \hat{\boldsymbol{\Theta}}) \right\} \tag{3.1}$$

Here $\hat{\boldsymbol{\Theta}} = (\hat{\omega}_1, \dots, \hat{\omega}_r)$, with $r \leq |\boldsymbol{\Theta}|$, is a regularization point for the parameter vector $\check{\boldsymbol{\Theta}} = (\omega_1, \dots, \omega_r)$, with $\omega_i \in \boldsymbol{\Theta}$ for $i = 1, \dots, r$. Furthermore, H is a function that penalizes solutions to (3.1) in proportion to how far $\check{\boldsymbol{\Theta}}$ is from the regularization point, $\hat{\boldsymbol{\Theta}}$. Finally, χ is the regularization coefficient which determines how severely the penalty impacts the calibration. When χ is set large enough, the objective function becomes convex in $\check{\boldsymbol{\Theta}}$. If it is too high, however, potentially good solutions outside the immediate vicinity of the regularization point are ignored. The question then arises which χ, $\hat{\boldsymbol{\Theta}}$, and H we should choose.

One obvious choice for $\hat{\boldsymbol{\Theta}}$ is a historical estimate of the proper model parameters (Cont et al., 2004). In other words, we estimate $\{\kappa, \eta, \sigma, \rho\}$ using the time series of prices of the underlying realized over time. On a conceptual level, the reason why we want historical data in the calibration is to make the calibration process more robust against the noise inherent in option prices. This noise comes from sources such as the bid-ask spreads, ”easy number pricing” (i.e. the market prices of options tend to be given in convenient numbers at the expense of accuracy), and asynchronous pricing in end-of-day data. Option prices are therefore only an imperfect proxy for the market pricing measure.

When the estimation of the model parameters is given by (2.1), the parameter values we obtain correspond to the risk neutral measure, since the benchmark options are themselves priced under that measure. Estimation of the parameters under the objective measure, however, involves a considerably different formulation of the problem, and requires a study of the historical realization of the price of the underlying. This is relatively straightforward for discrete-time models, such as GARCH and its variants, where maximum likelihood can be applied directly to the model. In contrast, the application of standard econometric techniques to continuous-time models are complicated by the fact that the density is usually not known for the model, so maximum likelihood is not directly applicable. In addition, the instantaneous volatility is assumed to be unobserved in the 3/2 model, which further complicates the estimation.

Several methods have been developed in recent years to tackle this problem for continuous-time stochastic volatility models, such as 'implied state' GMM (Pan, 2002), and simulated maximum likelihood (Pedersen, 1995; Santa-Clara, 2002). The approach we have chosen is Bayesian inference through the Markov Chain Monte Carlo (MCMC) method, see Johannes et al. (2009) for an introduction. Let $\{S_t\}_{t=0}^T$ be a realized time series of prices for an underlying asset, whose dynamics are given by (1.2), and let $Y_t = \ln S_t - \ln S_{t-1}$, and $\boldsymbol{Y} = \{Y_t\}_{t=1}^T$. Assume further, for the time being, that we do not observe the corresponding realization of the time series of instantaneous volatility, $\boldsymbol{V} = \{V_t\}_{t=1}^T$. Lastly, let $\tilde{\boldsymbol{\Theta}} = (\tilde{\omega}_1, \ldots, \tilde{\omega}_r)$ be the parameters we want to estimate using the historical data, interpreted in the conventional Bayesian sense (i.e. as stochastic variables). Estimating $\tilde{\boldsymbol{\Theta}}$ using the MCMC approach involves repeatedly drawing samples from the distribution given by the posterior distribution $p(\tilde{\boldsymbol{\Theta}}, \boldsymbol{V}|\boldsymbol{Y})$. More specifically, assume we have drawn n samples using the posterior distribution, then the $(n+1)$-th sample for $\tilde{\boldsymbol{\Theta}}$ is drawn in the following way,

1. Draw $\tilde{\Theta}_1^{(n+1)} \sim p\left(\tilde{\Theta}_1|\tilde{\Theta}_2^{(n)}, \tilde{\Theta}_3^{(n)}, \ldots, \tilde{\Theta}_r^{(n)}, \boldsymbol{V}^{(n)}\right)$
2. Draw $\tilde{\Theta}_2^{(n+1)} \sim p\left(\tilde{\Theta}_2|\tilde{\Theta}_1^{(n+1)}, \tilde{\Theta}_3^{(n)}, \ldots, \tilde{\Theta}_r^{(n)}, \boldsymbol{V}^{(n)}\right)$

$\vdots$

r. Draw $\tilde{\Theta}_r^{(n+1)} \sim p\left(\tilde{\Theta}_r|\tilde{\Theta}_1^{(n+1)}, \tilde{\Theta}_2^{(n+1)}, \ldots, \tilde{\Theta}_{r-1}^{(n+1)}, \boldsymbol{V}^{(n)}\right)$

With $\boldsymbol{V}$ unobserved we would then, in a similar fashion, also have to repeatedly sample the spot volatilities, using $p\left(V_t|\tilde{\boldsymbol{\Theta}}^{(n)}, V_1^{(n)}, \ldots, V_T^{(n)}\right)$ to generate $V_t^{(n+1)}$, for each $t = 1, \ldots, T$. As is (currently) the case for most stochastic volatility models, we do not have a tractable form for $p(V_t|\cdots)$, so an approximate procedure, such as the Metropolis-Hastings algorithm, is required to sample it. This is by far the most computationally challenging aspect of the MCMC estimation.

Note that the procedure outlined above only gives us estimates for the parameters of the model under the objective measure. This can be useful, for instance, if we are concerned with how well the model fits the realized price history of the underlying. However, (2.1) is formulated for option prices, which means the parameters we are after are under the risk neutral measure. This is not an issue for σ and ρ which, by Girsanov's theorem, stay the same under each measure, whereas η and κ change. To obtain the risk neutral version of η and κ we need to include option prices in the historical estimation as well to estimate the variance risk premium.

To make this point clear, assume that the market price of volatility risk is given by $\nu V^{1/2}$, i.e. it is linear in volatility[6], and let W_t^p and W_t^q be Brownian motions under $\mathcal{P}$ and $\mathcal{Q}$, respectively. To further clarify our discussion, denote with κ^p and κ^q the rate of mean reversion under $\mathcal{P}$ and $\mathcal{Q}$, respectively, and η^p and η^q the long run volatility under $\mathcal{P}$ and $\mathcal{Q}$, respectively (as mentioned above σ and ρ stay the same).

From Girsanov's theorem, we have that under a change of measure from $\mathcal{P}$ to $\mathcal{Q}$ the volatility process

$$dV_t = \kappa^p(\eta^p V_t - V_t^2)dt + \sigma V_t^{3/2} dW_t^p$$

becomes,

$$\begin{aligned} dV_t &= \kappa^p(\eta^p V_t - V_t^2)dt + \sigma V_t^{3/2} dW_t^q - \sigma\nu V_t^2 dt \\ &= (\kappa^p + \sigma\nu)\left(\frac{\kappa^p\eta^p}{\kappa^p + \sigma\nu} V_t - V_t^2\right) dt + \sigma V_t^{3/2} dW_t^q \\ &= \kappa^q(\eta^q V_t - V_t^2)dt + \sigma V_t^{3/2} dW_t^q \end{aligned}$$

As we can see from this derivation,

$$\kappa^q = \kappa^p + \sigma\nu \tag{3.2}$$

and

$$\eta^q = \frac{\kappa^p\eta^p}{\kappa^p + \sigma\nu} \tag{3.3}$$

so ν affects the rate of mean reversion and long run volatility when we change from $\mathcal{P}$ to $\mathcal{Q}$, which means that if we want historical regularization values for the full set of the model parameters under $\mathcal{Q}$ then we need to estimate ν.

[6] This is a standard simplifying assumption in both the Heston model and the 3/2 model, as it implies these models are the same under the objective measure and risk neutral measure, as well as having economic justification, see Heston (1993).

