

Calibration of local volatility surfaces under PDE constraints

Love Lindholm

Abstract

The calibration of a local volatility surface to option market prices is an inverse problem that is ill-posed as a result of the relatively small number of observable market prices and the unsMOOTH nature of these prices in strike and maturity. We adopt the practice advanced by some authors to formulate this inverse problem as a least squares optimization under the constraint that option prices follow Dupire's partial differential equation. We develop two algorithms for performing the optimization: one based on techniques from optimal control theory and another in which a numerical quasi-Newton algorithm is directly applied to the objective function. Regularization of the problem enters easily in both problem formulations. The methods are tested on three months of daily option market quotes on two major equity indices. The local volatility surfaces resulting from both methods yield excellent replications of the observed market prices.

1 Introduction

Since the so called local volatility model was introduced in 1994 (Dupire [15], Derman and Kani [13]) it has become one of the most extensively used models in derivatives pricing across all asset classes. In the case of an equity stock or index S , the price dynamics in the local volatility model under the risk neutral measure are given as

$$dS_t = (r_t - q_t)S_t dt + \sigma(t, S_t)S_t dW_t, \quad (1)$$

where W_t is a Brownian motion, r_t is the risk free interest rate and q_t is a continuous dividend yield at time t . The squared local volatility σ^2 gives the instantaneous variance of the logarithm of S as a deterministic function of the time t and the spot value S_t . Under the dynamics (1), it can be shown [15], [13] that the prices of call options $C(T, K)$ of time to maturity T , strike K and a given value of the spot S_0 at time $t = 0$, can be related through a parabolic partial differential equation known as Dupire's equation,

$$\begin{aligned} \partial_T C(T, K) &= \frac{1}{2}\sigma^2(T, K)K^2\partial_{KK}C(T, K) - q_T C(T, K) - (r_T - q_T)K\partial_K C(T, K) \\ (T, K) &\in \mathbb{R}_+ \times \mathbb{R}_+, \\ C(0, K) &= (S_0 - K)_+, \quad C(T, 0) = S_0 e^{-\int_0^T q_t dt}, \quad C(T, \infty) = 0, \end{aligned} \quad (2)$$

from which the volatility function can be expressed in terms of option prices as

$$\sigma^2(T, K) = \frac{\partial_T C(T, K) + q_T C(T, K) + (r_T - q_T)K\partial_K C(T, K)}{\frac{1}{2}K^2\partial_{KK}C(T, K)}. \quad (3)$$

The model's popularity stems from this simple relation between option prices and the volatility function: given a surface of option prices $C : (T, K) \rightarrow \mathbb{R}_+$ that is differentiable in T and twice differentiable in K , the function σ can be retrieved from a mere differentiation of C . An appealing consequence of that observation is, of course, that with σ chosen according to (3), the partial differential equation (PDE) (2) tells us that an asset with the dynamics (1) will match all option prices $C(T, K)$.

The simplicity of the expression (3) is, however, somewhat illusive. In practice, market prices on options are not given as continuous, smooth surfaces, but as discrete values that can not obviously be seen to be sampled from a differentiable function. So even though the equation (3) gives a seemingly easy way of constructing the function σ from option prices, this is difficult in real life where we can only observe option prices at a finite number of maturity-strike pairs. This inverse problem of choosing a local volatility function that makes the model replicate observable market prices has been extensively treated by both practitioners and academics, but there is no consensus regarding what to use to solve it, and different methods are used in the daily activity at different financial institutes.

A tempting approach is to try to interpolate the observed market quotes by some smooth interpolant in (T, K) and then differentiate the resulting prices according to (3) to obtain the desired σ that reproduces the input prices. Unfortunately this approach is difficult in practice. Market prices are not necessarily smooth, so an interpolation and subsequent differentiation will lead to unstable results. Also, an interpolation will not guarantee that the ratio of derivatives defining σ^2 in (3) be positive. Even so, methods of this type are commonly used by market practitioners. It is then customary to represent option prices via their corresponding Black-Scholes volatility surface (see for example [6]), approximate the Black-Scholes volatility by some parametrization in (T, K) and obtain the local volatility from this representation. As an interesting example of this we refer to Sepp [34], in which a stochastic volatility model is used to parameterize the Black-Scholes volatility surface and subsequently calculate a local volatility function.

One way of understanding the ill-posed character of the problem of calibrating a local volatility to fit market data is to relate the typical number of observed quotes to the number of grid points needed in a discretization of Dupire's equation (2) in order to obtain prices with a meaningful precision. The precision by which we wish to reproduce prices by solving (2) is dictated by the bid-ask spread of observed market quotes. Let $\mathcal{T} = \{\bar{T}^i, 1 \leq i \leq M\}$ be the maturities at which prices are quoted and for each maturity T^i let $\mathcal{K}^i = \{\bar{K}_j^i, 1 \leq j \leq N_i\}$ be the quoted strikes, so that

$$\mathcal{I} = \{(\bar{T}^i, \bar{K}_j^i) : \bar{T}^i \in \mathcal{T}, \bar{K}_j^i \in \mathcal{K}^i, 1 \leq i \leq M\} \quad (4)$$

is the set of all quoted maturity-strike pairs. Let $\bar{C}^{\text{bid}}(T, K)$, $\bar{C}^{\text{ask}}(T, K)$ be the observable bid-and ask prices for each $(T, K) \in \mathcal{I}$. The bid-ask spread for a given pair $(T, K) \in \mathcal{I}$ is then simply

$$\text{spread}(\bar{T}, \bar{K}) := \bar{C}^{\text{ask}}(\bar{T}, \bar{K}) - \bar{C}^{\text{bid}}(\bar{T}, \bar{K}). \quad (5)$$

For the Euro Stoxx 50 Index - the most liquidly traded European equity index - the number of observable quotes $\#\mathcal{I}$ in the data sets we use is usually around 300 for around 9 distinct maturities up to between 2 and 4 years. In contrast, if we want to approximate a solution to (2) to a precision smaller than the spread by means of a finite difference scheme, the number of grid points needed in the discretization will be of order 10^5 . In other words, we basically need to estimate the unknown volatility σ at 300 different points (T, K) for each observable data point.

In this report, we take the approach to formulate the calibration of a local volatility function $\sigma(T, K)$ as an optimization problem. Let \bar{C} be a vector containing the market prices for the maturities and strikes in \mathcal{I} , and let C be the corresponding prices resulting from a given local volatility function σ . Given some suitable space Σ of positive, real valued functions on \mathbb{R}^2 , we then look for a function σ that solves the following minimization problem,

$$\begin{aligned} & \min_{\sigma \in \Sigma} \|C - \bar{C}\|^2 \\ & \text{subject to: } C, \sigma \text{ satisfy (2),} \end{aligned} \quad (6)$$

where $\|\cdot\|$ is some appropriate norm in \mathbb{R}^N with $N = \#\mathcal{I}$.

The problem (6) and variations thereof have been the subject of several studies. Lagnado and Osher [26] add a Tikhonov regularization to the objective function and solve the problem by a gradient method in which a PDE for the gradient is obtained and solved numerically. Jackson et

al [23] exploit the ideas of Lagnado and Osher to calculate a local volatility defined in a space Σ of spline functions and employ a quasi-Newton optimization technique rather than a gradient based algorithm. Coleman et al. [12] also let the space Σ consist of spline functions defined on a grid of but use a Jacobian based optimization algorithm in which the Jacobian is numerically evaluated. A similar approach is used by Glover and Ali in [20] but now with Σ consisting of a space of radial basis functions rather than splines. Achdou and Pironneau [1] regularize the objective function by adding penalties on the derivatives of the squared volatility σ^2 and find an equation for the gradient of the objective function which is used to solve the problem on a successively refined grid with a finite element method. Andreasen and Huge [2] develop a fast algorithm for solving (6) with a fully implicit finite difference scheme on a coarse grid (corresponding to the set \mathcal{I} of observable strikes and maturities) and use the result to interpolate prices in time between the maturities with observable data. (We will exploit the results from [2] in Section 3 below.) A method in which Dupire's equation (2) is not explicitly used is developed by Lipton and Sepp [27] who use a transform method to find a semi-analytical solution to (2) (without interest rates and dividends) for the special case of a volatility σ that is piecewise constant in (T, K) and use that solution to optimize σ in a "bootstrapping manner" (see Sections 2.2 and 4) to match data for one maturity at the time. Finally we mention an approach in which the objective function is not formulated in terms of a norm in the difference between model prices and market prices, which is developed by Avellaneda et al. [4] who pose the calibration of a local volatility to market data as the problem of minimizing a certain entropy of the volatility under the constraint that the model prices match market prices. This approach leads to a Lagrange-formulation and a resulting minimization problem in the Lagrangian multiplier that is solved with a gradient method.

Below we will use two different methodologies for solving the problem (6). First, in Section 2, we formulate (6) as an optimal control problem that we solve with techniques from Sandberg and Szepessy [32] based on a regularization of the resulting Hamiltonian system. This method leads to a set of non-linear equations - in the price C and a dual variable λ - for which we can analytically calculate the Jacobian matrix. By this approach we thus get direct access to the Jacobian of the "correct" system to solve and regularization enters via a single parameter used to approximate the Hamiltonian system. From a numerical perspective, the explicit calculation of the Jacobian matrix is a great advantage that significantly improves performance. Other studies typically either calculate prices numerically to obtain the gradient or Jacobian of the system ([23], [12], [20], [27]) or find a new equation to solve for the gradient ([26], [1], [4]). A drawback of our implementation of the optimal control technique from [32] that we exploit, we use a space Σ with the same dimension as the dimension used to discretize Dupire's equation (2). When the number of unknowns in the definition of the volatility σ becomes too large, the problem gets more difficult to solve. It can be mentioned that - although it is not implemented in this report - this technique could be used with a volatility that is piecewise constant in the strike dimension with fewer gridpoints than used for the discretization of Dupire's equation. We do address, though, the problem with the large number of unknowns due to the discretization in time in Section 3 by exploiting the results from Andreasen and Huge [2] to calculate prices at a fine grid in T from a discretely defined volatility σ obtained on a coarse grid. This technique allows us to apply our dynamic programming technique on a coarse grid in the time dimension T and still obtain a local volatility for the finer time-grid. Second, in Section 4, we use a "simpler" approach in which we define Σ to be the space of functions that are piecewise constant in the time-dimension T and piecewise linear in the space-dimension K - over some discrete grid in (T, K) - and apply an optimization algorithm that calculates the gradient and Jacobian of $\|C - \bar{C}\|^2$ with respect to σ by numerically solving (2). The dimension of Σ will be much smaller than the number of grid points used for the discretization of (2). As we will see, we also incorporate smoothness penalties on σ in the objective function. With this approach, regularization therefore occurs at two levels: by reducing the number of unknowns in limiting the number of grid points used to define Σ and through regularization penalties of σ . This technique is similar in spirit to what is done in many of the previous studies cited above.

Optimization of a local volatility function is tested on constructed data in [32] where the theoretical basis for the techniques we exploit is developed. These techniques have also been used

for the calibration of a local volatility model with jumps by Kiessling [24, paper IV]. The two cited examples do not, however, address the issue of sufficient accuracy in the discretization of the numerical schemes involved in the optimization. Indeed, as is noted in [23, p 9], “it is all too easy to generate a representation for instantaneous σ via a discretized tree or finite difference/element method that appears to price a set of vanilla options correctly, only to find that a much finer discretization yields a significantly different set of prices”. In this study the aim is to find a local volatility function σ such that the corresponding *analytical* solution to (2) (the analytical solution will of course need to be approximated numerically) yields prices that are within the bid-ask spread of the quoted market prices used as data. This amounts to using a sufficiently high level of discretization in the numerical schemes in the optimization, and is more challenging than to find a solution that fits the market given a discretization on a coarse grid. We test our algorithms on a larger data set than what we have seen in other studies, both in that we test our techniques on data from 62 consecutive trading days on two different indices and in that our data on the larger of these two indices contains more quotes than used in other studies, thus making the calibration more demanding (it is more difficult to fit a large number of prices than a smaller one).

The rest of this report is organized as follows. In Section 2 we describe the optimal control technique (the principles are given in Section 2.1 and implementation details in 2.2). In Section 3 we make use of the results from [2] to construct a local volatility on a fine grid in time based on a coarser representation. Section 4 deals with the optimization of our piecewise constant, piecewise linear local volatility. The numerical results are presented in Section 5 and concluding remarks are made in Section 6.

2 Local volatility as optimal control

2.1 Regularized Hamiltonian system

A general optimal control problem for a function constrained to follow Dupire’s PDE can be stated as

$$\min_{\sigma \in \Sigma} \int_0^T h(T, C) dT \quad (7)$$

subject to: C, σ satisfy (2),

where Σ is some space of functions on $[0, T] \times \mathbb{R}_+$. If h is chosen as a distance between C and some observed market prices as in (6), then a volatility function σ that satisfies (7) can be seen as an optimal control that “steers” the model option prices C as close as possible to observed market data under the constraint that C satisfies Dupire’s equation (2). (Note that with an appropriate choice of h , (7) can be made equal to (6)).

The formulation (7) has the drawback of containing a PDE in the constraint. In order to solve the problem numerically, we will need to discretize the PDE. A discretization in space will transform Dupire’s PDE into an ODE and problem (7) will be transformed into a corresponding problem with an ODE constraint. (For an example of optimal control of PDE, we refer to [31, paper II].) For this new problem we will use the well developed framework available for optimal control of ODE. To handle the ODE problem numerically, discretization in time will also be necessary.

Let us thus discretize the space dimension $K \in \mathbb{R}_+$ from (2) on a grid $\{K_0 < K_1 < \dots < K_{n+1}\}$ with $K_0 = 0$. We approximate the differential operators in (2) by finite difference operators according to

$$\begin{aligned} \partial_K C(T, K_j) &\approx D_j^1 C_j := \frac{1}{\mu_j^- + \mu_j^+} \left[\frac{\mu_j^-}{\mu_j^+} (C_{j+1} - C_j) + \frac{\mu_j^+}{\mu_j^-} (C_j - C_{j-1}) \right] \\ \partial_{KK} C(T, K_j) &\approx D_j^2 C_j := \frac{2}{\mu_j^- + \mu_j^+} \left[\frac{C_{j-1}}{\mu_j^-} - \left(\frac{1}{\mu_j^+} + \frac{1}{\mu_j^-} \right) C_j + \frac{C_{j+1}}{\mu_j^+} \right]. \end{aligned} \quad (8)$$

where C_j is shorthand for $C(T, K_j)$ and $\mu_j^- = K_j - K_{j-1}$, $\mu_j^+ = K_{j+1} - K_j$. We use these finite difference operators since we will use a non-uniform grid in order to refine the mesh at strikes

K close to S_0 (see Section 2.2). A Taylor expansion yields that the operator D_j^1 is $O(\mu_j^- \mu_j^+)$ and D_j^2 is $O(\max(|\mu_j^- - \mu_j^+|, \mu_j^- \mu_j^+))$. In other words, D_j^2 is formally second order accurate if $|\mu_j^- - \mu_j^+| \sim \mu_j^- \mu_j^+$.

We can now approximate the solution $C(T, K)$ of Dupire's equation (2) at the grid points K_j , $1 \leq j \leq n$ by a function $C_j(T) : \mathbb{R}_+ \rightarrow \mathbb{R}^n$, solution to the following ODE

$$\begin{aligned} C'_j(T) &= \frac{1}{2}\sigma_j^2(T)K_j^2D_j^2C_j(T) - q_T C_j(T) - (r_T - q_T)K_j D_j^1 C_j(T), \\ C_j(0) &= (S_0 - K_j)_+, \quad C_0(T) = S_0 e^{-\int_0^T q_t dt}, \quad C_{n+1}(T) = 0, \\ (T, j) &\in \mathbb{R}_+ \times \{1, \dots, n\}, \end{aligned} \tag{9}$$

where $\sigma_j(T)$ is equal to $\sigma(T, K_j)$ from (2). Now let

$$f = (f_1, \dots, f_n) : \mathbb{R}_+ \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^n$$

be defined by the spatial finite differences in the ODE (9), so that for $C, \sigma \in \mathbb{R}^n$ we have

$$f_j(T, C, \sigma) := \frac{1}{2}\sigma_j^2 K_j^2 D_j^2 C_j - q_T C_j - (r_T - q_T)K_j D_j^1 C_j, \tag{10}$$

where we use the boundary conditions for C_0, C_{n+1} defined in (9) for the difference operators at the boundaries $j = 1$ and $j = n$. We can then write (9) on vector form according to

$$\begin{aligned} C'(T) &= f(T, C(T), \sigma(T)), \quad T \in \mathbb{R}_+ \\ C(0) &= (S_0 - K)_+, \end{aligned} \tag{11}$$

where K is the vector $(K_1, \dots, K_n) \in \mathbb{R}^n$ and $(S_0 - K)_+$ is to be understood componentwise. Instead of the minimization problem (7), where the function C is constrained to follow Dupire's PDE, we now consider the corresponding problem where C satisfies the ODE (11),

$$\begin{aligned} &\min_{\sigma \in \Sigma} \int_0^{\bar{T}} h(T, C(T)) dT \\ \text{subject to: } &C'(T) = f(T, C(T), \sigma(T)), \quad 0 < T \leq \bar{T} \\ &C(0) = (S_0 - K)_+, \end{aligned} \tag{12}$$

and h is our penalty function defined on $\mathbb{R}^n \times \mathbb{R}_+$ and Σ is some space of functions $\sigma : \mathbb{R}_+ \rightarrow \mathbb{R}_+^n$. We will discuss our choices of h and Σ shortly.