Estimation of the variance risk premium is in itself an important part of the study of incomplete markets, for which MCMC estimation is also suitable. There is, however, the issue that incorporating option pricing into the MCMC estimation is very computationally challenging. The intuition we present here is that because we are only interested in the historical estimates of the parameters of the 3/2 model under the risk neutral measure, we can simply cut out the intermediate step of estimating the parameters and volatility process under the objective measure altogether by using historical implied volatility as a proxy for the volatility process V_t^Q. In other words, instead of filtering V_t^P under $\mathcal{P}$ and jointly estimating ν by incorporating past option prices and using formulas (3.2) and (3.3) to find the historical estimates of κ and η under $\mathcal{Q}$, we simply take the implied volatility of short dated options as a proxy for V_t and jump straight into the estimation of the 3/2 model parameters under $\mathcal{Q}$.

While this idea can be implemented in several different ways, we use CBOEs volatility index VIX for our numerical tests. The VIX is the square root of the risk-neutral expectation of the S&P500 variance over the following 30 calendar days, as determined by the market.[7]

With the proxy for $\boldsymbol{V}$, we now only need to sample $p(\tilde{\boldsymbol{\Theta}}|\boldsymbol{Y}, \boldsymbol{V})$. And, as we show below, we can characterize the posterior distribution for each parameter in $\tilde{\boldsymbol{\Theta}}$ in closed form, which makes the MCMC estimation process of the historical data extremely fast.

We begin by discretizing (1.2) to obtain

$$\begin{aligned} Y_t &= \left(r - \frac{1}{2}V_{t-1}\right)\Delta t + \sqrt{\Delta t V_{t-1}}\epsilon_t^s \\ V_t &= V_{t-1} + V_{t-1}\kappa(\eta - V_{t-1})\Delta t + V_{t-1}^{3/2}\sqrt{\Delta t}\epsilon_t^v \end{aligned} \tag{3.4}$$

where $\epsilon_t^s \sim N(0,1)$ and $\epsilon_t^v \sim N(0,\sigma)$. Note that as we are treating the process (3.4) under the risk neutral measure, the drift of Y is given by the risk free rate, r. The equations above can be rewritten in terms of the errors, to give

$$\begin{aligned} \epsilon_t^s &= \frac{Y_t - r\Delta t + \frac{1}{2}V_{t-1}\Delta t}{\sqrt{V_{t-1}\Delta t}} \\ \epsilon_t^v &= \frac{V_t - V_{t-1} - V_{t-1}\kappa(\eta - V_{t-1})\Delta t}{V_{t-1}^{3/2}\Delta t} \end{aligned}$$

[7] In theory, we could choose a volatility index with any time horizon, and deduce the instantaneous volatility by solving $(\text{VIX}/100)^2 = \mathbb{E}^{\mathbb{Q}}\left[\int_t^T v_u du | V_t = v\right]$ for V_t. However, while we know the right hand side in closed form (Carr et al., 2007), the expression is complicated enough that we would in fact be better off simply including estimation of $\boldsymbol{V}$ in the MCMC procedure.

Following the setup of Li et al. (2006) for the MCMC analysis of affine jump-diffusion, we change variables and set $\psi = \rho\sigma$, and $\Omega = \sigma^2(1-\rho^2)$. This gives us

$$(\epsilon_t^s, \epsilon_t^v) \sim N\left((0,0), \begin{bmatrix} 1 & \psi \\ \psi & \psi^2 + \Omega \end{bmatrix}\right)$$

We now require the posterior distributions, $p(\mathbf{\Theta}|\boldsymbol{Y}, \boldsymbol{V})$, which we use to sample the parameter space. By Bayes' theorem,

$$p(\tilde{\omega}_i|\tilde{\mathbf{\Theta}}_{-i}, \boldsymbol{V}, \boldsymbol{Y}) \propto p(\boldsymbol{Y}, \boldsymbol{V}|\tilde{\omega}_i, \tilde{\mathbf{\Theta}}_{-i})p(\tilde{\omega}_i)$$

for $i = 1, \dots, 5$, where $\tilde{\mathbf{\Theta}}_{-i} \equiv \tilde{\mathbf{\Theta}} \setminus \{\tilde{\omega}_i\}$. The joint likelihood function for $(\boldsymbol{Y}, \boldsymbol{V})$ is given by

$$p(\boldsymbol{Y}, \boldsymbol{V}|\tilde{\mathbf{\Theta}}) = \frac{1}{\Omega^{T/2}}\left(\prod_{t=1}^{T}\frac{1}{V_{t-1}^2\Delta T}\right) \times \exp\left(-\frac{1}{2\Omega}\sum_{t=1}^{T}\left((\Omega+\psi^2)(\epsilon_t^s)^2 - 2\psi\epsilon_t^s\epsilon_t^v + (\epsilon_t^v)^2\right)\right)$$

Following standard results on the conjugate prior for a normal distribution with unknown mean and variance (Koch, 2007), the prior we choose for Ω is the inverse gamma distribution, i.e. $\Omega \sim \mathcal{IG}(\alpha_0, \beta_0)$, and for the prior for ψ conditioned on Ω we choose the normal distribution, i.e. $\psi|_\Omega \sim N(\psi_0, p_0\Omega)$.

With these priors, the posterior distribution for Ω is given by $\mathcal{IG}(\widehat{\alpha}, \widehat{\beta})$, where $\widehat{\alpha} = \frac{T}{2} + \alpha_0$, and

$$\widehat{\beta} = \beta_0 + \frac{1}{2}\sum_{t=1}^{T}(\epsilon_t^v)^2 + \frac{p_0\psi_0^2}{2} - \frac{p_0\psi_0 + \sum_{t=1}^{T}(\epsilon_t^s\epsilon_t^v)^2}{p_0 + \sum_{t=1}^{T}(\epsilon_t^2)^2}$$

The posterior conditional distribution for ψ given Ω is given by $N(\widehat{\psi}, \widehat{\sigma}_\psi^2)$, where

$$\widehat{\psi} = \frac{p_0 + \sum_{t=1}^{T}\epsilon_t^s\epsilon_t^v}{p_0 + \sum_{t=1}^{T}(\epsilon_t^s)^2}$$

and

$$\widehat{\sigma}_\psi^2 = \frac{\Omega}{p_0 + \sum_{t=1}^{T}(\epsilon_t^s)^2}$$

These posteriors turn out to be the same for the Heston model. The remaining estimators, i.e. for η and κ, however, turn out to be different.

The prior we choose for κ is the normal distribution, i.e. $\kappa \sim N(\kappa_0, \sigma_\kappa^2)$. The posterior is then given by $N(\widehat{\kappa}, \widehat{\sigma}_\kappa^2)$, where

$$\widehat{\kappa} = \frac{\widehat{\sigma}_\kappa^2}{\Omega}\sum_{t=1}^{T}\frac{(\eta - V_{t-1})((\eta - V_{t-1})V_{t-1}\Delta t - \psi(Y_t - r\Delta t + \frac{1}{2}V_{t-1}\Delta t) - 1)}{V_{t-1}} + \frac{\widehat{\sigma}_\kappa^2}{\Omega}\sum_{t=1}^{T}\frac{V_t(\eta - V_{t-1})}{V_{t-1}^2} + \frac{\widehat{\sigma}_\kappa^2\kappa_0}{\sigma_\kappa^2}$$

and

$$\widehat{\sigma}^2_\kappa = \frac{1}{\sum_{t=1}^T \frac{(V_{t-1} - \eta)^2 \Delta t}{\Omega V_{t-1}} + \frac{1}{\sigma^2_\kappa}}$$

Likewise, the prior we choose for η is the normal distribution, i.e. $\eta \sim N(\eta_0, \sigma^2_\eta)$, and the posterior is given by $N(\widehat{\eta}, \widehat{\sigma}^2_\eta)$, where

$$\widehat{\eta} = \frac{\widehat{\sigma}^2_\eta}{\Omega} \sum_{t=1}^T \frac{\kappa(\kappa V_{t-1}\Delta t - \psi(Y_t - r\Delta t + \frac{1}{2}V_{t-1}\Delta t) - 1)}{V_{t-1}} + \frac{\widehat{\sigma}^2_\eta}{\Omega} \sum_{t=1}^T \frac{V_t \kappa}{V_{t-1}^2} + \frac{\widehat{\sigma}^2_\eta \eta_0}{\sigma^2_\eta}$$

and

$$\widehat{\sigma}^2_\kappa = \frac{1}{\sum_{t=1}^T \frac{\kappa^2 \Delta t}{\Omega V_{t-1}} + \frac{1}{\sigma^2_\eta}}$$

These posterior distributions allow us to sample, and therewith estimate, the parameters κ, η, σ and ρ in terms of historical returns and the implied volatility proxy. One advantage of the MCMC estimation procedure is that we naturally get a gauge for the uncertainty over our parameter estimates. More specifically, from the samples we draw we can calculate the sample covariance matrix

$$\boldsymbol{Q} = \begin{bmatrix} q_{1,1} & \cdots & q_{1,4} \\ \vdots & \ddots & \\ q_{4,1} & \cdots & q_{4,4} \end{bmatrix}$$

where

$$q_{i,j} = \frac{1}{N-1} \sum_{n=1}^N (\tilde{\omega}_{i,n} - \overline{\omega}_i)(\tilde{\omega}_{j,n} - \overline{\omega}_j)$$

Here, $\tilde{\omega}_{i,n}$ and $\tilde{\omega}_{j,n}$ are the nth realizations of parameter i and j, respectively, and $\overline{\omega}_i$ and $\overline{\omega}_j$ are their respective sample averages.

We can then utilize this information to determine how to penalize deviations across the individual historical estimates, $\tilde{\omega}_1, \ldots, \tilde{\omega}_4$, of the model parameters during the calibration. For this purpose we have chosen the following parsimonious formulation for the penalty function H.

$$H(\breve{\boldsymbol{\Theta}} - \hat{\boldsymbol{\Theta}}) \equiv (\breve{\boldsymbol{\Theta}} - \hat{\boldsymbol{\Theta}})^T \boldsymbol{Q}^{-1} (\breve{\boldsymbol{\Theta}} - \hat{\boldsymbol{\Theta}})$$

For a linear least squares problem, this form of the penalty function H is often referred to as Tikhonov regularization (Tikhonov et al., 1979), and has been studied extensively in the literature.

One computational advantage of the MCMC method relevant to our regularization effort is that if we want to incorporate new data into a prior historical estimate, we do not have to repeat the estimation procedure for the entire data set. More specifically, let $\pi^0(\tilde{\mathbf{\Theta}})$ be a prior distribution for the model parameters. Given observations $\boldsymbol{Y} = \{Y_t\}_{t=1}^T$ and $\boldsymbol{V} = \{V_t\}_{t=1}^T$, let

$$\pi^T(\tilde{\mathbf{\Theta}}) \equiv p(\tilde{\mathbf{\Theta}}|\boldsymbol{V}, \boldsymbol{Y}) \propto \pi^0(\tilde{\mathbf{\Theta}})p(\boldsymbol{V}, \boldsymbol{Y}|\tilde{\mathbf{\Theta}})$$

denote the posterior distribution. If we then acquire an additional observation (Y_{T+1}, V_{T+1}), we have that

$$\pi^{T+1}(\tilde{\mathbf{\Theta}}) \propto \pi^T(\tilde{\mathbf{\Theta}})p(Y_{T+1}, V_{T+1}|\boldsymbol{V}, \boldsymbol{Y}, \tilde{\mathbf{\Theta}})$$

where $\pi^{T+1}(\tilde{\mathbf{\Theta}})$ is the updated posterior distribution. This means that as time progresses, maintaining an up-to-date historical estimate by adding price movements and volatilities as they get realized is reduced to sampling the parameter space for those incoming values.

Lastly, the choice of χ is done on case-by-case basis. No results exist so far, to our knowledge, that deal definitively with the issue. Roughly speaking, we are after a value for χ which is at least high enough to produce a convex objective function, but not so high that the parameter search space becomes degenerate. Our search for χ is discussed in more detail in section 4.

4. Numerical Results

Our numerical experiments tested the speed of the GMSC formulation proposed in section 2 and summarized in Algorithm 2, as well as the pure strike caching formulation of Kilin (2011) summarized in Algorithm 1, the COS method of Fang et al. (2008), and the Fast Fourier Transform method of Carr et al. (1999), for a fixed level of accuracy. For this part, we used option prices derived from the 3/2 model with randomly selected parameter values. We also tested the effects of regularization as described in section 3 and for this we used S&P500 option data in addition to realized historical returns of the S&P500. The calibration code was written in C/C++ with Alglib and GNU GSL using full compiler optimization (/O2,/Ot) with MSVC. The historical estimation code was written in Python with Numpy and Scipy. The tests were run on an Intel Core i5-4310U (2.0GHz and 2.6GHz) with 8 GB of memory.

For the first part of our numerical experiments, we timed the gradient computation of (2.1) using (i) the central finite difference gradient scheme with the FFT method of Carr et al, (ii) the central finite difference gradient scheme with the COS method, (iii) the central finite difference gradient scheme with strike caching, and (iv) the analytic formula for the gradient of the characteristic function where gradient values were cached across both the strike and maturity dimension. Each gradient evaluation was done using 100 different options; 10 strikes per maturity, for a total of

10 maturities per evaluation. The list of maturities consisted of the following: 15, 30, 45, 60, 90, 120, 150, 220, 270 and 365 days. The strikes were close to the money, with strike price increments $\Delta K = 5$ for $T \leq 45$; $\Delta K = 10$ for $45 < T \leq 120$; and $\Delta K = 20$ for $120 < T \leq 365$. The parameter values $\mathbf{\Theta}$ were randomly generated on the intervals given in Table 1.

Once a random value had been generated for $\mathbf{\Theta}$, each of the methods in (i)-(iv) was used to calculate the gradient of (2.1), and the CPU time for each was recorded. This procedure was repeated for 400 different parameter realizations. For the FFT method we used 2^{12} integration points with a spacing of 0.125. For the SC and GMSC methods we used Gauss-Legendre quadrature with 128 points. For the COS method we used the truncation for $[a, b]$ given in section 2.2, where the cumulants were calculated using numerical differentiation of the characteristic function. Furthermore, we used a fully vectorized form for the COS method as given in Fang et al. (2008). The error tolerance of the inversion component for both the vectorized COS method and the FFT method[8] was set at 10^{-6}. Lastly, the step size for the finite difference gradient approach was set to 64 bit floating point precision.

In addition, we timed the full nonlinear least squares calibration of the 3/2 model with box constraints using each of these Fourier inversion methods, using the same setup for generating the option data and the true parameter values as in the gradient tests, with the same parameter constraints as given in Table 1. The parameter values were randomly generated, as well as the starting point of the optimization, and then each of the methods (i)-(iv) used to solve the inverse problem of retrieving the realized parameter values (each method was tested using the same set of realized parameter values and initial guesses to ensure unbiased comparison). This procedure was repeated 100 times. The optimization routine was set to stop when the gradient changed less than 2/3 of 64 bit floating point precision between iterations. The summary statistics for these tests are given in Table 3.

The calibration converged to the true parameter values and terminated within the specified stopping condition in all 100 test cases for each of the methods (i)-(iv). The mean absolute difference between the least squares parameter estimates and the true parameter values is given in Table 4.

As can be seen from Tables 2 and 3, the GMSC clearly outperforms the other methods both

[8] More specifically, the loop calculating the summands of the vectorized inversion of (2.9), and the inversion of (2.8) was terminated as soon as five consecutive summands turned out to be below the error threshold in absolute value.

	min	max
κ	1.0	50.0
η	0.01	0.90
v	0.01	0.90
σ	5.0	50.0
ρ	-1.0	0.50

Table 1: Variable Constraints. These represent the intervals within which we constrained the optimization procedure for each parameter.

	FFT	COS	SC	GMSC
mean	2.621	0.6109	0.5514	0.03519
std	1.730	0.8214	0.5410	0.02320
max	12.79	4.331	3.918	0.1047
min	1.549	0.2988	0.3710	0.01500

Table 2: Computational times in seconds for the central finite difference gradient approximation calculations for the FFT method, the COS method, the strike caching method (SC), and for the analytic gradient with the gradient-maturity-strike caching method (GMSC).

in computing the gradient and in the calibration runs. We notice a slight decrease in the speed gap for the calibration tests, as compared to the pure gradient computations. This is because the implementation of the Levenberg-Marquardt procedure that we used required in some instances objective function evaluations independent of the gradient evaluations. Overall, the full calibration testing procedure is a somewhat noisier method of measurement than the pure gradient testing procedure, since it introduces the issue of finite accuracy of the finite difference gradient, which can have an implementation-specific effect on the convergence of the algorithm. Also, different implementations of a given optimization algorithm can involve varying memory access overhead which is difficult to control for. From Table 4 we see that the GMSC produces slightly more accurate estimates on average than the comparison methods. This is not a surprise since the stopping conditions for the optimization are set right at the cutoff for the accuracy of the central finite difference gradient. However, all the different calibration methods exhibit good accuracy.