If we define

$$\begin{aligned} u(t, c) &= \min_{\sigma \in \Sigma} \int_t^{\bar{T}} h(T, C) dT \\ \text{subject to: } &C'(T) = f(T, C, \sigma) \\ &C(t) = c, \end{aligned} \tag{13}$$

then it is well established that if h and f are Lipschitz continuous and bounded, the value function $u(t, c)$ is given as the unique viscosity solution [17] to the Hamilton-Jacobi-Bellman equation,

$$\begin{aligned} \partial_t u(t, c) + H(t, c, \nabla_c u(t, c)) &= 0, \quad (t, c) \in (0, \bar{T}) \times \mathbb{R}^n \\ u(t, c) &= 0, \quad (t, c) \in \{t = \bar{T}\} \times \mathbb{R}^n, \end{aligned} \tag{14}$$

with the Hamiltonian H defined by

$$H(T, C, \lambda) := \inf_{\sigma \in \Sigma} \{f(T, C, \sigma) \cdot \lambda + h(T, C)\}. \tag{15}$$

The Hamilton-Jacobi-Bellman equation can be used to establish a fundamental result in optimal control theory known as the Pontryagin maximum principle, see [5], that gives optimality conditions for problems of the type (12). If C, σ are optimal for the problem (12), then the Pontryagin maximum principle states the existence of a $\lambda(T)$ that satisfies

$$\begin{aligned} -\lambda'(T) &= \nabla_C f(T, C(T), \sigma(T, C(T), \lambda(T))) \cdot \lambda(T) + \nabla_C h(T, C(T)) \\ \lambda(\bar{T}) &= 0 \\ \sigma(T, C(T), \lambda(T)) &\in \operatorname{argmin}_{\alpha \in \Sigma} \{f(T, C(T), \alpha(T)) \cdot \lambda(T) + h(T, C(T))\}. \end{aligned} \quad (16)$$

It can be shown [32, eq (1.8), Remark 3.1] that if H, f, h are differentiable in $C, \lambda \in \mathbb{R}^n$, then for σ that satisfies (16), we have

$$\begin{aligned} \nabla_C H(T, C, \lambda) &= \nabla_C f(T, C, \sigma(T, C, \lambda)) \cdot \lambda + \nabla_C h(T, C) \\ \nabla_\lambda H(T, C, \lambda) &= f(T, C, \sigma(T, C, \lambda)) \\ H(T, C, \lambda) - \nabla_\lambda H(T, C, \lambda) \cdot \lambda &= h(T, C). \end{aligned} \quad (17)$$

The relations (17) together with the Pontryagin maximum principle (16) say that a necessary condition for optimality for the problem (12) is the existence of C, λ that satisfy

$$\begin{aligned} C'(T) &= \nabla_\lambda H(T, C(T), \lambda(T)), \quad 0 \leq T \leq \bar{T}, \quad C(0) = (S_0 - K)_+ \\ -\lambda'(T) &= \nabla_C H(T, C(T), \lambda(T)), \quad 0 \leq T \leq \bar{T}, \quad \lambda(\bar{T}) = 0 \end{aligned} \quad (18)$$

known as the Hamiltonian system corresponding to the problem (12) (or the characteristics of the Hamilton-Jacobi-Bellman equation (14), see e.g. [17]). We therefore see that if C, λ satisfy (18) and σ is chosen according to (16), then σ, C is a candidate solution to the problem (12).

We will follow the technique used in [32], [24, paper IV] and choose the space Σ in which we look for an optimal control as all functions bounded above by a given level $\bar{\sigma}$ and bounded below by some other level $\underline{\sigma}$. Unlike what is done in the cited papers, we will not choose constant bounds, but let the bounds be functions of both time and space. For the original control problem (7) this would correspond to letting the bounds $\underline{\sigma}, \bar{\sigma}$ be functions of T and K . For the discretized problem in (12), this means we let the bounds be functions $\underline{\sigma}, \bar{\sigma} : \mathbb{R}_+ \rightarrow \mathbb{R}_+^n$. Then Σ will be the space

$$\Sigma = \{\sigma : \mathbb{R}_+ \rightarrow \mathbb{R}_+^n : \underline{\sigma} \leq \sigma \leq \bar{\sigma}\}, \quad (19)$$

where the inequalities are to be understood componentwise.

As in [32], we now see that we can get an explicit expression for the Hamiltonian (15). The definition of Σ together with (10) gives

$$\begin{aligned} H(T, C, \lambda) &= \inf_{\sigma(T) \leq \sigma \leq \bar{\sigma}(T)} \{f(T, C, \sigma) \cdot \lambda + h(T, C)\} \\ &= \inf_{\sigma(T) \leq \sigma \leq \bar{\sigma}(T)} \left\{ \sum_{j=1}^n \left(\frac{1}{2} \sigma_j^2 K_j^2 D_j^2 C_j - q_T C_j - (r_T - q_T) K_j D_j^1 C_j \right) \lambda_j + h(T, C) \right\}. \end{aligned} \quad (20)$$

We observe that the infimum is achieved by minimizing the sum componentwise and that the minimum of each component is obtained by choosing either the lower or the upper bound as a function of the sign of the term $K_j^2 D_j^2 C_j \lambda_j$. If we introduce the function

$$s_{[a, b]}(x) := \begin{cases} b^2 x, & x \leq 0 \\ a^2 x, & x > 0, \end{cases} \quad (21)$$

for some $a, b \in \mathbb{R}$, then we can write the Hamiltonian H above as

$$\begin{aligned} H(T, C, \lambda) &= \sum_{j=1}^n \left(s_{[\sigma_j(T), \bar{\sigma}_j(T)]} \left(\frac{1}{2} K_j^2 (D_j^2 C_j) \lambda_j \right) - q_T C_j \lambda_j - (r_T - q_T) K_j D_j^1 C_j \lambda_j \right) \\ &\quad + h(T, C). \end{aligned} \quad (22)$$

Now H is clearly not differentiable, since it contains the function s which is only Lipschitz continuous. To solve this situation, the function s is replaced by a smooth approximation s_δ (cf [32]) for some positive constant δ according to

$$s_{\delta, [a, b]}(x) = x \frac{a^2 + b^2}{2} - \frac{b^2 - a^2}{2} \int_0^x \tanh\left(\frac{y}{\delta}\right) dy. \quad (23)$$

By replacing s in the Hamiltonian (20) by the smooth function s_δ we obtain a regularized version of H that we denote by H_δ ,

$$H_\delta(T, C, \lambda) := \sum_{j=1}^n \left(s_{\delta, [\sigma_j(T), \bar{\sigma}_j(T)]} \left(\frac{1}{2} K_j^2 (D_j^2 C_j) \lambda_j \right) - q_T C_j \lambda_j - (r_T - q_T) K_j D_j^1 C_j \lambda_j \right) + h(T, C). \quad (24)$$

This method of approximating the Hamiltonian by a smoother version has an interpretation in terms of a Tikhonov regularization [8]. With this smooth approximate Hamiltonian, we can attempt to solve the Hamiltonian system (18) with H replaced by H_δ :

$$\begin{aligned} C'_\delta(T) &= \nabla_{\lambda_\delta} H_\delta(T, C_\delta(T), \lambda_\delta(T)), \quad 0 \leq T \leq \bar{T}, \quad C_\delta(0) = (S_0 - K)_+ \\ -\lambda'_\delta(T) &= \nabla_{C_\delta} H_\delta(T, C_\delta(T), \lambda_\delta(T)), \quad 0 \leq T \leq \bar{T}, \quad \lambda_\delta(\bar{T}) = 0. \end{aligned} \quad (25)$$

A natural and important question is how the solution to our problem (12) changes when the Hamiltonian H is replaced by the smooth approximation H_δ . This question can be addressed through the theory of Hamilton-Jacobi-Bellman equations. We specify a trivial control problem whose corresponding Hamiltonian system is given by (25), and define the corresponding Hamilton-Jacobi-Bellman equation. We first notice that the smoothed Hamiltonian (24) implicitly defines the control as a function of C_δ, λ_δ . To see this, let us calculate the λ -gradient that appears in the Hamiltonian system (25). From (24) we have

$$\begin{aligned} C'_{\delta j}(T) &= \partial_{\lambda_{\delta j}} H_\delta(T, C_\delta, \lambda_\delta) = s'_{\delta, [\sigma_j(T), \bar{\sigma}_j(T)]} \left(\frac{1}{2} K_j^2 (D_j^2 C_{\delta j}) \lambda_{\delta j} \right) \frac{1}{2} K_j^2 (D_j^2 C_{\delta j}) \\ &\quad - q_T C_{\delta j} - (r_T - q_T) K_j (D_j^1 C_{\delta j}). \end{aligned} \quad (26)$$

A comparison of (26) and (9) shows that the first equation in the Hamiltonian system (25) corresponds to solving an ODE in C_δ of the type (10) but where σ has been replaced by a function implicitly defined by C_δ, λ_δ . Let us, for functions $C, \lambda : \mathbb{R}_+ \rightarrow \mathbb{R}^n$, define this function as $\sigma_\delta : \mathbb{R}_+ \rightarrow \mathbb{R}^n$ with j :th component

$$\sigma_{\delta j}(T, C(T), \lambda(T)) := s'_{\delta, [\sigma_j(T), \bar{\sigma}_j(T)]} \left(\frac{1}{2} K_j^2 (D_j^2 C_j(T)) \lambda_j(T) \right). \quad (27)$$

In view of (17), we define

$$h_\delta(T, C(T), \lambda(T)) := H_\delta(T, C(T), \lambda(T)) - \nabla_{\lambda} H_\delta(T, C(T), \lambda(T)) \cdot \lambda(T). \quad (28)$$

Let us rewrite the Hamiltonian system (25) for a general initial condition and with initial time t :

$$\begin{aligned} C'_\delta(T) &= \nabla_{\lambda_\delta} H_\delta(T, C_\delta(T), \lambda_\delta(T)), \quad t \leq T \leq \bar{T}, \quad C_\delta(t) = c \\ -\lambda'_\delta(T) &= \nabla_{C_\delta} H_\delta(T, C_\delta(T), \lambda_\delta(T)), \quad t \leq T \leq \bar{T}, \quad \lambda_\delta(\bar{T}) = 0. \end{aligned} \quad (29)$$

If we then let

$$\begin{aligned} u_\delta(t, c) &:= \min_{C_\delta, \lambda_\delta} \int_t^{\bar{T}} h_\delta(T, C_\delta(T), \sigma_\delta(T, C_\delta(T), \lambda_\delta(T))) dT \\ \text{subject to: } &C_\delta, \lambda_\delta \text{ satisfy (29)} \end{aligned} \quad (30)$$

we can consider $u_\delta(t, c)$ to be the unique viscosity solution of a degenerate control problem whose possible controls σ_δ are defined by (27) for solutions (C, λ) to (29). From the definition (15), it is direct to see that the Hamiltonian corresponding to (30) is given by H_δ , and so (29) is the Hamiltonian system of (30). The Hamilton-Jacobi-Bellman equation satisfied by u_δ reads

$$\begin{aligned} \partial_t u_\delta(t, c) + H_\delta(t, c, \nabla_c u_\delta(t, c)) &= 0, \quad (t, c) \in (0, \bar{T}) \times \mathbb{R}^n \\ u_\delta(t, c) &= 0, \quad (t, c) \in \{t = \bar{T}\} \times \mathbb{R}^n. \end{aligned} \tag{31}$$

Stability results for viscosity solutions of the Hamilton-Jacobi-Bellman equation [32] now gives us the error estimate

$$\|u - u_\delta\|_{\mathcal{C}([0, \bar{T}] \times \mathbb{R}^n)} \leq \bar{T} \|H - H_\delta\|_{\mathcal{C}([0, \bar{T}] \times \mathbb{R}^n)} \tag{32}$$

where $\mathcal{C}([0, \bar{T}] \times \mathbb{R}^n)$ is the space of continuous functions on $[0, \bar{T}] \times \mathbb{R}^n$ and $\|\cdot\|_{\mathcal{C}([0, \bar{T}] \times \mathbb{R}^n)}$ the corresponding L_∞ -norm. In replacing our original Hamiltonian system (15) by the approximate (25) we replace the solution to the original system (12) by the approximate solution (30), but the stability estimate (32) guarantees that corresponding value functions are pointwise close if the regularized Hamiltonian H_δ is pointwise close to the original H .

Let us now turn to the definition of the penalty function $h(T, C)$. We will penalize prices obtained from a given control σ if they deviate from market prices $\bar{C}(T)$ in Euclidian norm. We take $\bar{C}_j(\bar{T}^i)$ to be the mid-price for strike \bar{K}_j^i , i.e,

$$\bar{C}_j(\bar{T}^i) := \frac{1}{2} [\bar{C}^{\text{bid}}(\bar{T}^i, \bar{K}_j^i) + \bar{C}^{\text{ask}}(\bar{T}^i, \bar{K}_j^i)]. \tag{33}$$

However, since the strikes of the observable market data do not necessarily coincide with the grid we construct, the actual market quotes are here replaced by a smooth approximation as described in Appendix C. By replacing the real data by a smooth approximation, we also provide an extra regularization to our problem.

In continuous time, the h we use in our problem (12) is defined by

$$h(T, C(T)) = \sum_{j=1}^n \omega_j(T) (C_j(T) - \bar{C}_j(T))^2, \tag{34}$$

where the positive weight functions $\omega_j(T)$ attribute different importance to options of different strikes K_j and maturities T and will be discussed in Section 2.2.