For the second part of our numerical experiments, we tested the effect of the regularization on

	FFT	COS	SC	GMSC
mean	168.3	53.46	48.65	4.499
std	101.9	66.17	43.64	4.013
max	821.7	205.9	174.6	12.17
min	9.562	3.588	1.244	0.0658

Table 3: Computational times in seconds of the calibration runs for the FFT method, the COS method, the strike caching method (SC), and the gradient-maturity-strike caching method (GMSC).

	FFT	COS	SC	GMSC
$\lvert\kappa^{\dagger} - \kappa^{*}\rvert$	4.10×10^{-2}	3.05×10^{-2}	1.61×10^{-2}	5.16×10^{-3}
$\lvert\theta^{\dagger} - \theta^{*}\rvert$	4.73×10^{-4}	3.89×10^{-4}	3.66×10^{-4}	1.19×10^{-4}
$\lvert v^{\dagger} - v^{*}\rvert$	8.26×10^{-6}	1.50×10^{-5}	1.38×10^{-6}	6.80×10^{-7}
$\lvert\sigma^{\dagger} - \sigma^{*}\rvert$	1.20×10^{-3}	3.25×10^{-3}	9.44×10^{-4}	1.16×10^{-4}
$\lvert\rho^{\dagger} - \rho^{*}\rvert$	4.99×10^{-4}	6.05×10^{-4}	2.12×10^{-4}	8.63×10^{-5}

Table 4: Mean absolute difference between the model parameter estimates $(\kappa^{*}, \ldots, \rho^{*})$ from the least squares calibration and the true values $(\kappa^{\dagger}, \ldots, \rho^{\dagger})$ for the FFT method, the COS method, the strike caching method (SC), and the gradient-maturity-strike caching method (GMSC).

the objective function on market data. More specifically, instead of solving (2.1), we solved (3.1) for a data setup similar to that of the non-regularized tests. We performed our numerical tests on S&P500 option data from the 3rd of January 2017 to the 8th of March 2017, for a total of 44 trading days. The total number of options (SPX and SPXW) quoted during this period was 376865.

As in the first part, each calibration consisted of 100 benchmark options; 10 strikes per maturity, for a total of 10 maturities per trading day. The list of maturities consisted of approximately[9] the same maturities as listed for the speed tests. For each maturity, we chose the 5 out-of-the money call options and 5 out-of-the money put options with the highest trading volume. We also included

[9] In the event that less than ten liquid options were present for a given target maturity for a given trading day, we picked the closest maturity instead which fulfilled the liquid option criteria. In no instance was there a trading day in which we did not have exactly ten maturities, with ten strikes per maturity.

pricing error as an additional stopping criterion. More specifically, we added the condition that change in pricing error should exceed machine precision and that the pricing error should be strictly decreasing with each iteration.

Using this market data, we find signs of apparent non-convexity of the least squares problem to be prevalent when calibrating the 3/2 model. This is in contrast to our tests on simulated data. Another issue is that for many of the trading days, the calibration tends to return implausible parameter estimates when unconstrained. In the case of box constraints, the solution to (2.1) generally tended to be on one or more of the boundaries. To illustrate this, we solve (2.1) using three different starting points, each of which is close to a different boundary corner. As can be seen in Figure B.1 in Appendix B, the parameter estimates returned by the optimization routine for the different starting points differ greatly, implying that the objective function either contains strict local minima, or is so flat that the optimization fails to find the global optimum due to insufficient precision.

The reason we observe this behavior is due to the interaction between the model and the data. The calibration of option pricing models in the risk neutral world for indices like the S&P500 is dominated by close-to-the-money plain European options. The reason we use these options is because they are the most liquid financial derivatives available, which means they are the most accurate representation we have for the market pricing measure. The problem is that the prices of these particular options (unlike, say, path-dependent options) tend to be insensitive to changes in the parameters of stochastic volatility models like the 3/2 model. It is this lack of sensitivity that leads to an ill-conditioned Hessian of the least squares objective function, which poses numerical problems for a second order gradient-based minimization procedure like the Levenberg-Marquardt method. A consequence of an ill-conditioned Hessian is that noise in the data (see section 3 for a discussion on noise in option price data) can lead to outsized effects on the calibration results.

To address this problem, we calibrated the regularized version of the objective function, i.e. (3.1), using the penalty function proposed in section 3. As previously mentioned, the regularization parameter χ can be thought to represent how strongly the user of the pricing model believes in the historical estimate, as well as the user's preferences over model risk. In other words, this value cannot be derived objectively a priori. Instead, the value ultimately depends on the real world context in which it is being used.

For illustrative purposes, however, we describe the approach given in Cont et al. (2004) which builds on Morozov's discrepancy principle (Morozov, 1966) to calculate χ. It requires that we determine the intrinsic error ϵ_0 in the data we have. In our case that means the lower bound on the quadratic calibration error. An obvious source of intrinsic error in the option price data is the bid-ask spread. Let

$$\epsilon_0^2 = \sum_{i=1}^{N} w_i \left(C_i^{\text{Ask}} - C_i^{\text{Bid}} \right)^2$$

where w_i is defined as in (2.1). The basic intuition behind Morozov's discrepancy principle is that we expect that a successful least squares calibration of our model will yield an error

$$\epsilon(\chi)^2 = \min_{\mathbf{\Theta}} \left\{ \sum_{i=1}^{I} w_i f_i(\mathbf{\Theta})^2 + \chi H(\mathbf{\Theta} - \tilde{\mathbf{\Theta}}) \right\}$$

that is not too far away from ϵ_0^2. The problem then becomes that of solving $\epsilon(\chi) = \delta\epsilon_0$ for χ, where δ is some number larger than 1.

This equation can only be solved through iterative root methods. Each iteration requires solving (3.1), making this a rather expensive operation. For this reason we only solve this equation for the first trading day, and then use the resulting χ^* for the regularized calibration of the remaining days. We find that this generates very stable parameter estimates across trading days, while the calibration error remains close to the error of the non-regularized calibration for most of the trading days. In addition, restarting the calibration from different initial guesses for each trading day reveals that the regularization eliminates any sign of non-convexity.

We performed the initial historical estimation using the history of daily closing prices of the S&P500 index from the 2nd of January 2014 to the 2nd of January 2017 and the VIX from that same period. For the MCMC estimation we used the following parameter priors:

$$\psi \sim N\left(0, \frac{\Omega}{2}\right), \Omega \sim IG\left(2, \frac{1}{200}\right), \kappa \sim N(0,1), \eta \sim N(0,1)$$

where the distributions for κ and η were truncated at zero (making the posteriors also truncated at zero)[10]. These priors derive from those used in Li et al. (2006) and Eraker et al. (2003) for the estimation of the Heston model. The number of samples drawn was 20000, with a burn-in of 5000 iterations, and the total estimation procedure took roughly 18.3 seconds.

For each successive trading day for which we calibrated the 3/2 model we updated the historical estimates by the change in price and VIX value realized on that same day, using 2000 iterations

[10] These are uninformative priors and are completely dominated by the likelihood function due to the size of our data sample. As we verified numerically, changing the parameters of the prior distributions appeared to have no discernible effect on the historical estimates.

and no burn-in, since the previous estimate acts as an informative prior, as explained in section 3. The updating estimation took roughly 1.5 seconds on average.