We can now give componentwise expressions for the equations in our system (25) with the regularized Hamiltonian. The λ -gradient was given in (26). In the sequel, for ease of notation, we will write C, λ rather than C_δ, λ_δ for solutions to (25). The C -gradient of H_δ is

$$\begin{aligned} -\lambda'_j(T) &= \partial_{C_j} H_\delta(T, C, \lambda) \\ &= \sum_{l=-1}^1 \frac{1}{2} K_{j+l}^2 \lambda_{j+l} (\partial_{C_j} D^2 C_{j+l}) s'_{\delta, [\sigma_{j+l}, \bar{\sigma}_{j+l}]} \left(\frac{1}{2} K_{j+l}^2 (D^2_{j+l} C_{j+l}) \lambda_{j+l} \right) \\ &\quad - q_T \lambda_j - (r_T - q_T) \sum_{l=-1}^1 K_{j+l} \lambda_{j+l} (\partial_{C_j} D^1 C_{j+l}) \\ &\quad + 2 \omega_j (C_j - \bar{C}_j). \end{aligned} \tag{35}$$

We want to solve the system (25) (with components explicitly given in (26),(35)) of coupled ODE and we now turn to its time-discretization. We define a grid $\{0 = T^0 < T^1 < \dots < T^m = \bar{T}\}$ in time and replace the function $C(T) : \mathbb{R}_+ \rightarrow \mathbb{R}^n$ from (11) by a vector of variables $C_j^i \in \mathbb{R}$, $1 \leq i \leq n$, $1 \leq j \leq m$, where C_j^i is meant to approximate $C_j(T^i)$. We let λ_j^i and ω_j^i be the corresponding

discretization of $\lambda_j(T)$ and $\omega_j(T)$ respectively, and we let r^i, q^i be the values of the interest rate and dividend yield at T^i . With the function χ_θ given by

$$\chi_\theta(x, y) = (1 - \theta)x + \theta y, \quad (36)$$

for $\theta \in [0, 1]$, $x, y \in \mathbb{R}$, we define a θ -scheme in time. We let $\Delta T^i := T^i - T^{i-1}$ and discretize, for $1 \leq i \leq m$, $1 \leq j \leq n$, the components of $\nabla_\lambda H_\delta$ from (26) as

$$\begin{aligned} \frac{C_j^i - C_j^{i-1}}{\Delta T^i} &= -\chi_\theta(q^{i-1}, q^i) \chi_\theta(C_j^{i-1}, C_j^i) \\ &\quad - [\chi_\theta(r^{i-1}, r^i) - \chi_\theta(q^{i-1}, q^i)] K_j \chi_\theta(D_j^1 C_j^{i-1}, D_j^1 C_j^i) \\ &\quad + s'_{\delta, [\chi_\theta(\sigma_j^{i-1}, \sigma_j^i), \chi_\theta(\bar{\sigma}_j^{i-1}, \bar{\sigma}_j^i)]} \left(\frac{1}{2} K_j^2 \chi_\theta(D_j^2 C_j^{i-1}, D_j^2 C_j^i) \chi_{1-\theta}(\lambda_j^{i-1}, \lambda_j^i) \right) \\ &\quad \times \frac{1}{2} K_j^2 \chi_\theta(D_j^2 C_j^{i-1}, D_j^2 C_j^i), \end{aligned} \quad (37)$$

and $\nabla_C H_\delta$ in (35) as

$$\begin{aligned} -\frac{\lambda_j^i - \lambda_j^{i-1}}{\Delta T^i} &= \sum_{l=-1}^1 \left\{ \frac{1}{2} K_{j+l}^2 \chi_{1-\theta}(\lambda_{j+l}^{i-1}, \lambda_{j+l}^i) (\partial_{C_j} D^2 C_{j+l}) \right. \\ &\quad \times s'_{\delta, [\chi_\theta(\sigma_{j+l}^{i-1}, \sigma_{j+l}^i), \chi_\theta(\bar{\sigma}_{j+l}^{i-1}, \bar{\sigma}_{j+l}^i)]} \left(\frac{1}{2} K_{j+l}^2 \chi_\theta(D_{j+l}^2 C_{j+l}^{i-1}, D_{j+l}^2 C_{j+l}^i) \chi_{1-\theta}(\lambda_{j+l}^{i-1}, \lambda_{j+l}^i) \right) \Big\} \\ &\quad - \chi_\theta(q^{i-1}, q^i) \chi_{1-\theta}(\lambda_j^{i-1}, \lambda_j^i) \\ &\quad - [\chi_\theta(r^{i-1}, r^i) - \chi_\theta(q^{i-1}, q^i)] \sum_{l=-1}^1 K_{j+l} \chi_{1-\theta}(\lambda_{j+l}^{i-1}, \lambda_{j+l}^i) (\partial_{C_j} D^1 C_{j+l}) \\ &\quad + \chi_\theta(2\omega_j^{i-1} [C_j^{i-1} - \bar{C}_j^{i-1}], 2\omega_j^i [C_j^i - \bar{C}_j^i]) \end{aligned} \quad (38)$$

With a slight abuse of notation, we abbreviate (37), (38) into

$$\begin{aligned} C_j^i - C_j^{i-1} &= \Delta T^i \nabla_\lambda H_\delta (\chi_\theta(T^{i-1}, T^i), \chi_\theta(C_j^{i-1}, C_j^i), \chi_{1-\theta}(\lambda_j^{i-1}, \lambda_j^i)) \\ &\quad - (\lambda_j^i - \lambda_j^{i-1}) = \Delta T^i \nabla_C H_\delta (\chi_\theta(T^{i-1}, T^i), \chi_\theta(C_j^{i-1}, C_j^i), \chi_{1-\theta}(\lambda_j^{i-1}, \lambda_j^i)), \end{aligned} \quad (39)$$

bearing in mind that the expressions $\nabla_C H_\delta, \nabla_\lambda H_\delta$ in (39) are not only functions of $C_j^{i-1}, C_j^i, \lambda_j^{i-1}, \lambda_j^i$, and that the time-averaging denoted as $\chi_\theta(T^{i-1}, T^i)$ means that we evaluate the discretized version of r_T and other functions of time according to $\chi_\theta(r^{i-1}, r^i)$.

Note from (25) that we have an initial value problem in C and a terminal value problem in λ . For $\theta = 1$ the scheme is therefore fully implicit in both C and λ . To get a scheme that is second order in time we can set $\theta = 0.5$ and obtain a mid-point scheme. We will use both $\theta = 1$ and $\theta = 0.5$ for different purposes below.

Given a solution C, λ to the discrete Hamiltonian system (39), we can use (27) to calculate a control σ_j^i , $1 \leq i \leq m$, $1 \leq j \leq n$ according to

$$\sigma_j^i = s'_{\delta, [\chi_\theta(\sigma_j^{i-1}, \sigma_j^i), \chi_\theta(\bar{\sigma}_j^{i-1}, \bar{\sigma}_j^i)]} \left(\frac{1}{2} K_j^2 \chi_\theta(D_j^2 C_j^{i-1}, D_j^2 C_j^i) \chi_{1-\theta}(\lambda_j^{i-1}, \lambda_j^i) \right). \quad (40)$$

We see that for our mid-point scheme with $\theta = 0.5$, the volatility σ_j^i actually approximates the control $\sigma_j(T)$ of the ODE problem (12) at the time $T^{i-\frac{1}{2}} := \frac{1}{2}(T^{i-1} + T^i)$. We therefore have that σ_j^i in (40) is an approximation of $\sigma(T^{i-\frac{1}{2}}, K_j)$, where $\sigma(\cdot, \cdot)$ is a local volatility function assumed to satisfy the original control problem (7).

We want to solve the discrete Hamiltonian system (39) which is a non-linear equation in the unknown variables $C_j^i \in \mathbb{R}$, $1 \leq i \leq m$, $1 \leq j \leq n$ and $\lambda_j^i \in \mathbb{R}$, $0 \leq i \leq m-1$, $1 \leq j \leq n$. (Recall again from (25) that C is known at $T = 0$ and λ is known at $T = \bar{T}$.) But the dependence on C , λ is made explicit in (37), (38) and we can therefore analytically calculate the Jacobian of the system, which will facilitate the use of iterative optimization algorithms in its solving. To be precise, let us first define the variable $X \in \mathbb{R}^{2mn}$ according to

$$\begin{aligned} X_{2(i-1)n+j} &:= \lambda_j^{i-1}, & 1 \leq i \leq m, & 1 \leq j \leq n \\ X_{(2i-1)n+j} &:= C_j^i, & 1 \leq i \leq m, & 1 \leq j \leq n. \end{aligned} \quad (41)$$

We can then redefine (39) by means of a function $F(X) : \mathbb{R}^{2mn} \rightarrow \mathbb{R}^{2mn}$ given, for $1 \leq i \leq m$, $1 \leq j \leq n$, as

$$\begin{aligned} F_{2(i-1)n+j}(X) &:= -X_{2(i-1)n+j} + X_{2in+j} \\ &\quad + \Delta T^i \nabla_C H_\delta(\chi_\theta(T^{i-1}, T^i), \chi_\theta(X_{(2i-3)n+j}, X_{(2i-1)n+j}), \chi_{1-\theta}(X_{2(i-1)n+j}, X_{2in+j})) \\ F_{(2i-1)n+j}(X) &:= X_{(2i-3)n+j} - X_{(2i-1)n+j} \\ &\quad + \Delta T^i \nabla_\lambda H_\delta(\chi_\theta(T^{i-1}, T^i), \chi_\theta(X_{(2i-3)n+j}, X_{(2i-1)n+j}), \chi_{1-\theta}(X_{2(i-1)n+j}, X_{2in+j})), \end{aligned} \quad (42)$$

where we make the replacements $X_{(2i-3)n+j} = C_j^0$ for $i = 1$ and $X_{2in+j} = \lambda_j^m$ for $i = m$ to account for the initial value in C at T^0 and the terminal value in λ at T^m . The discretized Hamiltonian system (39) is now equivalent to

$$F(X) = 0, \quad (43)$$

and the Jacobian of F , useful for solving numerically for X , can be obtained through direct differentiation of (42).

2.2 Implementation

Previous studies ([8], [31, paper II], [24, paper IV]) that exploit the technique from [32] of regularizing the Hamiltonian system of an optimal control problem have exploited the Newton method to solve the resulting non-linear system. However, it is not in general easy to find a starting point such that the unaltered Newton method will converge for the problems (43). Rather than the direct Newton method, we therefore use the trust-region Newton method implemented in Matlab¹ in the lsqnonlin (non-linear least squares) function, which is substantially more robust in our case. The references given in Matlab's user manual for the trust-region algorithm are [11], [10].

The non-uniform grid we use in space for the ODE (9), including the boundary points K_0 and K_{n+1} , is defined according to

$$\begin{aligned} K_j &= S_0 \left[1 + \frac{1}{c} \tan \left(\left[1 - \frac{j}{n+1} \right] d_- + \frac{j}{n+1} d_+ \right) \right], & 0 \leq j \leq n+1 \\ d_- &= \arctan(c[y_{\min} - 1]), & d_+ &= \arctan(c[y_{\max} - 1]), \end{aligned} \quad (44)$$

for a positive c . This grid is most refined for $K_j \approx S_0$ and the step length at a given K_j is a function of the distance $|S_0 - K_j|$. We use $y_{\min} = 0$, $y_{\max} = 2.5$, which gives $K_0 = 0$, $K_{n+1} = 3S_0$. We choose $n = 440$ and $c = 1.02$ which approximately gives a mesh size $\Delta K := K_j - K_{j-1} \approx \frac{S_0}{250}$ for $K_j \approx S_0$ and $\Delta K \approx \frac{S_0}{125}$ for $K_j \approx 0$. Our grid in time, used for (39) is defined according to

$$T^i = \frac{\bar{T}}{c} \tan \left(\frac{i}{m} \arctan(c) \right), \quad 0 \leq i \leq m, \quad (45)$$

with \bar{T} from (12). As an example, for $\bar{T} = 4$, we use $c = 1.82$ in which gives $\Delta T := T_j - T_{j-1} \approx \frac{1}{275}$ for $t \approx \frac{10}{365}$ and $\Delta T \approx \frac{1}{150}$ for $T \approx 2$. We modify the grid obtained from (45) by inserting the

¹Matlab Version 7.12.0.635 (R2011a), Optimization Toolbox Version 6.0, The MathWorks, Inc., Natick, Massachusetts, United States.

maturities of observed market options, and by refining the grid further for small $T < \frac{10}{365}$. The extra refinement provides better accuracy for short dated options.

The full problem (43) for $2mn$ unknown becomes computationally heavy to solve for reasonable grid sizes. To reduce the problem size, we split our problem into smaller subproblems that are solved in sequence with a “bootstrapping” strategy (as is also done in e.g. [2], [27]) where we create a single problem for the first maturity and then solve for two maturities at the time. As in (4), we let \bar{T}^i , $1 \leq i \leq M$ be the maturities at which we have observable market prices and define $\bar{T}^0 = 0$. Let i_k , $1 \leq k \leq \bar{M}$ be a subsequence of $1, \dots, M$ with $i_{k-1} < i_k$ and $i_{\bar{M}} = M$. Now define a series of time-intervals $(\bar{T}^{i_{k-1}}, \bar{T}^{i_k}]$ for $k \geq 1$ and $i_0 = 0$. Remember that the system (43) results from the optimal control problem (12). Rather than searching directly for a solution to (12), we now define a space Σ_k for each k as

$$\Sigma_k = \{\sigma : (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}] \rightarrow \mathbb{R}_+^n : \underline{\sigma}_k \leq \sigma \leq \bar{\sigma}_k\}, \quad (46)$$

for functions $\underline{\sigma}_k, \bar{\sigma}_k : (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}] \rightarrow \mathbb{R}_+^n$ with $\underline{\sigma}_k \leq \bar{\sigma}_k$, and we look for functions $C_k, \sigma_k : (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}] \rightarrow \mathbb{R}^n$ that satisfy

$$\begin{aligned} & \min_{\sigma_k \in \Sigma_k} \int_{\bar{T}^{i_{k-1}}}^{\bar{T}^{i_k}} h(T, C_k(T)) dT \\ \text{subject to: } & C'_k(T) = f(T, C_k(T), \sigma_k(T)), \quad \bar{T}^{i_{k-1}} \leq T \leq \bar{T}^{i_k} \\ & C_k(\bar{T}^{i_{k-1}}) = C_{k-1}(\bar{T}^{i_{k-1}}). \end{aligned} \quad (47)$$

We let $C_0(0) := (S_0 - K)_+$, where K is a vector as in (11) to account for the initial condition at time 0. Each of the problems (47) is treated in the manner described above for the full problem, leading to a system of the type (43). Given solutions σ_k, C_k to the sequence of problems (47) for $k = 1, \dots, \bar{M}$, we can define

$$\sigma(T) = \sigma_k(T), \quad T \in (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}], \quad 1 \leq k \leq \bar{M} \quad (48)$$

and use $\sigma(T)$ as an approximation of the solution to the full problem (12).

The weights ω_j^i in (38) are defined by means of the bid-ask spread. We first put a maximum and minimum value on the observed spreads and define, for any observable pair of maturities and strikes (\bar{T}, \bar{K}) ,

$$s(\bar{T}, \bar{K}) := \max(aS_0, \min(\text{spread}(\bar{T}, \bar{K}), bS_0)) \quad (49)$$

for $a = 2 \cdot 10^{-4}$ and $b = 2 \cdot 10^{-3}$. For a date T^i with smallest observable strike \bar{K}_{\min}^i and largest strike \bar{K}_{\max}^i , the weights ω_j^i are given by

$$w_j^i := \begin{cases} \frac{1}{\sqrt{s(T^i, K_j)}}, & K_j \in [\bar{K}_{\min}^i, \bar{K}_{\max}^i] \\ 0, & K_j \notin [\bar{K}_{\min}^i, \bar{K}_{\max}^i], \end{cases} \quad (50)$$

so that the weight is zero outside of the observable range of strikes. For dates T^i in the grid that do not correspond to observable maturities, we put the weights to zero, since no data is available.

The resulting systems of the type (39) are easier to solve for large δ (meaning a large regularization) and for a small distance between $\underline{\sigma}$ and $\bar{\sigma}$ (meaning that we search for our control σ in a “small” space Σ). Indeed, for large δ , the system (18) tends to a linear ODE with constant volatility value $0.5(\bar{\sigma} + \underline{\sigma})$, and if $\bar{\sigma} - \underline{\sigma} \equiv 0$, then (18) is also a linear ODE with volatility $\bar{\sigma} \equiv \underline{\sigma}$. This insight leads us to implement an algorithm where we solve for decreasing values in δ (as in ([8], [31, paper II], [24, paper IV])), but also for increasing values in $\bar{\sigma} - \underline{\sigma}$. We calculate an initial guess σ_{init} , for the local volatility from affine model parameters as described in Appendix B and define $\bar{\sigma} = \sigma_{\text{init}} + \Delta_\sigma$, $\underline{\sigma} = \max(\sigma_{\text{init}} - \Delta_\sigma, 0.05)$. The system (39) is then solved for a sequence of decreasing values of δ . We calculate the control σ resulting from the solution of the problem, and check if our solution C is close enough to the data. If the solution is not satisfactory, we update

the control space by increasing the value Δ_σ and redefine Σ by setting the bounds as $\bar{\sigma} = \sigma + \Delta_\sigma$, $\underline{\sigma} = \max(\sigma - \Delta_\sigma, 0.05)$. Then we proceed to another round of solving with the same sequence of decreasing δ as before. When one system (47) with index k is solved we move to problem $k+1$, using the solution k at its last time-step as initial condition for problem $k+1$.

We give a more succinct description of the procedure in Algorithm 1. The stopping criteria we use is that at a maturity T^i where original market data was available, all options within the range of observable strikes should be below a certain fraction of the modified spread $s(T^i, K)$ from (49). The notation in Algorithm 1 is as follows. The indices i_k are as in (46) and (47). The prices \hat{C} is the approximation of market data at each maturity as in Appendix C. \mathcal{T} is the set of market maturities as in (4). The maximal number of repetitions rep_{\max} for increasing the boundaries $\underline{\sigma}$ and $\bar{\sigma}$ is set to 12. The regularizing constant δ is, for each solving step, decreased from 5 to 0.15.

Algorithm 1 Optimal control algorithm for local volatility.