The resulting historical estimates, $\tilde{\boldsymbol{\Theta}}$ and the corresponding covariance matrix Q for those estimates are given respectively by

$$\tilde{\boldsymbol{\Theta}} \equiv \begin{bmatrix} \kappa \\ \eta \\ \sigma \\ \rho \end{bmatrix} = \begin{bmatrix} 22.46 \\ 0.4947 \\ 22.72 \\ -0.9085 \end{bmatrix},$$

and

$$\boldsymbol{Q} = \begin{bmatrix} 2.702 & -0.007951 & 0.02372 & -0.0002194 \\ -0.007951 & 0.0001007 & -0.0003903 & 0.000004462 \\ 0.02372 & -0.0003903 & 0.3577 & -0.003051 \\ -0.0002194 & 0.000004462 & -0.003051 & 0.00002612 \end{bmatrix}$$

We used the bisection root finding method for calculating χ^*, with the bisection terminating once a search interval of length 0.0001 was reached. We set the initial interval for χ as $[0, 5]$, and the calibration weight, w_i, for option i to the reciprocal of its bid-ask spread, i.e.

$$w_i = \frac{1}{C_i^{\text{Ask}} - C_i^{\text{Bid}}}$$

for $i = 1, \ldots, I$. The corresponding calibration results are shown in Figures 2 and 3 in Appendix B.

The results in Figure 2 show that the parameter estimates obtained from the regularized calibration are far more stable across trading days than the estimates we obtain from the unregularized calibration. Furthermore, starting the regularized calibration from the different initial points described in the caption of Figure 1 all lead to the same solution, meaning that we could not detect any non-convexity with the objective function of the regularized calibration.

The results in Figure 3 show that the increased parameter stability is accompanied on average by a very modest decrease in the in-sample fit, as measured by the normalized mean squared absolute price error as well as the root mean squared implied volatility error. Overall, the trade-off between parameter stability and in-sample fit of the regularized version appears very favorable, making clear the usefulness of the regularization approach.

5. Conclusion and Discussion

In this paper we have derived the analytic gradient for the characteristic function of the 3/2 model of stochastic volatility and devised an algorithm that exploits its mathematical features to avoid redundant calculations when calibrating the 3/2 model to option data using the standard nonlinear least squares approach. We have shown how this can greatly speed up the calibration process. In addition, we have proposed a form of regularization for the calibration problem that uses MCMC estimation on historical data to produce both a regularization point, as well as a damping matrix which we use to produce a parsimonious L_2 penalty function.

In our numerical experiments, we compared our calibration algorithm to the FFT method of Carr and Madan, the COS method of Fang and Oosterlee, and the strike caching method of Kilin. We find that the method presented here outperforms these by a factor of roughly 37, 12 and 11, respectively. For researchers or practitioners looking to calibrate the 3/2 model, the method presented here is, to our knowledge, the fastest way to do it so far presented in the literature.

Our numerical experiments furthermore demonstrated that for the market data we had at our disposal, the least squares calibration of the 3/2 model leads to highly unstable parameter estimates, both with respect to the initial point of the optimization procedure, as well as across trading days. The regularization method proposed in this paper was shown to be an effective tool to deal with these issues, producing a very favorable ratio of parameter estimation stability to in-sample fit.

One aspect of characteristic function pricing which we have not delved into here is the issue of discontinuities due to branch cuts in the complex plane. This problem has attracted considerable attention in the case of the Heston model (see e.g. Cui et al. (2017); del Baño et al. (2010); Albrecher et al. (2007); Jäckel et al. (2005)). However, it remains an open problem in the case of the 3/2 model, and is of interest with respect to further study and refinement of this model.

Acknowledgement

The authors would like to thank the anonymous referees for providing helpful comments and suggestions. Their insight and comments led to a better presentation of the ideas expressed in this paper.

References

Abramowitz, M. Stegun, I (1964). Confluent Hypergeometric Functions, *Handbook of Mathematical Functions* (pp. 503-537). Dover Publications

Aichinger, M., & Binder, A. (2013). Characteristic Function Methods for Option Pricing. *A Workout in Computational Finance* (pp 193-209). John Wiley & Sons.

Albanese, C., Jackson, K., & Wiberg, P. (2004). A new Fourier transform algorithm for value-at-risk. *Quantitative Finance*, 4(3), 328-338.

Albrecher, H., Mayer, P., Schoutens, W., & Tistaert, J. (2007). The Little Heston Trap. *Wilmott Magazine*, (Jan), 83-92.

Attari, M. (2004). Option pricing using Fourier transforms: A numerically efficient simplification. *Working Paper.*

Bakshi, G., Ju, N., & Ou-Yang, H. (2006). Estimation of continuous-time models with an application to equity volatility dynamics. *Journal of Financial Economics*, 82(1), 227-249.

del Baño Rollin, S., Ferreiro-Castilla, A., & Utzet, F. (2010). On the density of log-spot in the Heston volatility model. *Stochastic Processes and their Applications*, 120(10), 2037-2063.

Bates, D. S. (1996). Jumps and stochastic volatility: Exchange rate processes implicit in deutsche mark options. *The Review of Financial Studies*, 9(1), 69-107.

Carr, P., & Madan, D. (1999). Option valuation using the fast Fourier transform. *Journal of Computational Finance*, 2(4), 61-73.

Carr, P., & Sun, J. (2007). A new approach for option pricing under stochastic volatility. *Review of Derivatives Research*, 10(2), 87-150.

Chen, M. Y., & Chen, B. T. (2014). Online fuzzy time series analysis based on entropy discretization and a Fast Fourier Transform. *Applied Soft Computing*, 14, 156-166.

Chourdakis, K. (2005). Option pricing using the fractional FFT. *Journal of Computational Finance*, 8(2), 1-18.

Cont, R., & Tankov, P. (2004). Nonparametric calibration of jump-diffusion option pricing models. *Journal of Computational Finance*, 7, 1-49.

Cui, Y., del Baño Rollin, S., & Germano, G. (2017). Full and fast calibration of the Heston stochastic volatility model. *European Journal of Operational Research*, 263(2), 625-638.

Drimus, G. G. (2012). Options on realized variance by transform methods: a non-affine stochastic volatility model. *Quantitative Finance*, 12(11), 1679-1694.

Eraker, B., Johannes, M., & Polson, N. (2003). The impact of jumps in volatility and returns. *The Journal of Finance*, 58(3), 1269-1300.

Fang, F., & Oosterlee, C. W. (2008). A novel pricing method for European options based on Fourier-cosine series expansions. *SIAM Journal on Scientific Computing*, 31(2), 826-848.

Fusai, G., Germano, G., & Marazzina, D. (2016). Spitzer identity, Wiener-Hopf factorization and pricing of discretely monitored exotic options. *European Journal of Operational Research*, 251(1), 124-134.

Grasselli, M. (2017). The 4/2 stochastic volatility model: a unified approach for the Heston and the 3/2 model. *Mathematical Finance*, 27(4), 1013-1034.

Griewank, A., & Walther, A. (2008). *Evaluating derivatives: principles and techniques of algorithmic differentiation*. Society for Industrial and Applied Mathematics.

Guillaume, F., & Schoutens, W. (2010). Use a reduced Heston or reduce the use of Heston?. *Wilmott Journal*, 2(4), 171-192.

Heston, S. L. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*, 6(2), 327-343.

Johannes, M., & Polson, N. (2009). MCMC methods for continuous-time financial econometrics. *Handbook of Financial Econometrics*, 2(1), 1-72.

Jäckel, P., Kahl, C.& (2005). Not-so-complex logarithms in the Heston model. *Wilmott magazine*, 19(9), 94-103.

Kilin, F. (2011). Accelerating the calibration of stochastic volatility models. *The Journal of Derivatives*, 18(3), 7-16.

Koch, K. R. (2007). *Introduction to Bayesian statistics*. Springer Science & Business Media.

Li, H., Wells, M. T., & Yu, C. L. (2006). A Bayesian analysis of return dynamics with Lévy jumps. *The Review of Financial Studies*, 21(5), 2345-2378.

Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2), 431-441.

Mikhailov, S., & Nögel, U. (2004). *Heston's stochastic volatility model: Implementation, calibration and some extensions.* John Wiley and Sons.

Morozov, V. A. (1966). On the solution of functional equations by the method of regularization. In Soviet Math. Dokl, 7(1), 414-417.

Pan, J. (2002). The jump-risk premia implicit in options: Evidence from an integrated time-series study. *Journal of Financial Economics*, 63(1), 3-50.

Pedersen, A. R. (1995). A new approach to maximum likelihood estimation for stochastic differential equations based on discrete observations. *Scandinavian Journal of Statistics*, 22(1), 55-71.

Poteshman, A. M. (1998). Estimating a general stochastic variance model from option prices. *manuscript, University of Chicago.*

Brandt, M. W., & Santa-Clara, P. (2002). Simulated likelihood estimation of diffusions with an application to exchange rate dynamics in incomplete markets. *Journal of Financial Economics*, 63(2), 161-210.

Shephard, N. G. (1991). From characteristic function to distribution function: a simple framework for the theory. *Econometric Theory*, 7(4), 519-529.

Sepp, A. (2008). Pricing Options on Realized Variance in Heston Model with Jumps in Returns and Volatility. *Journal of Computational Finance*, 11(4), 33-70.

Tikhonov, A. N., Arsenin, V. I., & John, F. (1979). Solutions of ill-posed problems. *SIAM Rev.*, 21(2), 266-267.

Wong, H. Y., & Lo, Y. W. (2009). Option pricing with mean reversion and stochastic volatility. *European Journal of Operational Research*, 197(1), 179-187.

Appendix A. Proof of Theorem 2.1

Proof. First, recall that the gamma function $\Gamma(x)$ and the digamma function

$$\psi(x) \equiv \frac{\Gamma'(x)}{\Gamma(x)}$$

have the following properties, respectively:

$$(x)_n = \frac{\Gamma(x+n)}{\Gamma(x)},$$

and

$$\psi(x+n) = \psi(x) + \sum_{k=0}^{n-1} \frac{1}{x+k}$$

With this in mind, we see that

$$\begin{aligned} \frac{\partial}{\partial\omega}\left[\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\right] &= \frac{(\beta'-\alpha')\psi(\beta-\alpha)\Gamma(\beta-\alpha)\Gamma(\beta) - \beta'\psi(\beta)\Gamma(\beta-\alpha)\Gamma(\beta)}{\Gamma(\beta)^2} \\ &= \frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\left((\beta'-\alpha')\psi(\beta-\alpha) - \beta'\psi(\beta)\right), \end{aligned}$$

and

$$\begin{aligned} & \frac{\partial}{\partial\omega}\left[\sum_{n=0}^{\infty} \frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^n}{n!}\right] \\ = \; & \frac{\partial}{\partial\omega}\left[\sum_{n=0}^{\infty} \frac{\Gamma(\alpha+n)}{\Gamma(\beta+n)}\frac{\Gamma(\beta)}{\Gamma(\alpha)}\frac{(-\zeta)^n}{n!}\right] \\ = \; & (\beta'\psi(\beta) - \alpha'\psi(\alpha))\left(M(\alpha,\beta;-\zeta) - 1\right) \\ & + \sum_{n=1}^{\infty}\left(\alpha'(\psi(\alpha) + H_n^{\alpha}) - \beta'(\psi(\beta) + H_n^{\beta})\right)\frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^n}{n!} - \sum_{n=0}^{\infty} n\zeta'\frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^{n-1}}{n!} \\ = \; & \sum_{n=1}^{\infty}\left(\alpha' H_n^{\alpha} - \beta' H_n^{\beta}\right)\frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^n}{n!} - \frac{\alpha}{\beta}\zeta' M(\alpha+1,\beta+1,-\zeta) \end{aligned}$$

Furthermore, we have that

$$\frac{\partial\zeta^{\alpha}}{\partial\omega} = \zeta^{\alpha}\left(\alpha'\ln(\zeta) + \frac{\alpha\zeta'}{\zeta}\right)$$

With these results, we have that the derivative of $\tilde{\phi}$ is given by

$$\begin{aligned}
\frac{\partial\tilde{\phi}}{\partial\omega} = &\ \frac{\partial}{\partial\omega}\left[e^{iux}\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\zeta^{\alpha}M\left(\alpha;\beta;-\zeta\right)\right] \\
= &\ e^{iux}\Bigg(\frac{\partial}{\partial\omega}\left[\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\right]\zeta^{\alpha}M(\alpha,\beta;-\zeta)+\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\frac{\partial\zeta^{\alpha}}{\partial\omega}M(\alpha,\beta;-\zeta) \\
&\ +\frac{\Gamma(\beta-\alpha)}{\Gamma(\beta)}\zeta^{\alpha}\frac{\partial}{\partial\omega}\left[M(\alpha,\beta;-\zeta)\right]\Bigg) \\
= &\ \big((\beta'-\alpha')\psi(\beta-\alpha)-\beta'\psi(\beta)\big)\,\tilde{\phi}+\left(\alpha'\ln(\zeta)+\frac{\alpha\zeta'}{\zeta}\right)\tilde{\phi} \\
&\ +\left(\sum_{n=1}^{\infty}\left(\alpha' H_n^{\alpha}-\beta' H_n^{\beta}\right)\frac{(\alpha)_n}{(\beta)_n}\frac{(-\zeta)^n}{n!}-\frac{\alpha}{\beta}\zeta' M(\alpha+1,\beta+1,-\zeta)\right)\frac{\tilde{\phi}}{M(\alpha,\beta;-\zeta)} \\
= &\ \tilde{\phi}\Bigg(\big((\beta'-\alpha')\psi(\beta-\alpha)-\beta'\psi(\beta)\big)+\left(\alpha'\ln(\zeta)+\frac{\alpha\zeta'}{\zeta}\right) \\
&\ +\frac{\beta G(\alpha,\beta;-\zeta)-\alpha\zeta' M(\alpha+1,\beta+1,-\zeta)}{\beta M(\alpha,\beta;-\zeta)}\Bigg)
\end{aligned}$$

□

Appendix B. Regularized Calibration Results

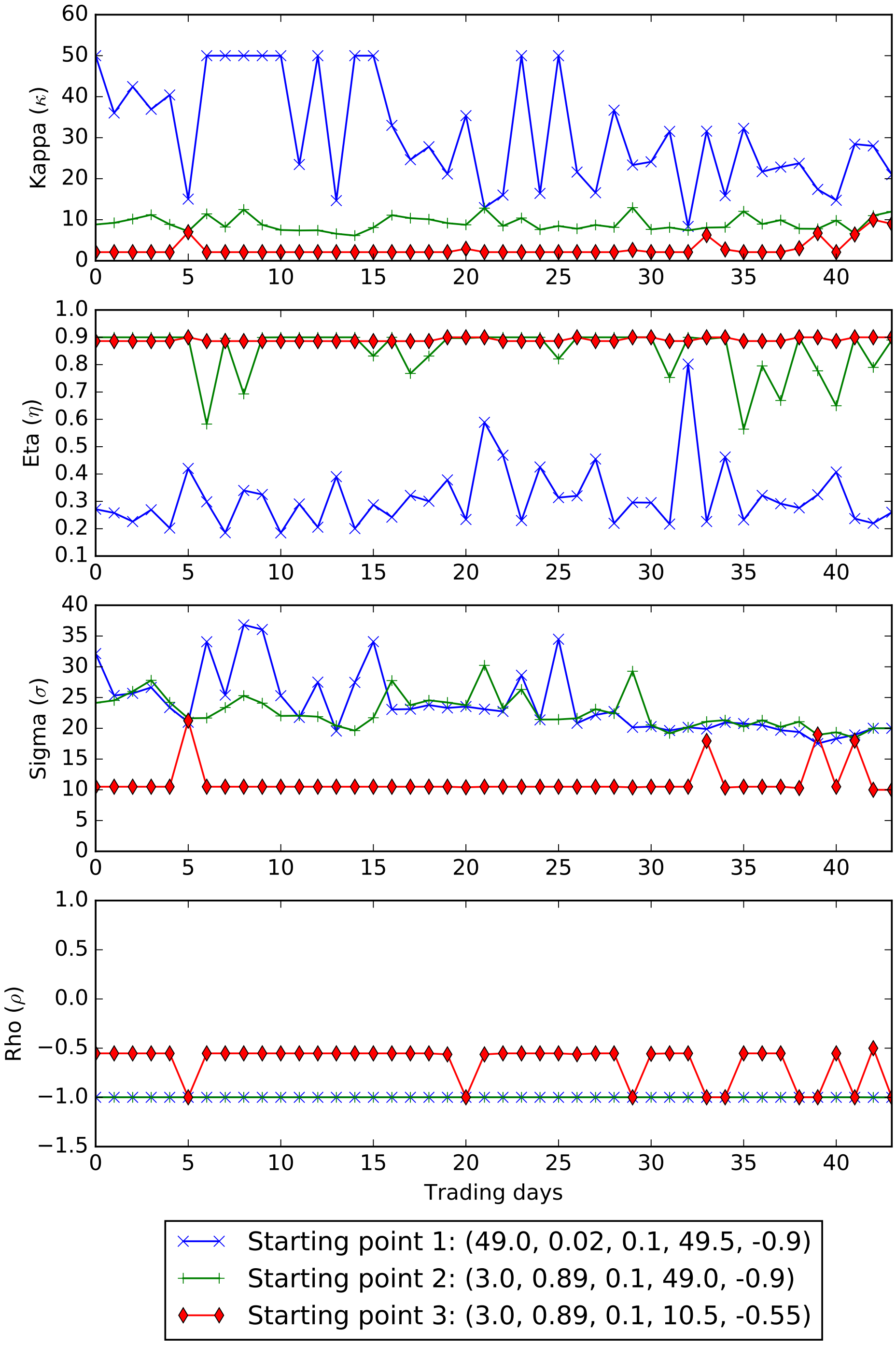

Figure B.1: Estimated parameter values obtained by solving (2.1) for S&P 500 option data for the first 44 trading days of 2017, using different starting points for the model parameters $\mathbf{\Theta} = (\kappa, \eta, v, \sigma, \rho)$.