Solve (72) and calculate the resulting prices C_{init} and local volatility σ_{init} .
Set $C = C_{\text{init}}$, $\lambda = 0$ for $T^i \in [0, \bar{T}^M]$.
Initiate X according to (41).
Calculate the approximation of market data \hat{C} in (77).
for $k = 1 \rightarrow \bar{M}$ **do**
 Set $\underline{\sigma} = \bar{\sigma} = \sigma_{\text{init}}$ for $T^i \in (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}]$.
 Set $\Delta_\sigma = 0$.
 Set $\text{rep} = 0$.
 repeat
 Update $\text{rep} = \text{rep} + 1$.
 Set $\delta = 5$.
 for $i = 1 \rightarrow 10$ **do**
 Solve system $F(X) = 0$ from (43) corresponding to the problem (47) with index k .
 Update X with the solution to $F(X) = 0$.
 Update $\delta = \delta \sqrt[10]{0.03}$.
 end for
 Calculate a control σ_k on $(\bar{T}^{i_{k-1}}, \bar{T}^{i_k}]$ according to (40).
 Update $\Delta_\sigma = \min(\Delta_\sigma + 0.02, 0.1)$.
 Update $\underline{\sigma} = \max(\sigma_k - \Delta_\sigma, 0.05)$.
 Update $\bar{\sigma} = \sigma_k + \Delta_\sigma$.
 until $|C_j^i - \hat{C}^i(K_j)| \leq 0.35 s(T^i, K_j)$ for all (i, j) such that $T^i \in (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}] \cap \mathcal{T}$, $K_j \in [\bar{K}_1^i, \bar{K}_{N_i}^i]$ **or** $\text{rep} = \text{rep}_{\max}$
 Set $\sigma = \sigma_k$ for $T^i \in (\bar{T}^{i_{k-1}}, \bar{T}^{i_k}]$.
end for

Splitting the optimal control problem (12) into parts as in (47) reduces the problem into a sequence of smaller problems that are faster to solve. Nonetheless, in order to make the numerical discretization error of the system (37), (38) small in relation to the spread, we need a rather fine grid. As we will see below in Section 5, the fairly large number of variables affects not only the calculation time, but also the quality of the result: it is more difficult to obtain a good result when the problem is greatly underdetermined. It is therefore desirable to reduce the number of variables without losing the numerical accuracy. This is the topic of Section 3.

3 Creating a fine local volatility from a coarse grid

In this section we exploit a technique developed by Andreassen and Huge [2] to consistently interpolate option prices from a coarse grid to a finer mesh in the time dimension by means of a specific use of a fully implicit finite difference scheme. This makes it possible for us to solve our problems (47) with a fully implicit scheme (i.e., $\theta = 1$) in (37), (38), interpolate prices onto an arbitrarily fine mesh in time and obtain the local volatility on this finer mesh via numerical differentiation.

Suppose we have a time grid $0 = T^0 < \dots < T^m$, a space-grid $0 = K^0 < \dots < K^n$ and a matrix $\sigma \in \mathbb{R}^{mn}$ of positive volatility values. Let C_j^i for $1 \leq i \leq m$, $1 \leq j \leq n$ be the solution of a fully implicit time discretization of (9), so that

$$\begin{aligned} C_j^i &= \left[1 - (T^{i+1} - T^i) \left(\frac{1}{2} (\sigma_j^{i+1})^2 K_j^2 D_j^2 + q^{i+1} + (r^{i+1} - q^{i+1}) K_j D_j^1 \right) \right] C_j^{i+1} \\ C_j^0 &= (S_0 - K_j)_+, \quad C_0^i = S_0 e^{-\int_0^{T^i} q_t dt}, \quad C_{n+1}^i = 0, \end{aligned} \quad (51)$$

where $r^i = r_{T^i}$ represents the interest rate at T^i and $q^i = q_{T^i}$ is the dividend yield. The idea in [2] is to make use of the scheme (51) to calculate prices defined on the same space grid K_j but for maturities $T \notin \{T^1, \dots, T^m\}$. Let C_j^i solve (51). Andreasen and Huge show (for the case $r^i \equiv q^i \equiv 0$) that if we for a given $T \in (0, T^n]$ identify the index i such that $T \in (T^i, T^{i+1}]$ and let $C_j(T)$, $j = 1, \dots, n$ be given as the solution to

$$\begin{aligned} C_j^i &= \left[1 - (T - T^i) \left(\frac{1}{2} (\sigma_j^{i+1})^2 K_j^2 D_j^2 + q^{i+1} + (r^{i+1} - q^{i+1}) K_j D_j^1 \right) \right] C_j(T) \\ C_0(T) &= S_0 e^{-\int_0^T q_t dt}, \quad C_{n+1}(T) = 0, \end{aligned} \quad (52)$$

then the values $C_j(T)$, seen as prices of call options with maturity T and strike K_j , are arbitrage free. Note that for any $T \in [T^i, T^{i+1}]$, the value $C_j(T)$ is obtained from a single fully implicit time-step from T^i . Also note that $C_j(T^{i+1}) = C_j^{i+1}$ for all i , so that $C_j(T)$ provides an arbitrage free interpolation of the prices C_j^i . Andreasen and Huge use the scheme (52) to interpolate option prices in time after having optimized volatility values σ to fit market prices on a coarse grid with only one grid point in time per observed market maturity.

Our objective is to make use of this interpolation technique to construct a local volatility $\bar{\sigma}$ on a fine mesh that will give a solution to Dupire's equation (2) that coincides with the values obtained from solving the sequence of problems (47) for a coarse time-grid and a fully implicit finite difference scheme. By using fewer time-steps, we will speed up computations. Note that with $\theta = 1$ in (36) our system (39) becomes equal to (51) in the C -variable. We now let $\bar{T}^1 < \dots < \bar{T}^m$ be a refinement of the grid T^i . We then calculate a new volatility $\bar{\sigma}_j^{i+\frac{1}{2}}$ at the points $(\bar{T}^{i+\frac{1}{2}}, K_j)$ with $\bar{T}^{i+\frac{1}{2}} = \frac{1}{2}(\bar{T}^i + \bar{T}^{i+1})$ with a mid-point differentiation according to

$$(\bar{\sigma}_j^{i+\frac{1}{2}})^2 = \frac{C_j(\bar{T}^{i+1}) - C_j(\bar{T}^i)}{(\bar{T}^{i+1} - \bar{T}^i) \left(\frac{1}{2} K_j^2 D_j^2 \bar{C}_j^{i+\frac{1}{2}} - q^{i+\frac{1}{2}} \bar{C}_j^{i+\frac{1}{2}} - (r^{i+\frac{1}{2}} - q^{i+\frac{1}{2}}) K_j D_j^1 C_j^{i+\frac{1}{2}} \right)}. \quad (53)$$

where we use the notation $\bar{C}_j^{i+\frac{1}{2}} = 0.5(\bar{C}_j^i + \bar{C}_j^{i+1})$ and likewise for $\bar{r}^{i+\frac{1}{2}}$, $\bar{q}^{i+\frac{1}{2}}$. The values $C_j(\bar{T}^i)$ are calculated from (52), \bar{r}^i is the interest rate and \bar{q}^i the dividend yield at \bar{T}^i .

If the prices $C_j(\bar{T}^i)$ are arbitrage free, then the right-hand side of (53) will be positive and $\bar{\sigma}$ well defined. However, this calculation of quotes of finite differences is not robust and can lead to undesirable results in two ways. Firstly, even if the prices $C_j(\bar{T}^i)$ are arbitrage free, the resulting $\bar{\sigma}$ can be very unsmooth, especially for small K_j where the prices are almost linear. Secondly, even if the prices are formally arbitrage free, round-off errors in the numerically calculated prices can (and do) produce sharp spikes in $\bar{\sigma}$ in regions where both the numerator and denominator in (53) are close to zero by machine precision. Indeed, even when we let $\bar{T}^i = T^i$ for $i = 1, \dots, m$, so that $C_j(\bar{T}^i) = C_j^i$, round-off errors in solving (51) will lead to sharp spikes in $\bar{\sigma}$ re-calculated from (53), even though now the equations (51) and (53) are algebraically equivalent.

To get a more robust result from (53), we want to add convexity to $C_j(\bar{T}^i)$. We do this by adding a small, convex function to the initial condition in (51). So instead of $C_j^0 = (S_0 - K_j)_+$, we set

$$C_j^0 = (S_0 - K_j)_+ + \gamma(K_j) \quad (54)$$

for some small, positive convex function γ . Specifically, we use

$$\gamma(K_j) = \epsilon S_0 K_1 \left(\frac{1}{K_j} - \frac{1}{K_n} \right), \quad 1 \leq j \leq n \quad (55)$$

where we set $\epsilon = 10^{-5}$. As we illustrate in Section 5, this simple change in the initial condition robustifies the differentiation (53) substantially while γ can be kept small enough not to produce any significant change in prices calculated from (51) and (52).

When we carry out the differentiation (53) from prices obtained with (52) without “convexification”, sharp discontinuities mostly occur for low strikes where the value of the volatility does not affect prices very much. In this respect, any smoothing of the volatility in this region can be considered to be mostly a cosmetic change. However, in numerical applications it is often desirable to have a function that is as smooth as possible. Also, the differentiation (53) can produce negative values of σ^2 due to round-off errors even for formally arbitrage free prices C , so regularization of some sort is desirable.

4 Optimization of a piecewise constant, piecewise linear local volatility

We let $T^0 = 0$ and $T^i, i = 1, \dots, m$ be the maturities for which we have observable market data and define a grid $(T^0, \dots, T^m) \times (K_0, \dots, K_n)$ in maturity and strike. We then define Σ in (6) to be the space of positive functions that are piecewise constant on (T^1, \dots, T^m) , piecewise linear and continuous on (K_1, \dots, K_n) and bounded below and above according to

$$\begin{aligned} \Sigma = \left\{ \sigma : (T, K) \rightarrow \mathbb{R}_+ : \sigma(T, K) = \frac{K_j - K}{K_j - K_{j-1}} \sigma(T^i, K_{j-1}) + \frac{K - K_{j-1}}{K_j - K_{j-1}} \sigma(T^i, K_j), \right. \\ \underline{\sigma}_j^i \leq \sigma(T^i, K_j) \leq \bar{\sigma}_j^i, \\ (T, K) \in [T^{i-1}, T^i] \times [K_{j-1}, K_j], 1 \leq i \leq m, 1 \leq j \leq n, \\ \left. \sigma(T, K) = \sigma(T^0, K), T < T^0, \sigma(T, K) = \sigma(T^m, K), T > T^m \right\}, \end{aligned} \quad (56)$$

for some positive constants $\underline{\sigma}_j^i, \bar{\sigma}_j^i, 1 \leq i \leq m, 1 \leq j \leq n$. We let the grid points T^i for the volatility be exactly the maturities \mathcal{T} from (4) at which we have observable data. As for the points K_j , we create the union of the strikes at which we have quoted prices at some maturity, $\mathcal{K} = \bigcup_{i=1}^m \mathcal{K}^i$. We then define one grid point K_j for each strike $K \in \mathcal{K}$ and add a few grid points in the intervals $(0, K_1]$ and $(K_n, 2.5S_0]$ where K_1 is the smallest strike in \mathcal{K} and K_n the largest. Since we use the union of all the quoted strikes to define the grid points that define Σ , the number of points in our grid will be larger than the number of quoted contracts in the market, but the resulting problem will have considerably fewer degrees of freedom than the optimal control problem we formulated in Section 2.

Just as for the optimal control problem, we approach (6) in a “bootstrapping” manner (as is also done in [2] and [27]). For this purpose, it is of course convenient that our space Σ consists of functions with local support, i.e., using a space of functions with global support would not be suitable for “splitting the problem into pieces” in the time variable T . We will optimize for a single maturity at the time while keeping σ unchanged for times $T < T^i$ when solving for maturity T^i .

Define C^i to be the solution to

$$\begin{aligned} \partial_T C^i(T, K) &= \frac{1}{2} \sigma^2(T, K) K^2 \partial_{KK} C^i(T, K) - q_T C^i(T, K) - (r_T - q_T) K \partial_K C^i(T, K) \\ (T, K) &\in (T^{i-1}, T^i] \times \mathbb{R}_+, \\ C^i(T^{i-1}, K) &= C^{i-1}(T^{i-1}, K), C^i(T, 0) = S_0 e^{-\int_0^T q_t dt}, C^i(T, \infty) = 0 \end{aligned} \quad (57)$$

with the convention that $C^0(0, K) = (S_0 - K)_+$. For $1 \leq i \leq m$ we can then consider the sequence

of problems

$$\begin{aligned} \min_{\underline{\sigma}_j^i \leq \sigma(T^i, K_j) \leq \bar{\sigma}_j^i, 1 \leq j \leq n} & \sum_{j=1}^n \omega_j^i (C^i(T^i, K_j) - \bar{C}(T^i, K_j))^2 + \\ & \epsilon_K S_0^2 \sum_{j=1}^n \left(100 D_j^2 \sigma(T^i, K_j) \mathbf{1}_{\{D_j^2 \sigma(T^i, K_j) < 0\}} + \right. \\ & \quad \left. D_j^2 \sigma(T^i, K_j) \mathbf{1}_{\{D_j^2 \sigma(T^i, K_j) \geq 0\}} \right)^2 + \\ & \epsilon_T S_0^2 \sum_{j=1}^n \left(\frac{\sigma(T^i, K_j) - \sigma(T^{i-1}, K_j)}{T^i - T^{i-1}} \right)^2 \end{aligned} \quad (58)$$

subject to: C^i, σ satisfy (57),

where S_0 is the spot price, ϵ_K, ϵ_T are positive constants and ω_j^i are as in (50). By $\underline{\sigma}_j^i, \bar{\sigma}_j^i$ we define lower and upper limits for the volatility $\sigma \in \Sigma$ we search on the interval $(T^{i-1}, T^i]$. The second sum in the objective function contains the finite difference operator D_j^2 from (8). By penalizing for large values of the second derivative we smooth the resulting σ in the strike range. We give a much larger penalty for negative values of D_j^2 which will tend to give a convex surface in strike. Likewise, the third sum in the objective function penalizes values of σ that fluctuates in T . The use of a regularization of this type in the objective function is well established for inverse problems in general (see e.g. [35]) and for the calibration of local volatility functions in particular ([1], [26], [23], [12], [20]).

The problem (58) is solved with Matlab¹ using the quasi-Newton method implemented in the “active-set” algorithm of the fmincon function. Matlab’s user manual gives the following references: [29], [30], [7], [19], [21], [21].

As in our optimal control algorithm 1, we solve for a successively larger space Σ in the following manner. Given an initial guess σ_{init} for the values of σ at the nodes (T^i, K_j) , we first solve (58) with $\underline{\sigma}_j^i = \max(\sigma_{\text{init}}(T^i, K_j) - 0.25, 0.05)$ and $\bar{\sigma}_j^i = \sigma_{\text{init}}(T^i, K_j) + 0.25$.

If the solution σ yields prices that are satisfactory (i.e., that are closer than some fraction of the spread to the mid-price) we stop, otherwise we redefine Σ by taking larger limits around the present σ . The initial guess σ_{init} we use is, as before, obtained from the affine model as described in Appendix B and as before we calibrate to the approximation (77) rather than to the raw market data. This gives Algorithm 2.

Algorithm 2 Algorithm for piecewise constant, piecewise linear local volatility.

Calculate the approximation \hat{C} in (77).

for $i = 1 \rightarrow m$ **do**

- Set $\sigma(T^i, K_j) = \sigma_{\text{init}}(T^i, K_j)$ for $1 \leq j \leq n$.
- Set $\Delta_\sigma = 0.25$.
- Set rep = 0.
- repeat**

 - Set $\underline{\sigma}_j^i = \max(\sigma(T^i, K_j) - \Delta_\sigma, 0.05)$ for $1 \leq j \leq n$.
 - Set $\bar{\sigma}_j^i = \sigma(T^i, K_j) + \Delta_\sigma$ for $1 \leq j \leq n$.
 - Solve (58) and obtain $C^i(T^i, K_j), \sigma(T^i, K_j)$ for $1 \leq j \leq n$.
 - Update $\Delta_\sigma = 2 \Delta_\sigma$.
 - Update count = count + 1.

- until** $|C^i(T^i, K_j) - \hat{C}^i(K_j)| \leq 0.35 s(T^i, K_j)$ for $1 \leq j \leq n$ **or** rep = 3
- end for**

The “bootstrapping” approach in (57) and Algorithm 2 obviously means that only a PDE on the domain $[T^{i-1}, T^i]$ needs to be discretized and solved at each step which reduces the calculational burden.

5 Numerical results

We use market data on European put- and call options from Bloomberg written on the Euro Stoxx 50 Index (SX5E, the largest European equity index with the most liquid option market), and on OMX the Stockholm 30 Index (OMX, the major Swedish equity index), and a zero-coupon curve in the currency of each index provided by Svenska Handelsbanken AB. Our algorithms deal with call options, and we have translated prices on put options into call option prices using the put-call parity relation. The forward price we use is obtained from put-call parity using the zero-coupon curve and the quoted strike closest to the spot price S_0 at each maturity. We only use prices on out-of-the-money (OTM) options, i.e., we use put options for strikes $K < S_0$ and call options for $K \geq S_0$, both because OTM options are usually more liquid and because OTM put options have much tighter spreads than their corresponding call options of the same strike. Only prices from maturity-strike pairs for which both a put- and a call option are quoted are considered. We filter our data slightly by removing some arbitrage opportunities as described in Appendix A. This is done because we do not wish to include prices we know beforehand to be impossible to replicate when we test the performance of our algorithms. The interest rate r_t and dividend yield q_t in (2) are obtained from the zero-coupon curve and the implicit forward prices as described in Appendix A.

We have run our algorithms on daily data registered at around 17.07 CET during 62 trading days in the first quarter of 2013, from the 3rd of January to the 28th of March 2013². The sets of options on SX5E and OMX are displayed in Figure 1(f) and 2(f) respectively. On SX5E we have 19374 quotes in total with a longest quoted maturity on each day usually between 2 and 4 years. On average we have 312 quoted prices each day, 9 distinct maturities and 34 strikes per maturity. For the smaller OMX index, we have a total of 4879 quotes with the longest quoted daily maturity usually just below 1 year, a daily average of 79 quotes, 7 maturities and 11 quoted strikes per maturity.

Our objective is to create a local volatility function that (given our forward prices and zero-coupon curve) reproduces the observed market quotes. By “reproduce” we mean that we want to find a continuously defined function $\sigma(T, K) : \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ such that the analytical solution to Dupire’s equation (2) is within the market’s bid- and ask prices for the observable maturities and strikes. For the piecewise constant, piecewise linear local volatility in Section 4, the local volatility σ is indeed defined for every (T, K) according to (56). As for the optimal control technique from Section 2, it will produce volatilities σ_j^i , $i = 1, \dots, m$, $j = 1, \dots, n$ that are only pointwise defined on a grid $[T_1, \dots, T_m] \times [K_1, \dots, K_n]$ of maturities and strikes. When testing our results, we will therefore consider a continuous function $\sigma(T, K)$ given as a bilinear interpolation of the discrete values σ_j^i . To obtain the solution of Dupire’s equation with high accuracy for a given, continuously defined σ , we will use a Crank-Nicholson finite difference scheme with a finer discretization in time T and strike K than used in our optimization algorithms.

Since we calibrate to mid-prices \bar{C} from (33), we want all obtained prices to be within $\bar{C} \pm \frac{1}{2}$ spread. We will illustrate which of our quoted prices that are not replicated within these bounds. We will also - for each calibration date - identify the “worst” result as the calibrated price that deviates most from the bid- or ask price. For each set of calibrated options we thus display the largest deviation in basis points between any of the calibrated prices $C(T, K)$ and the market’s bid- and ask prices,

$$d_{\text{bid}}^{\text{ask}} = \max_{(T, K) \in \mathcal{I}} \{ \bar{C}^{\text{bid}}(T, K) - C(T, K), C(T, K) - \bar{C}^{\text{ask}}(T, K) \} \cdot \frac{10^4}{S_0}, \quad (59)$$

with \mathcal{I} is as in (4). A negative value of $d_{\text{bid}}^{\text{ask}}$ means that all quotes were replicated within the spread.

We will display results from seven different calibrations, five of which are from variations of our optimal control algorithm plus two results for the piecewise constant, piecewise linear volatility optimization. The cases we will compare are:

²Data for the 29th of March was lacking from our sets.

Underlying	Method			
	a	b	d	e
SX5E	60	9	19	14
OMXS30	15	4	5	2.5

Table 1: Typical execution times in minutes for the different calibration algorithms a, bd and e.

- a. The optimal control algorithm for a mid-point scheme (i.e., setting $\theta = 0.5$ in (39)) and with the rather fine grid in T obtained from (45) and the grid in K from (44) as described in Section 2.2.
- b. The optimal control algorithm with a fully implicit scheme (i.e., setting $\theta = 1$ in (39)), a uniform grid with only 25 points per year in T (plus a refinement for the shortest maturities $T < \frac{10}{365}$) and a subsequent reconstruction of the obtained volatility on a much finer grid as described in Section 3.
- c. The same methodology as in b, but here the prices used as data at each maturity come from an approximation of the market data constructed by setting $\theta = 1$ in (77), i.e., we only use the affine model and no splines (see Appendix C). This gives us a smoother data but with lower accuracy.
- d. Optimization of a piecewise defined volatility as in Section 4 *without* any smoothing penalty (i.e., setting $\epsilon_K = \epsilon_T = 0$ in (58)).
- e. Optimization of the piecewise defined volatility with smoothing constants $\epsilon_K = 10$ and $\epsilon_T = 5 \times 10^{-8}$ in (58).
- f. The same methodology as in b, but with the regularizing coefficient δ starting at the value 10 in Algorithm 1 to be decreased successively to 5, meaning that we have a much larger regularizing constant than in b.
- g. The same methodology as in b, but the constant rep_{\max} in Algorithm 1 is set to 3, meaning that we restrict ourselves to a smaller span for the boundaries σ and $\bar{\sigma}$.

Cases f and g will only be displayed for a single date to depict how the stopping criterion (case f) and regularizing coefficient (case g) affects the smoothness of the resulting volatility.

We first compare typical execution times for the different calibration cases. Our code is written in C++ and Matlab and executed on an Intel i7 2.93 GHz CPU. Table 1 displays typical run times for case a, b, d and e for our two equity indices. The measured time includes construction of the initial guess and the calibration of the affine model parameters used for this purpose. The algorithm with the refined grid with optimal control is obviously more costly in terms of computation time than the case with a coarser grid in time and the algorithms with a piecewise defined volatility from Section 4. Also, the use of smoothing of the piecewise defined volatility is more costly than the problem without regularization. The faster algorithms have execution times that allow for usage in the production at a financial institute. Something that we have not explored is the possibility to use a previously obtained solution as initial guess when recalibrating the model during the day. A volatility surface that gives a good fit at noon should usually not need much modification to fit the market at two o'clock, so starting from the last produced surface might be much faster than running an algorithm "from scratch".

In Figure 1(f) we give a graphical display of the maturities and moneyness (strike divided by spot price) of all the options in our data set on SX5E and in Figure 1(a) to 1(e) we show graphs depicting the maturities and moneyness of those quotes on SX5E that were *not* replicated within the strike in each of the cases a to e above. Figure 2 shows the same graphs for the quotes on OMX. It is apparent that the difficult options to replicate are those of short maturity and of low strike. A majority of the options not replicated within the spread are of maturity less than 30

days. In all cases, we get a good match of the quoted prices, with roughly 98 – 99.5% of the quoted prices replicated within the spread.

The methodology that matches most options is b, the optimal control algorithm with a coarse grid in time and subsequent construction of a volatility on a fine grid, which matches about 99.2% of the quotes on SX5E and 99.4% of the OMX quotes. When the input data is replaced by smoother data as in c, fewer options are matched, which of course is an indication that the smoothing of the input data is not accurate enough to reproduce all prices. (We include this case to illustrate how the smoothing of the data affects the resulting volatility surface, which we will show below). The “direct” optimal control algorithm (case a) with a mid-point scheme and fine discretization in T yields relatively poor results compared to the other cases. With the piecewise defined volatility without smoothing (case c), we fit more options (around 99% of all quotes) than with the optimal control algorithm with fine T -grid (case a) but fewer than with the optimal control using a coarse grid (case b). The smoothing of the piecewise volatility optimisation (case e) yields a somewhat poorer fit to data, as expected.

The poorer result for calibration a (the optimal control algorithm with fine T -grid) indicates that the largely underdetermined character of this problem setup is problematic. For SX5E with this method, the average size of the total number of grid points used for each day is around 2.6×10^5 whereas the average number of quoted contracts used as input is 312, indicating that we have roughly 800 unknown variables per input data point. The corresponding numbers for the OMX data are an average of 1.1×10^5 grid points daily, 78 contracts daily and around 1400 unknowns per input. For calibration cases b and c, where we use optimal control with a coarser grid in time, the ratio of unknowns to input data is reduced to around 150 for SX5E and 320 for OMXS30. In contrast, for the calibration of the piecewise volatility in case d and e, we have just around 2.8 unknowns for each data point on SX5E and approximately 5.5 on OMXS30. The calibration of our piecewise volatility is thus still an underdetermined problem, but less severely so than our optimal control problems.

Figure 3 compare the daily results of the calibration of SX5E options with optimal control with a coarse grid (case b) and that with a piecewise volatility optimized with smoothing constraints (case e). For the two cases, we display the daily number of options not produced within the spread (Figure 3(a) and 3(c)) and error $d_{\text{bid}}^{\text{ask}}$ from (59) (Figure 3(b) and 3(d)). We see that although the number of options outside the spread from day to day is significantly larger for the smoothed, piecewise volatility, the maximum error $d_{\text{bid}}^{\text{ask}}$ is within the same range, and sometimes smaller. This is another way of saying that even less options are reproduced within the spread with the smoothed, piecewise volatility, the options that are outside the spread are usually still fairly accurately reproduced. We can also observe in Figure 3 that the largest errors occur on days where the shortest quoted maturities are of only one or two days (see the dates marked on the top vertical axis).

In Figure 4 we show the actual pricing errors obtained from case b and e for two different option maturities quoted on SX5E on the 30th of January 2013. For clarity, we have deduced the market mid-prices from the displayed prices, so that the zero-level in the graphs corresponds to a perfect replication. As can be seen, the spread is often rather small (the smallest in the example is ± 0.05 Euros) compared to the spot price (2732.75 Euros in this case). The rather high precision demanded is one of the numerical challenges in the calibration of a local volatility that matches quoted prices. The graphs show a typical situation in the sense that the options that are not perfectly replicated have the shorter maturity (16 days) whereas the fit is better for the longer maturity (233 days), and in that those options that are outside the spread are not “far” outside.

We now look at graphs of a few of our calibrated local volatility functions, first on SX5E for market data as of January 30 2013. In Figure 5(a) we display the local volatility used as initial guess, corresponding to a calibrated affine model as described in Appendix B. Figure 5(b) shows the local volatility resulting from the optimal control algorithm with a fine T -grid (case a) using this initial guess. We can see that the surface has some unsMOOTH sections.

Two surfaces obtained from the optimal control algorithm using a coarser grid in time (case b and c) are shown in Figure 6. The surface obtained with the smoother input data (case c, Figure 6(b)) is obviously smoother than the surface calibrated to the less smooth data (Figure 6(b)),

corresponding to case b). For this particular case, when we interpolate the respective surfaces as in Section 3 and reprice the quoted options, the smoother surface reproduces 345 and the less smooth surface 346 of the 348 quoted prices within the spread. This illustrates that smoothing of the input data can have a significant effect on the resulting volatility surface and still give a similar fit to market prices. However, as we saw in Figure 1 and 2, the smoother version of the data is not in general accurate enough to give results equivalent to the case where less smooth data is used.

In Figure 7 we show the local volatility reconstructed on a fine grid from the surface in Figure 6(b) using the interpolation technique from Section 3. The asymmetric character of the (52) translates into a “saw-toothed” shape of the local volatility for high and low strikes.³

In Figure 8 we illustrate results from calibration f and g. As mentioned in Section 2.1, the parameter δ in (24) can be interpreted as a Tikhonov regularization [8] and the use of a larger δ should produce a smoother solution. This is also the case, as can be seen by comparing figure 8(a) and 6(a), two surfaces obtained with exactly the same algorithm but with different values for δ . When interpolated and redefined on a finer grid as in Section 3, the surface in Figure 8(a) obtained with the larger δ reproduces 336 out of the 348 used quotes within the spread, to compare with the 346 produced by the less smooth surface obtained with a smaller δ . Figure 8(b) illustrates calibration case g and thus the effect of running fewer loops when updating the limits for the volatilities in Algorithm 1, meaning that the space Σ we stop at will be smaller than in the default case b. This surface (after redefinition on a finer grid) reproduces 339 out of the 348 used quotes within the spread. Thus, the extra unsMOOTHNESS in local volatility in the surface in Figure 6(a) makes it reproduce an extra 7 out of the 348 options used as input data within the spread. This illustrates that a stopping criteria which demands less accuracy, or a smaller space Σ , might need to be used to obtain a smoother resulting volatility surface, at the cost of a slightly less good fit to the input data.

An illustration of the effect of the regularization penalties on the second derivative in space and the first derivative in time in the optimization of the piecewise local volatility as in (58) is given in Figure 9. The surface obtained without any smoothing (Figure 9(a), corresponding to case d) is clearly less smooth than the surface obtained with smoothing penalties (Figure 9(b), corresponding to case d). For this particular case, both surfaces reproduce 346 of the 348 quoted options within the spread. However, as seen from Figure 1 and 2, just as for the case where smoother input data is used, the smoothed surfaces does not in general provide a fit to data comparable to the surfaces obtained without penalties on the derivatives. We did illustrate though (Figure 3) that the absolute errors as measured by (59) stays small for the smoothed, piecewise defined local volatility even though a larger number of options are not replicated within the spread than for the less smooth cases.

In a similar way, it is in general considerably easier to obtain fairly smooth surfaces that reproduce the quoted market prices on options on the smaller OMX index, which has relatively few quotes, than on the SX5E index with quite a large number of quotes in a wider range of strikes. Figure 10 shows two surfaces that both reproduce all the 67 option quotes in our data set on OMX as of January 9 2013 within the spread. The surface in Figure 10(a) is obtained with the optimal control algorithm using a fine grid in T (case a) and that in Figure 10(a) is obtained for the piecewise local volatility with smoothing constraints (case e). Booth surfaces are smooth, illustrating the quite intuitive idea that it is easier to make a smooth representation of few data points than of a larger set.

Given a local volatility $\sigma(T, K) : \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$, the prices $C(T, K) : \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ that satisfy (2) for this σ , will be once differentiable in time T and two in space K . Even an unsmooth σ will thus lead to prices $C(T, K)$ that are at least $C^{1,2}$. A common practice is to display option prices $C(T, K)$ in terms of their equivalent Black-Scholes volatility $\sigma_{BS}(T, K)$ (see for example [6] for an account of the Black-Scholes model). In Figure 11, we display the Black-Scholes volatilities corresponding to prices obtained from the local volatility surfaces in Figure 7(a) (corresponding to

³In [2], a time-change in the interpolation scheme (52) is proposed which could help remedy this asymmetry. Unfortunately, we have not managed to make our method consistently work with such time-changes.

case c) and 9(b) (corresponding to case e) respectively. Both surface 11(a) and 11(b) are smooth, but as can be seen in Figure 11, the “saw-tooth pattern” stemming from the interpolation used to produce the local volatility surface 7(a) causes some minor fluctuations in the corresponding Black-Scholes volatility.

6 Conclusion

As mentioned in the introduction, the problem of calibrating a local volatility surface is underdetermined: we need around 10^5 grid points to find an accurate solution to Dupire’s equation (2) by a standard discretization, whereas we only have around 10^2 data points. The optimal control technique we exploit handles the difficulty of the underdetermined systems via the regularization of the Hamiltonian and makes it possible to handle problems with a fairly large number of variables by giving analytical access to the Jacobian of the resulting systems (43). Even so, the optimal control technique yields better results in terms of the number of quotes reproduced within the spread when we reduce the number of variables by using a coarser grid (as in calibration b, Section 5), which illustrates that the underdetermined character of the problem is indeed an important issue. The volatility that we consider in Section 4 is defined by a smaller number of grid points and the optimization problem (58) is therefore less severely underdetermined than the problems stemming from our optimal control algorithm.

We illustrated in Section 5 that regularization (such as the penalties on the derivatives in equation (58)) can give much smoother volatility surface to the price of a somewhat poorer fit to data. We also showed examples of several different local volatility functions that produce very similar prices. These two observations indicate that other criteria than the fit to market data must guide the choice of local volatility function. Such criteria could for example be based on how efficiently surfaces with different characteristics can be used for hedging derivative products.