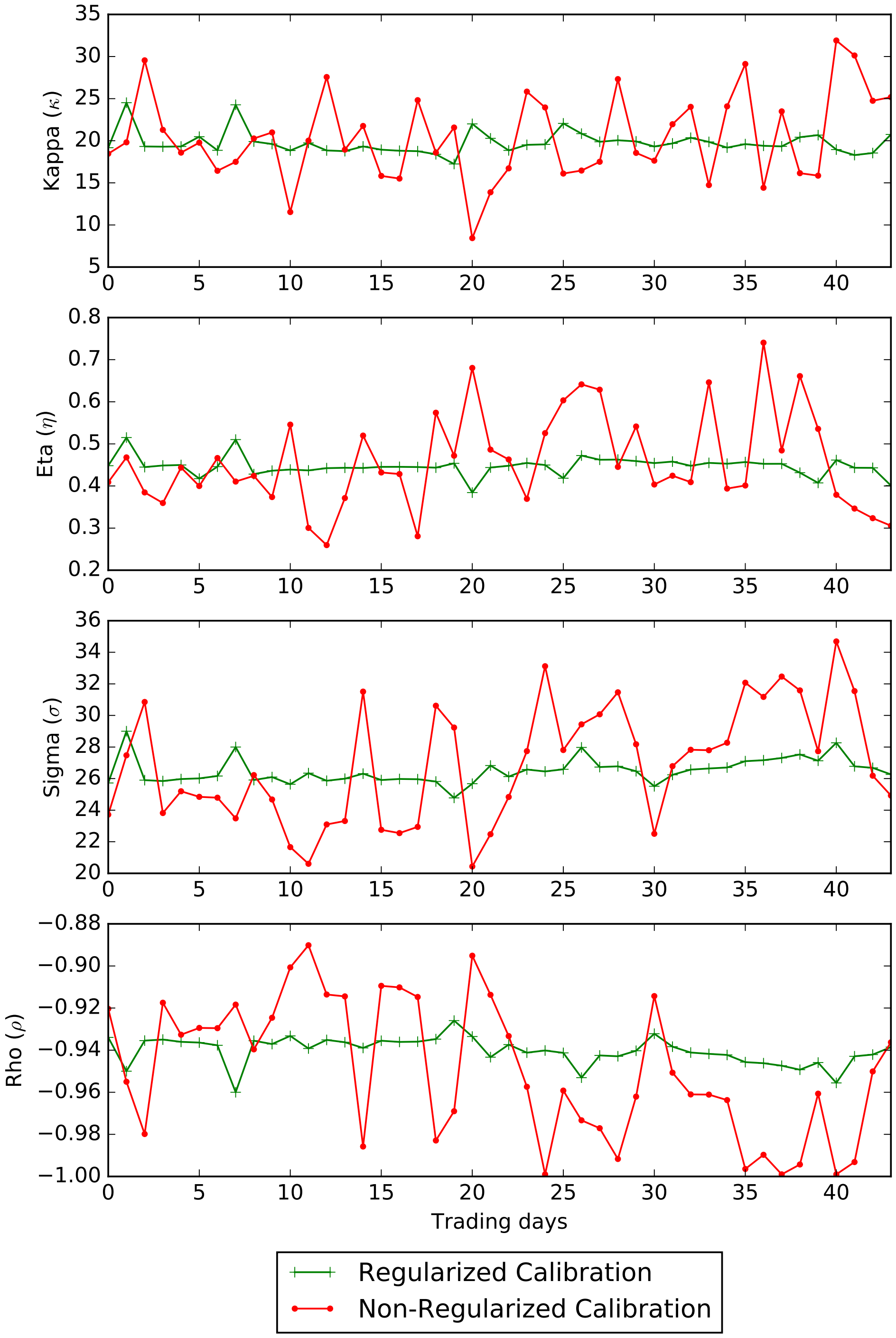

Figure B.2: Comparison of parameter stability for regularized ($\chi = 3.86$) and non-regularized calibration of the 3/2 model.

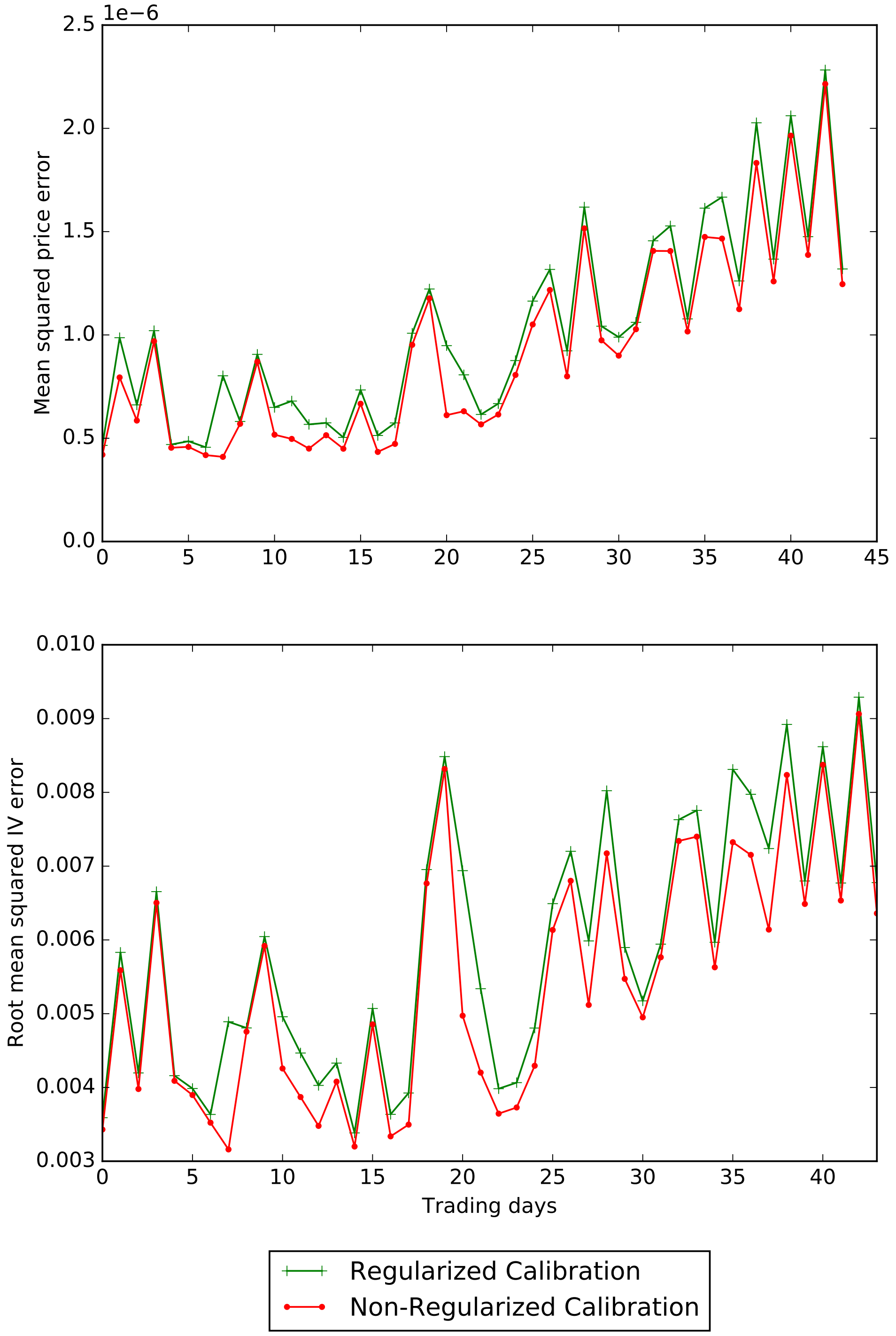

Figure B.3: Comparison of in-sample calibration error between regularized ($\chi = 3.86$) and non-regularized calibration of the 3/2 model.