The two optimization approaches (the technique based on optimal control and the optimization of a volatility function that is piecewise constant in time and piecewise linear in space) explored above both provide viable techniques for calibrating a local volatility function that fit observed market quotes on equity indices. The technique with a piecewise defined volatility function is appealing for its simplicity and it gives good results. The optimal control approach is appealing for the flexibility given by its fully non-parametric character and its efficiency in reproducing data. The best results in terms of goodness of fit come from this technique when we use the coarser grid in time and the interpolation technique described in Section 3. The improved results obtained when decreasing the number of variables in the optimal control problem indicate that optimal control techniques where the number of variables are fewer than the variables in the discretization of the underlying differential equations might be an interesting topic for future research.

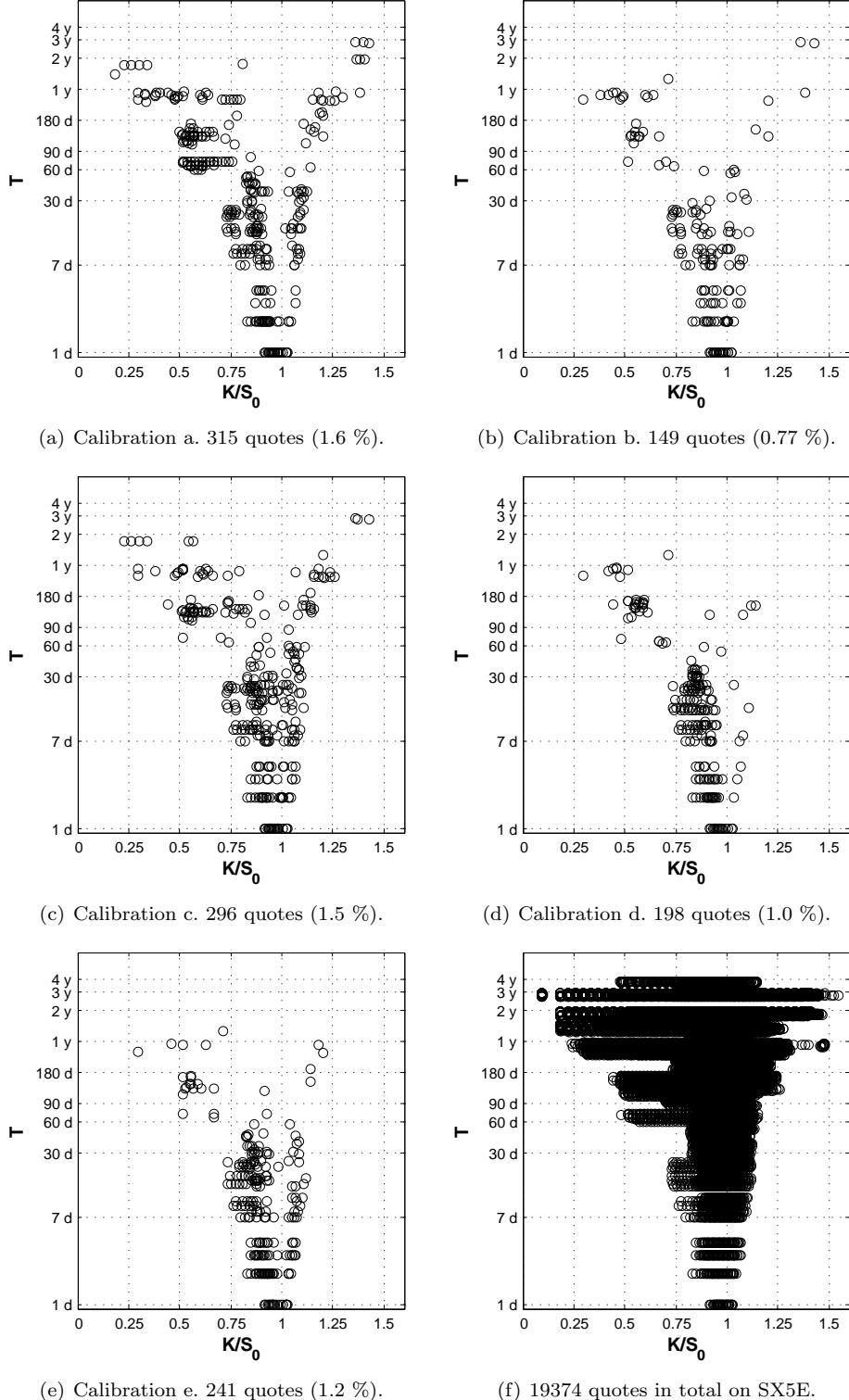


Figure 1: All the quoted maturities and strikes (in “moneyness” $\frac{K}{S_0}$) on SX5E (1(f)) and the quotes not reproduced within the spread for the calibration cases a to e (1(a) to 1(e)).

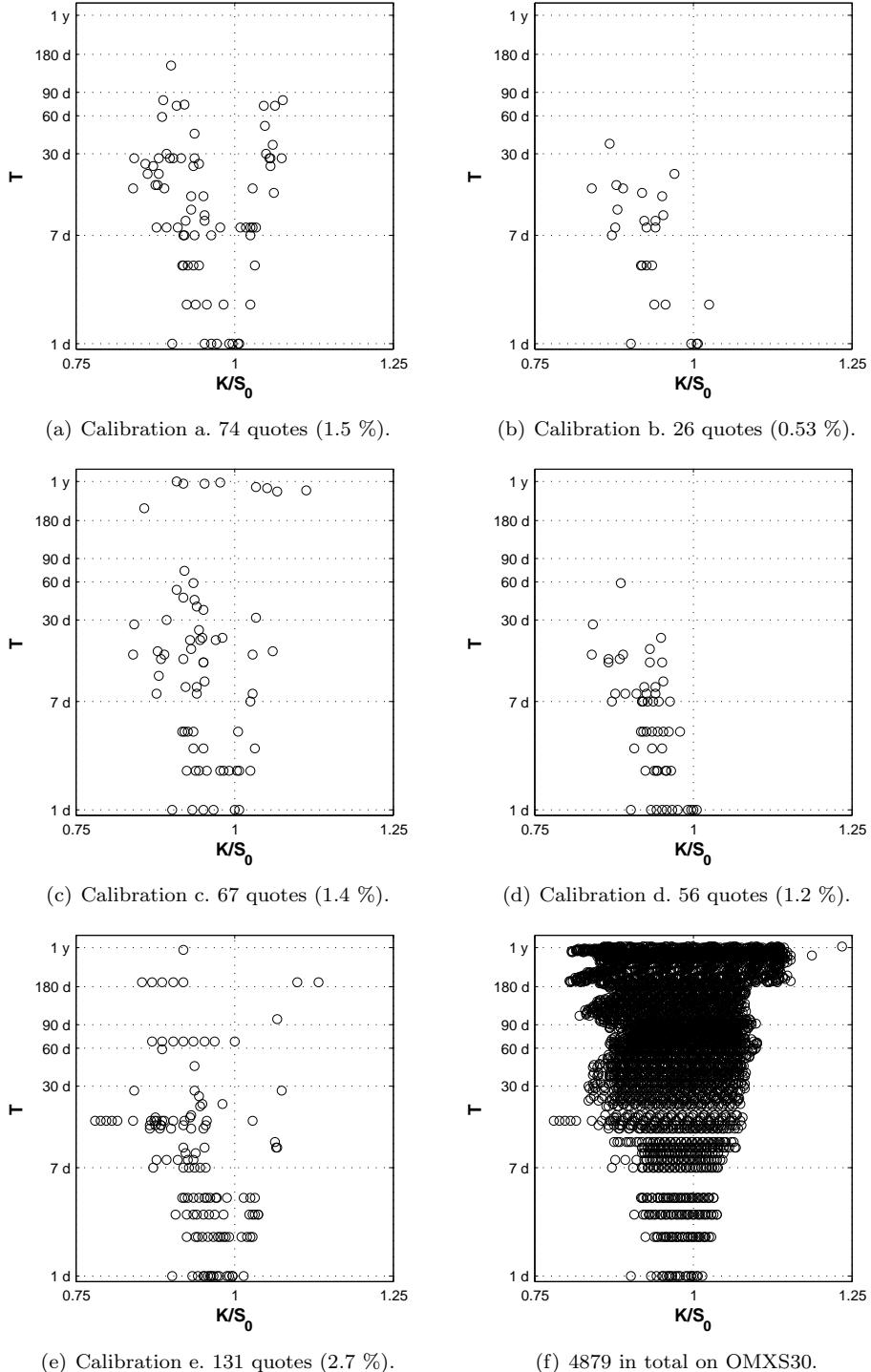
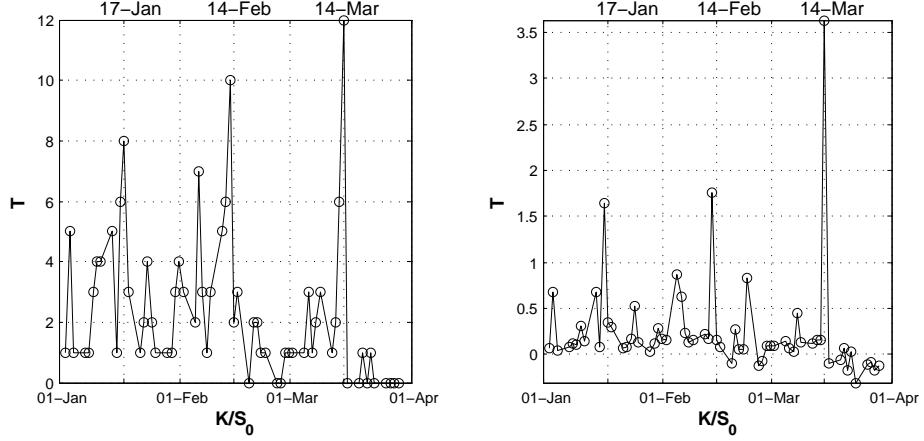
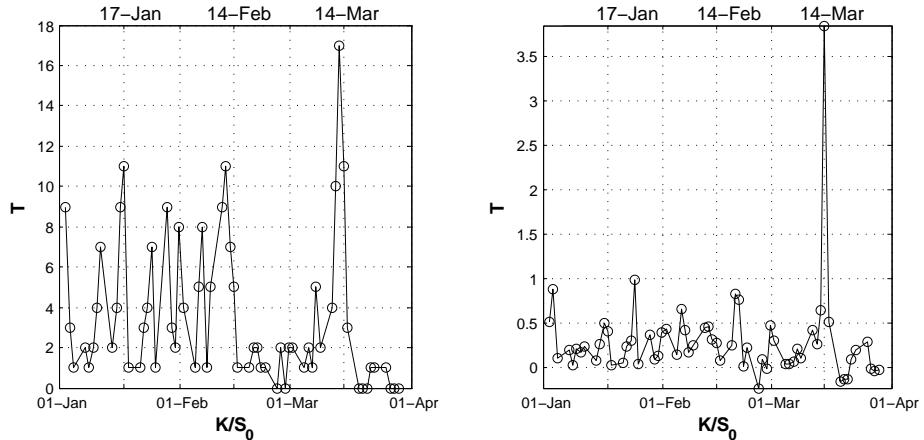


Figure 2: All the quoted maturities and strikes (in “moneyness” $\frac{K}{S_0}$) on OMX (2(f)) and the quotes not reproduced within the spread for the calibration cases a to e (2(a) to 2(e)).



(a) Number of options *not* replicated within the spread calibration b.

(b) The error measure $pd_{\text{bid}}^{\text{ask}}$ for calibration b.



(c) Number of options *not* replicated within the spread calibration e.

(d) The error measure $d_{\text{bid}}^{\text{ask}}$ for calibration e.

Figure 3: The number of quotes *not* replicated within the spread on a daily basis along with the error measure $d_{\text{bid}}^{\text{ask}}$ from (59) for calibration case b and e. The dates on the upper vertical axis indicate when one day is left to maturity for the options closest to expiry.

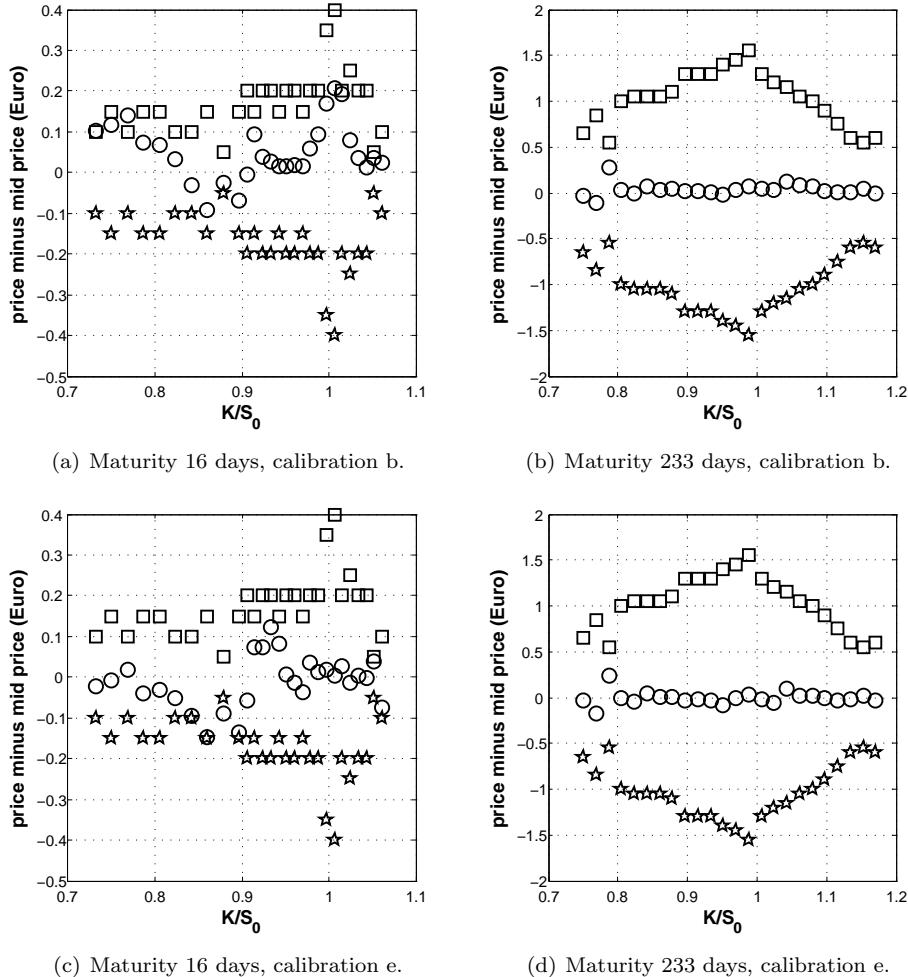
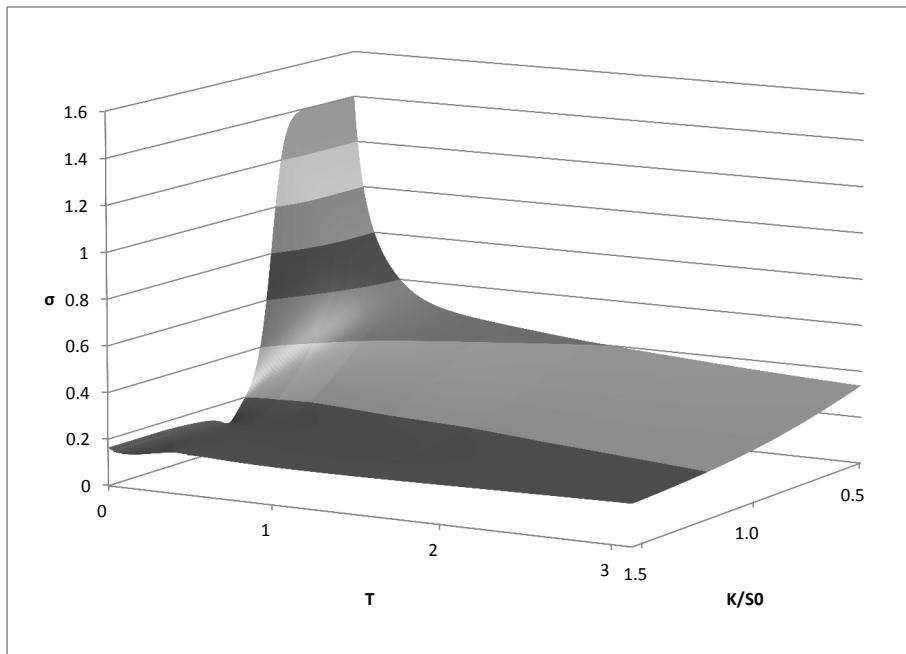
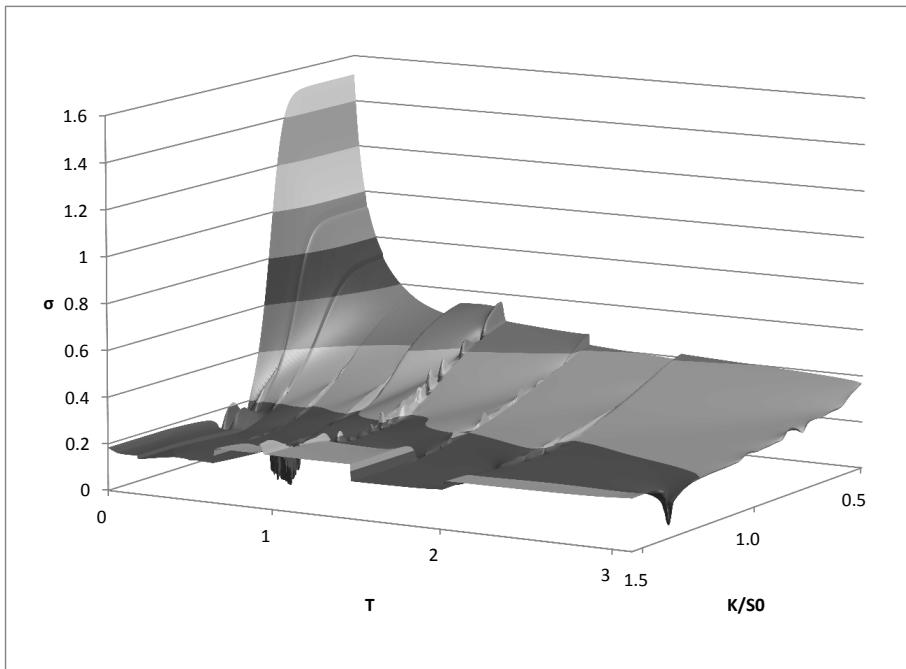


Figure 4: Quoted prices and model prices subtracted by market mid-prices on SX5E for two different maturities on January 30 2013. The spot price was 2732.75 Euros. \square = market ask prices, \star = market bid prices, \circ = prices from calibrated local volatility.

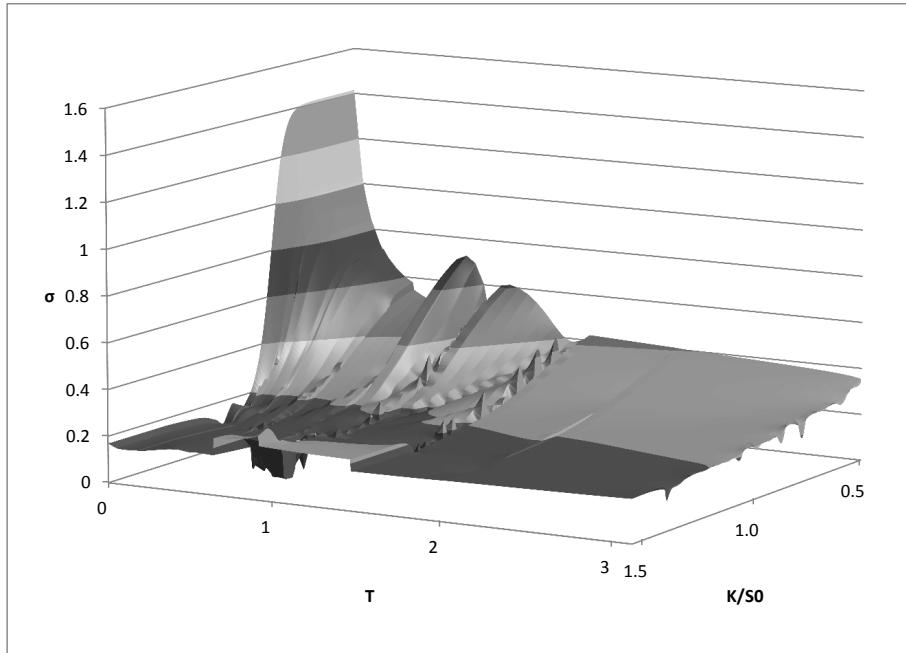


(a) Initial guess from affine model.

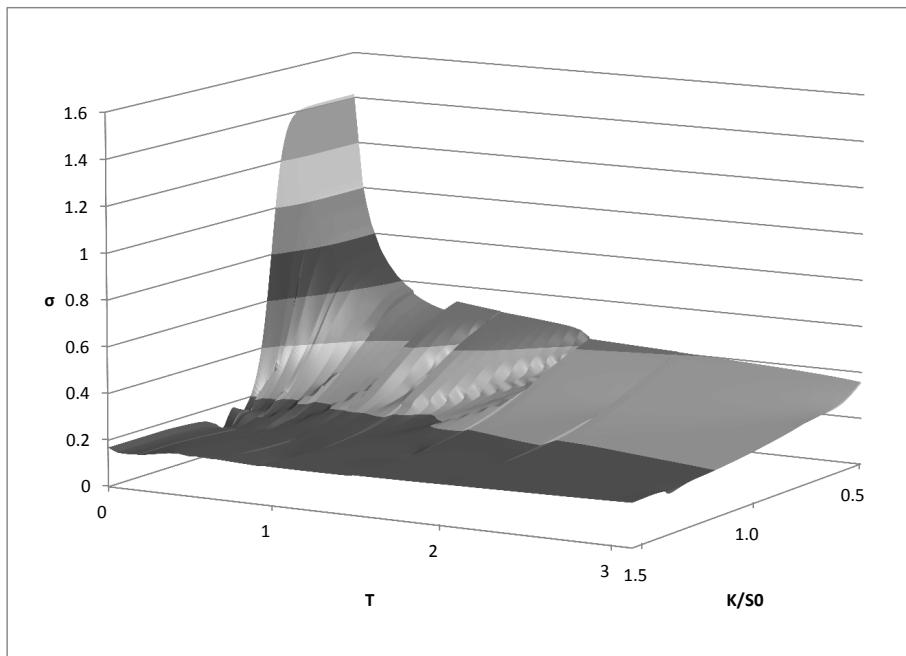


(b) Calibration a.

Figure 5: Local volatility on SX5E from January 30 2013. Initial guess obtained from our affine model and results from optimal control algorithm with a fine T -grid, calibration a.

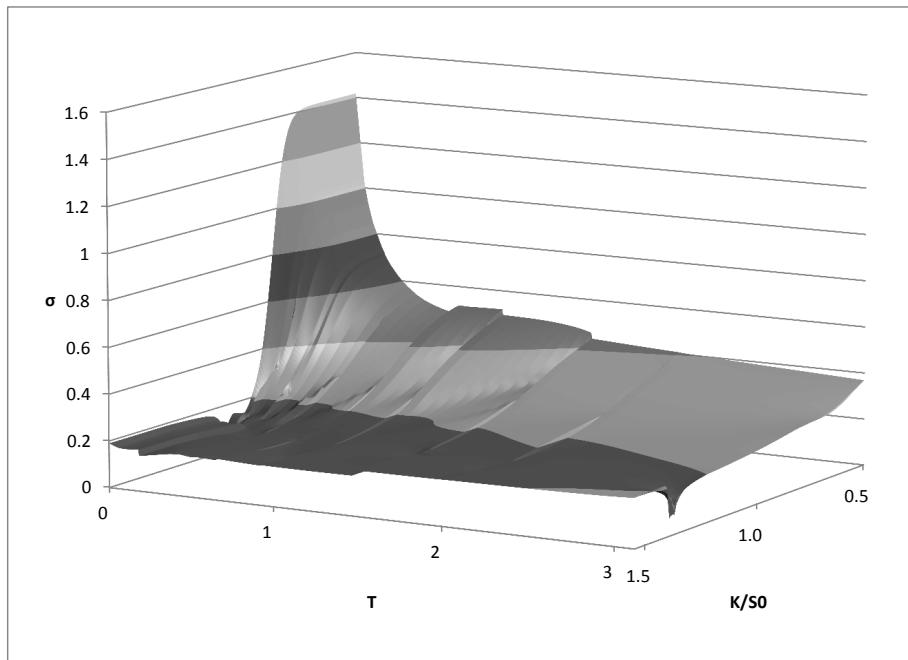


(a) Calibration b.

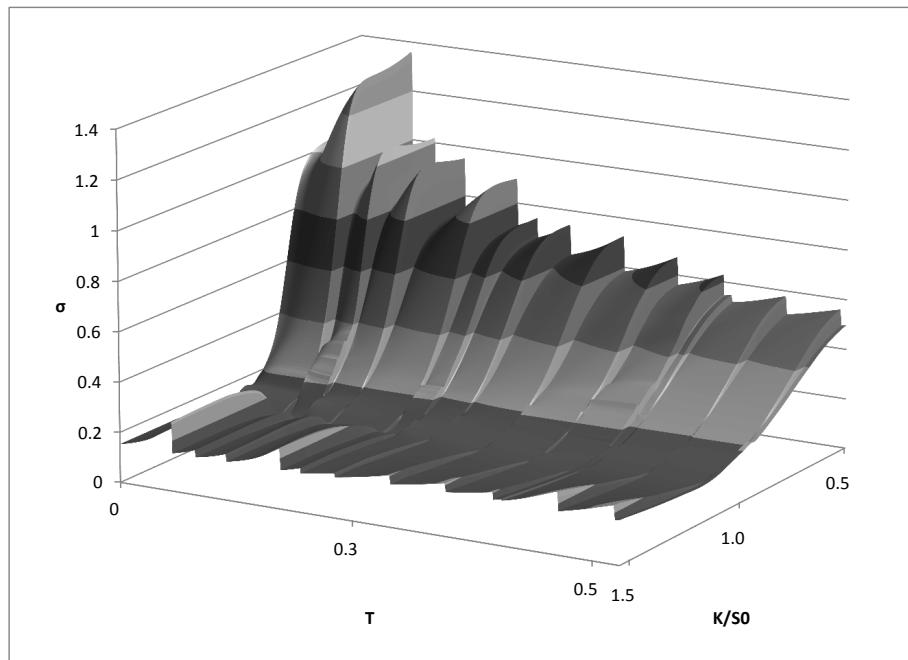


(b) Calibration c.

Figure 6: Local volatility functions on SX5E from January 30 2013. Results from optimal control algorithm with a coarse T -grid using an accurate approximation of market as input data (calibration b) and for using a somewhat smoother approximation of market prices as input data (calibration c).

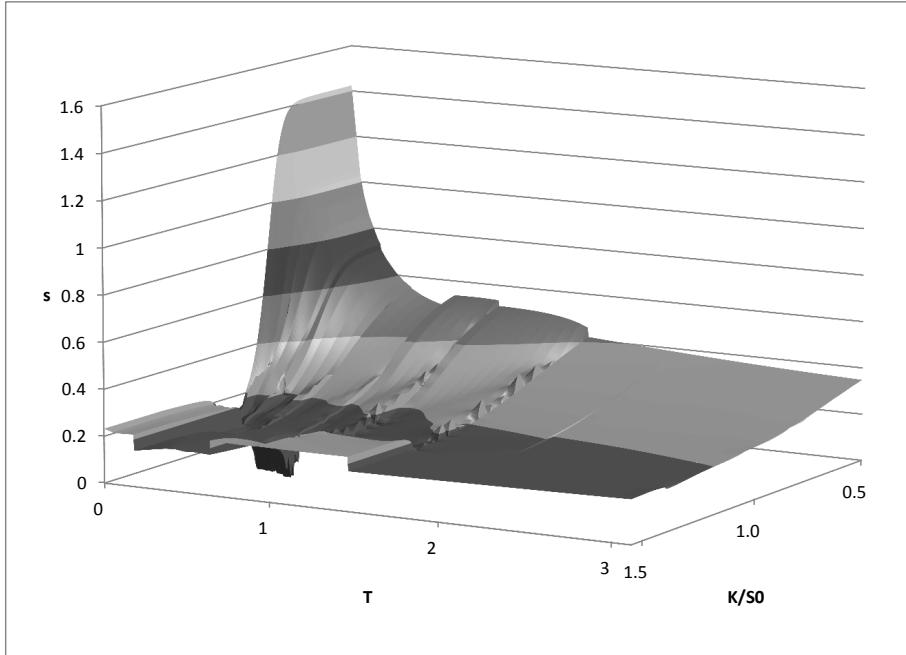


(a) Full domain in maturity T .

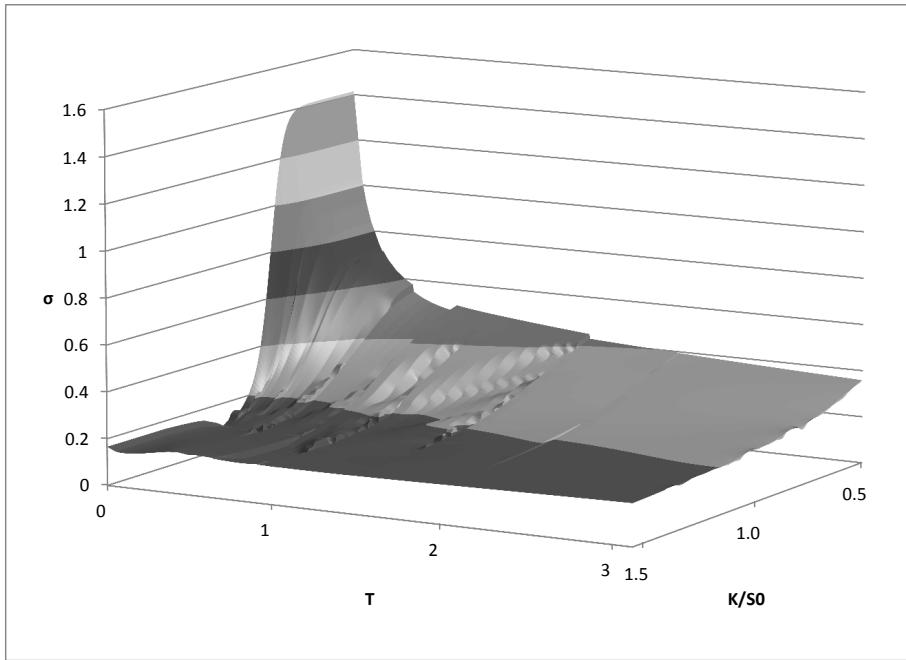


(b) Maturities T up to 0.5 years.

Figure 7: Local volatility constructed as described in Section 3 from the local volatility obtained with calibration c displayed in figure 6(b).

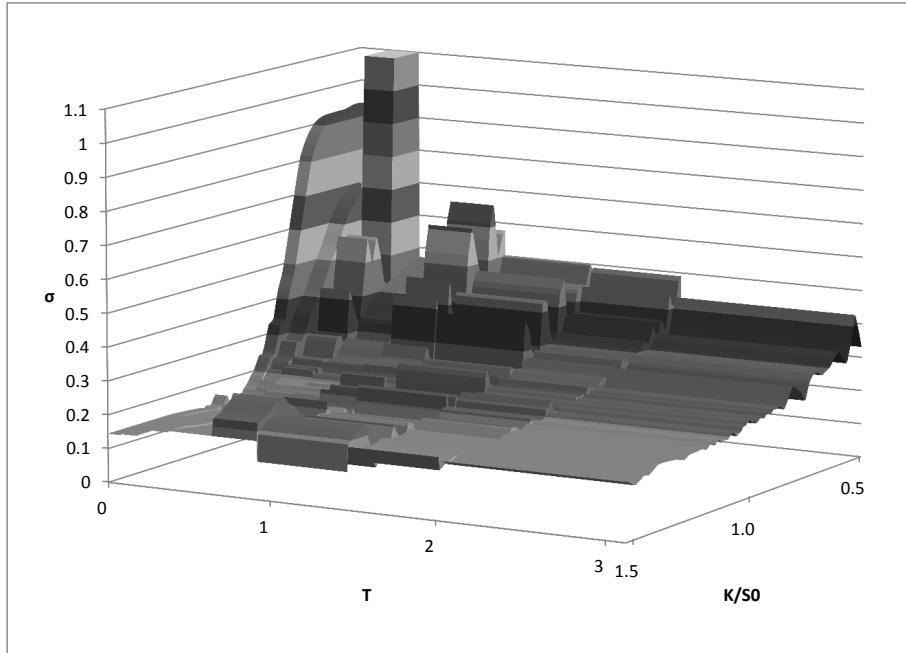


(a) Calibration b.