Appendix C. Calibration Of Multi-Factor Models

In this appendix section, we show how the GMSC method for the 3/2 model can be used when the model is coupled with other stochastic factors for which we know the characteristic function in closed form. We begin by recalling a well known fact from characteristic function pricing. For the purpose of demonstration, consider the following general multi-factor log-price model,

$$dX_t = \sum_{m=1}^{M} dZ_{t,m}, \quad 0 \leq t \leq T, \tag{C.1}$$

where $Z_{t,m}$ is an adapted semimartingale characterized by the parameter set $\mathbf{\Theta}^{(m)}$ such that for each m, the characteristic function for the density of $Z_{T,m}$ is known in closed form and given by $\phi_m\left(\mathbf{\Theta}^{(m)}; u\right)$. If the factors $Z_{t,m}$ and $Z_{t,m'}$ are stochastically independent for any m and m' such that $m \neq m'$, then we have that the characteristic function for X_T is known in closed form and is given by

$$\phi\left(\mathbf{\Theta}; u\right) = \prod_{m=1}^{M} \phi_m\left(\mathbf{\Theta}^{(m)}; u\right) \tag{C.2}$$

where $\mathbf{\Theta} = \left[\mathbf{\Theta}^{(1)}, \ldots, \mathbf{\Theta}^{(M)}\right]$. Calibrating (C.1) through least squares minimization using gradient-based methods means calculating the equivalent of

$$\nabla_{\mathbf{\Theta}}\phi = \left[\frac{\phi}{\phi_1}\nabla_{\mathbf{\Theta}^{(1)}}\phi_1, \ldots, \frac{\phi}{\phi_M}\nabla_{\mathbf{\Theta}^{(M)}}\phi_M\right] \tag{C.3}$$

Therefore, the gradient can be calculated separately for each factor while minimizing the least squares objective function, which means that the computational burden of the factors that correspond to a 3/2 volatility specification can be efficiently reduced using the results given in this paper. Furthermore, this can be done while simultaneously using efficient gradient calculation techniques developed for other factors nested in the model. The method given in Cui et al. (2017), for example, can be used here for any of the M factors that correspond to the square root volatility specification of the Heston model.

We conclude our discussion with a numerical study of the efficiency of the gradient calculation methods proposed in this paper for a double 3/2 model (D32),

$$\begin{aligned} dS_t &= \mu S_t dt + \sqrt{V_{t,1}} S_t dB_{t,1} + \sqrt{V_{t,2}} S_t dB_{t,2} \\ dV_{t,1} &= \kappa_1(\eta_1 V_{t,1} - V_{t,1}^2)dt + \sigma_1 V_{t,1}^{3/2} dW_{t,1} \\ dV_{t,2} &= \kappa_2(\eta_2 V_{t,2} - V_{t,2}^2)dt + \sigma_2 V_{t,2}^{3/2} dW_{t,2} \end{aligned} \tag{C.4}$$

and a mixed 3/2-Heston model (H32),

$$\begin{aligned} dS_t &= \mu S_t dt + \sqrt{V_{t,1}} S_t dB_{t,1} + \sqrt{V_{t,2}} S_t dB_{t,2} \\ dV_{t,1} &= \kappa_1(\eta_1 V_{t,1} - V_{t,1}^2)dt + \sigma_1 V_{t,1}^{3/2} dW_{t,1} \\ dV_{t,2} &= \kappa_2(\eta_2 - V_{t,2})dt + \sigma_2 \sqrt{V_{t,2}} dW_{t,2} \end{aligned} \tag{C.5}$$

where W_1, W_2, B_1 and B_2 are standard Brownian motions such that B_1 and W_1 have instantaneous correlation ρ_1, and B_2 and W_2 have instantaneous correlation ρ_2, with all other factor correlations equal to zero.

The setup of our numerical tests on D32 and H32 is the same as for the simulated gradient and calibration speed tests for the one factor 3/2 model in section 4, with the exception that we focus solely on the strike caching method (SC) of Kilin and the GMSC. In addition, the true values for $\kappa_2, \eta_2, v_2, \sigma_2, \rho_2$ for the square root volatility process in H32 are chosen randomly from within the box constraints given in Table C.5.

	min	max
κ_2	0.1	10.0
η_2	0.01	0.90
v_2	0.01	0.90
σ_2	0.1	1.0
ρ_2	-1.0	0.50

Table C.5: Variable Constraints. These represent the intervals within which we constrained the generation of random parameter values for the square root volatility component of H32. All other parameter values were simulated using the box constraints given in Table 1.

The numerical results are given in Tables C.6 and C.7, for the gradient computations and calibration tests, respectively.

For our tests on D32 we use the GMSC with the analytic gradient derived in this paper separately for each 3/2 factor. More specifically, we compute C.3 by applying Algorithm 2 separately to $\nabla_{\mathbf{\Theta}^{(1)}}\phi_1$, and $\nabla_{\mathbf{\Theta}^{(2)}}\phi_2$, where ϕ_1 and ϕ_2 are the characteristic functions for single factor 3/2 volatility models with $\mathbf{\Theta}^{(1)} = [\kappa_1, \eta_1, v_1, \sigma_1, \rho_1]$ and $\mathbf{\Theta}^{(2)} = [\kappa_2, \eta_2, v_2, \sigma_2, \rho_2]$, respectively.

For H32 we denote with ϕ_1 the characteristic function for the one factor 3/2 volatility model, and with ϕ_2 the characteristic function for the one factor square root volatility model. We compute C.3 by applying Algorithm 2 to $\nabla_{\mathbf{\Theta}^{(1)}}\phi_1$, and by applying the gradient computation formulas presented in Cui et al. (2017) for the Heston model to $\nabla_{\mathbf{\Theta}^{(2)}}\phi_2$. We refer to this combined method as the mixed GMSC (MGMSC) in Tables C.6 and C.7.

	SC D32	GMSC D32	SC H32	MGMSC H32
mean	1.231	0.08375	0.6502	0.04807
std	0.8012	0.03913	0.5824	0.02796
max	6.429	0.1799	4.313	0.1271
min	0.9140	0.0415	0.4040	0.01668

Table C.6: Computational times in seconds for the analytic gradient computations with the gradient-maturity-strike caching method (GMSC) for the double 3/2 stochastic volatility model (D32), and with the GMSC coupled with the formula from Cui et al. (2017) (MGMSC) for the mixed 3/2-Heston volatility model (H32). The computational times in seconds for the central finite difference gradient approximation using strike caching (SC) are given as comparison for both models.

	SC D32	GMSC D32	SC H32	MGMSC H32
mean	701.2	58.18	95.67	8.302
std	665.0	47.85	79.15	6.224
max	2374	186.6	325.0	29.08
min	193.6	9.077	10.12	1.257

Table C.7: Computational times in seconds for the calibration runs using the analytic gradient with gradient-maturity-strike caching method (GMSC) for the double 3/2 stochastic volatility model (D32), and the GMSC coupled with the formula from Cui et al. (2017) (MGMSC) for the mixed 3/2-Heston volatility model (H32). The computational times in seconds for the calibration runs using the central finite difference gradient approximation and strike caching (SC) are given as comparison for both models.

As expected, the time it takes to compute the gradient for D32 is roughly twice the time it takes for calculating the gradient for the single factor 3/2 model for both the SC and the GMSC. In contrast, the computational cost of the calibration procedure for the multifactor models is significantly higher. This is to be expected as well, since the iteration count of the Newton-based minimization procedure grows superlinearly with the number of decision variables. This serves to highlight the necessity for numerically efficient calibration techniques for multi-factor models.