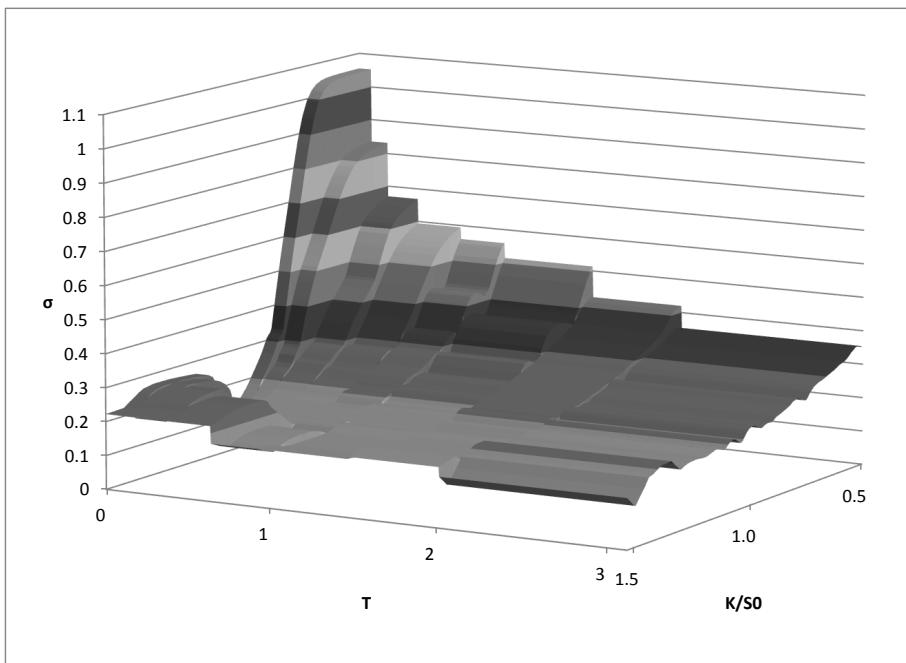


(b) Calibration c.

Figure 8: Local volatility functions on SX5E from January 30 2013. Figure 8(a) shows results from case f control algorithm with a coarse T -grid a large δ in (24) and Algorithm 1. Figure 8(a) shows results from case f using a smaller space Σ in Algorithm 1.

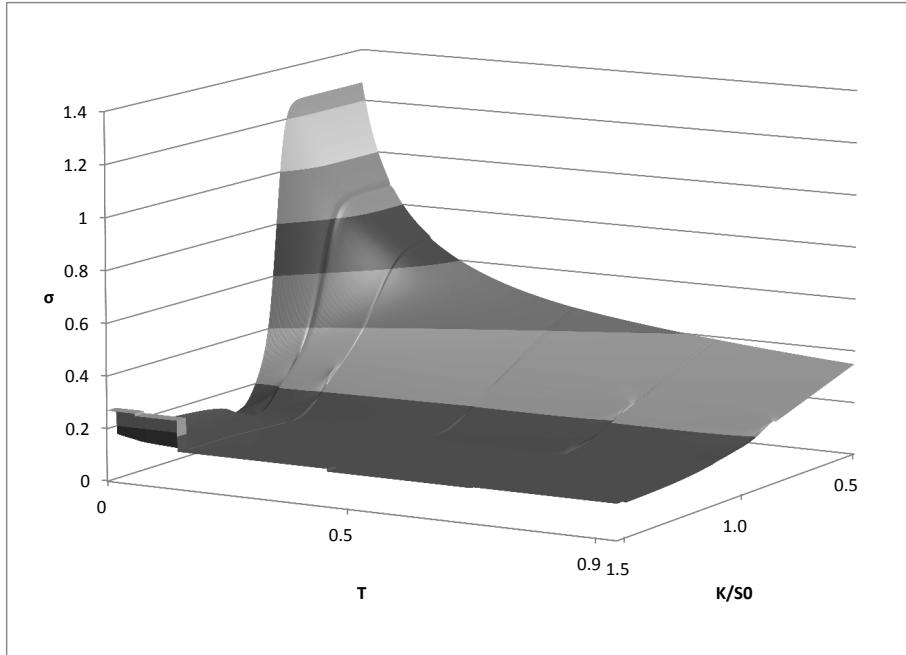


(a) Calibration d.

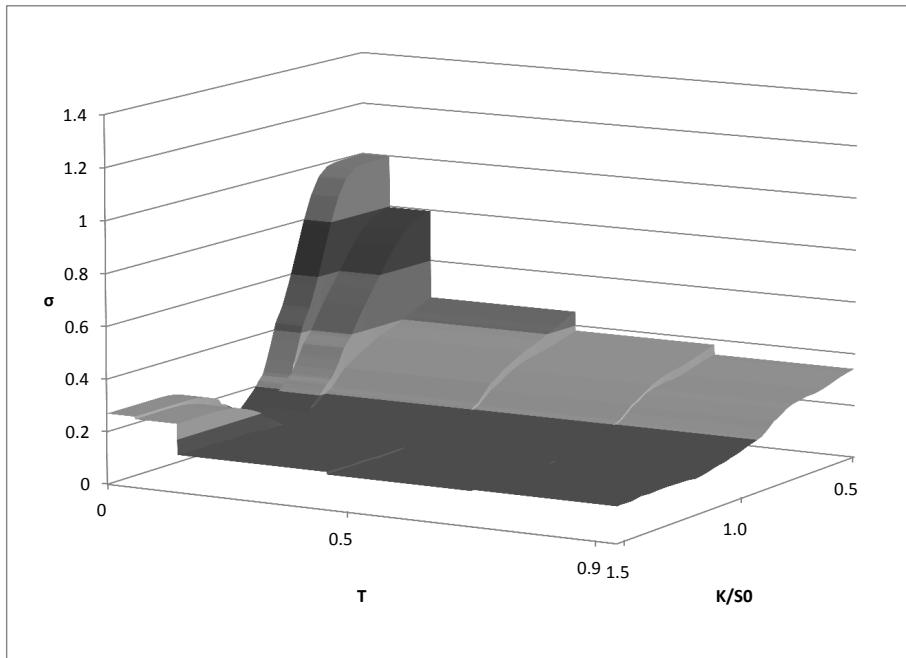


(b) Calibration e.

Figure 9: Local volatility functions on SX5E from January 30 2013. Results for a piecewise constant, piecewise linear volatility function without smoothing (calibration d) and with smoothing (calibration e).

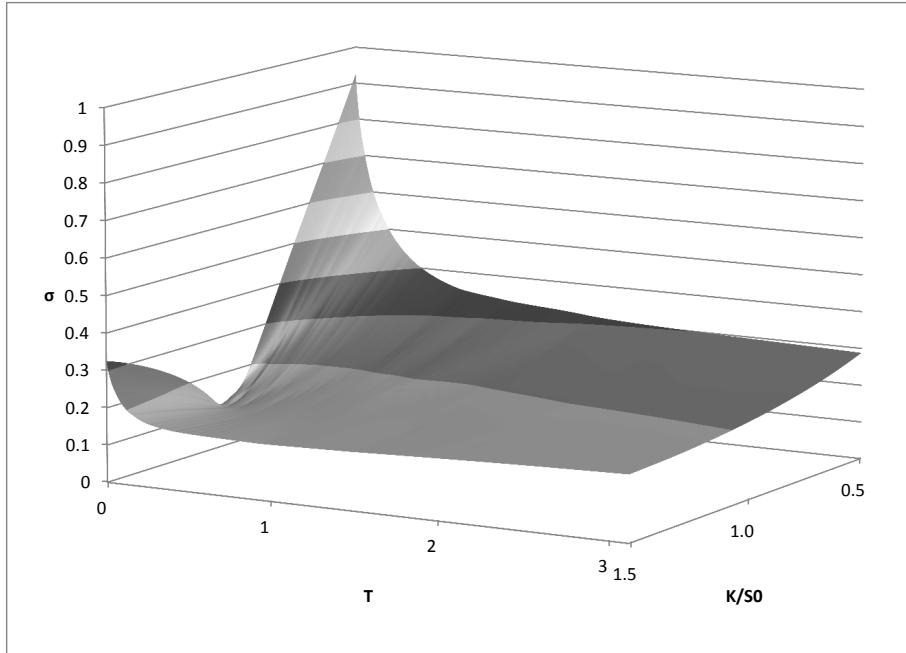


(a) Calibration a.

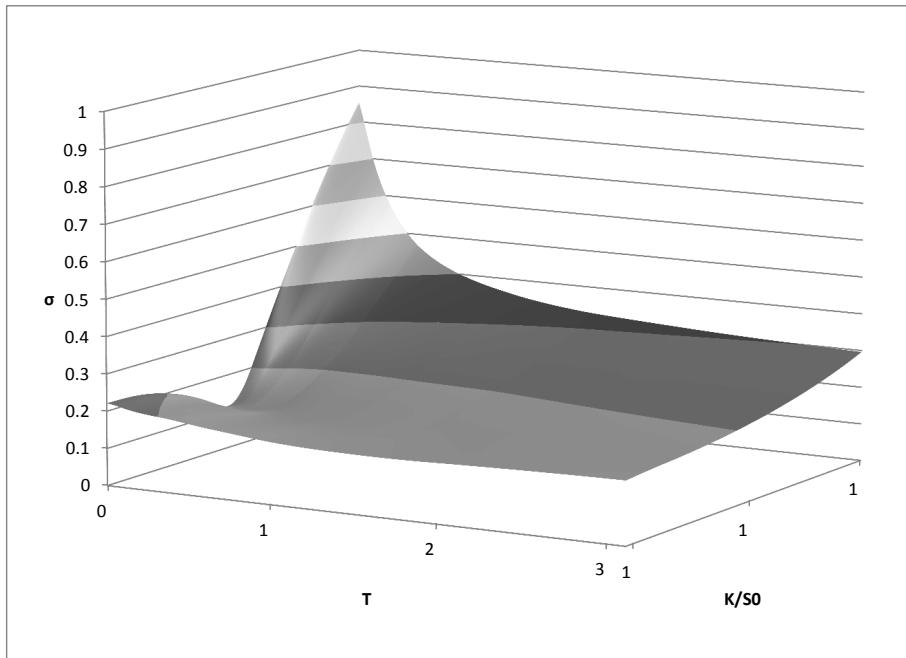


(b) Calibration e.

Figure 10: Local volatility functions on OMX from January 9 2013. Results for the optimal control algorithm with a fine T -grid (calibration a) and for the piecewise constant, piecewise linear volatility function with smoothing (calibration e).



(a) Corresponding to Figure 7(a).



(b) Corresponding to Figure 9(b).

Figure 11: Black-Scholes volatilities corresponding to local volatility surfaces on SX5E from January 30 2013. The two surfaces correspond to calibration cases a and e (the local volatilities in Figure 7(a) and Figure 9(b)).

References

- [1] Yves Achdou and Olivier Pironneau. Volatility smile by multilevel least square. *International Journal of Theoretical and Applied Finance*, 5(6):619–643, 2002.
- [2] Jesper Andreasen and Brian Huge. Volatility interpolation. *Risk Magazine*, pages 76–79, March 2011.
- [3] Mukarram Attari. Option pricing using fourier transforms: a numerically efficient simplification. *SSRN eLibrary*, 2004.
- [4] Marco Avellaneda, Craig Friedman, Richard Holmes, and Dominick Semperi. Calibrating volatility surfaces via relative-entropy minimization. *Applied Mathematical Finance*, 4:37–64, 1997.
- [5] Emmanuel Barron and Robert Jensen. The pontryagin maximum principle from dynamic programming and viscosity solutions to first-order partial differential equations. *Transactions of the American Mathematical Society*, 298(2):635–641, December 1986.
- [6] Tomas Björk. *Arbitrage Theory in Continuous Time*. Oxford University Press, 2004.
- [7] Charles George Broyden. The Convergence of a Class of Double-Rank Minimization Algorithms. *Journal Inst. Math. Applic.*, 6:76–90, 1970.
- [8] Jesper Carlsson, Mattias Sandberg, and Anders Szepessy. Symplectic Pontryagin approximations for optimal design. *ESAIM: Mathematical Modelling and Numerical Analysis*, 43:3–32, 2008.
- [9] Peter Carr and Dilip B. Madan. Option valuation using the fast Fourier transform. *Journal of Computational Finance*, 2(4):61–73, 1999.
- [10] Thomas F. Coleman and Yuying Li. On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds. *Mathematical Programming*, 67(1):189–224, 1994.
- [11] Thomas F. Coleman and Yuying Li. An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds. *SIAM Journal on Optimization*, 6:418–445, 1996.
- [12] Thomas F. Coleman, Yuying Li, and Arun Verma. Reconstructing the unknown local volatility function. *The Journal of Computational Finance*, 2(3):77–102, 1999.
- [13] Emanuel Derman and Iraj Kani. Riding on a Smile. *Risk Magazine*, 7(2):32–39, 1994.
- [14] Darrel Duffie, Jun Pan, and Kenneth Singleton. Transform Analysis and Asset Pricing for Affine Jump-Diffusions. *Econometrica*, 68(6):1343–1376, November 2000.
- [15] Bruno Dupire. Pricing with a smile. *Risk Magazine*, pages 18–20, January 1994.
- [16] Alberto Elices. Models with time-dependent parameters using transform methods: application to Heston’s model. *arXiv.org*, August 2007.
- [17] Lawrence C. Evans. *Partial Differential Equations*. American Mathematical Society, second edition, 2010.
- [18] Matthias R. Fengler. Arbitrage-Free Smoothing of the Implied Volatility Surface. *ISSN 1860-5664*, March 2005.
- [19] Roger Fletcher. A new approach to variable metric algorithms. *Computer Journal*, 13:317–322, 1970.

- [20] James Glover and Montaz M. Ali. Using radial basis functions to construct local volatility surfaces. *Applied Mathematics and Computation*, 217(9):4834–4839, 2011.
- [21] Donald Goldfarb. Family of Variable Metric Updates Derived by Variational Means. *Mathematics of Computing*, 24:23–26, 1970.
- [22] Steven L. Heston. 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, 1993.
- [23] Nicolas Jackson, Endre Süli, and Sam Howison. Computation of deterministic volatility surfaces. *Journal of computational finance*, 2(2):5–32, January 1998/99.
- [24] Jonas Kiessling. *Approximation and calibration of stochastic processes in finance*. PhD thesis, KTH, 2010.
- [25] Steven G. Kou. A Jump-Diffusion Model for Option Pricing. *Management Science*, 48(8):1086–1101, August 2002.
- [26] Ronald Lagnado and Stanley Osher. A technique for calibrating derivative security pricing models: numerical solution of an inverse problem. *Journal of Computational Finance*, 1(1):13–25, 1997.
- [27] Alex Lipton and Artur Sepp. Filling the gaps. *Asia Risk*, pages 85–91, October 2011.
- [28] Sergei Mikhailov and Ulrich Nögel. Heston’s Stochastic Volatility Model Implementation, Calibration and Some Extensions. *Wilmott magazine*, pages 74–79, July 2003.
- [29] Michael J.D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In Watsons, G.A., editor, *Numerical Analysis*, volume 630 of *Lecture Notes in Mathematics*. Springer Verlag, 1978.
- [30] M.J.D. Powell. The Convergence of Variable Metric Methods For Nonlinearly Constrained Optimization Calculations. In Mangasarian, O.L. and Meyer, R.R. and Robinson, S.M., editor, *Nonlinear Programming 3*. Academic Press, 1978.
- [31] Mattias Sandberg. *Approximation of optimally controlled ordinary and partial differential equations*. PhD thesis, KTH, 2006.
- [32] Mattias Sandberg and Anders Szepessy. Convergence rates of symplectic Pontryagin approximations in optimal control theory. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40:149–173, January 2006.
- [33] Wim Schoutens. The meixner process : theory and applications in finance. <https://perswww.kuleuven.be/~u0009713/004wsreport.pdf>, 2002.
- [34] Artur Sepp. Using SABR model to produce smooth local volatility surfaces. Merrill Lynch, July 2007.
- [35] Curtis R. Vogel. *Computational Methods for Inverse Problems*. SIAM, 2002.

A Treatment of market data

The value of the interest rate r_T in (1) at dates between those at which our given zero-coupon curve is defined are obtained through a simple cubic splines interpolation of the curve and subsequent numerical differentiation. At each maturity T we use values for the forward price F_T obtained from put-call parity for the option with strike closest to the money spot. The forward prices for different maturities are then linearly interpolated between maturities and the dividend yield q_t is obtained through the identity

$$F_T = S_0 B_T^{-1} e^{-\int_0^T q_t dt}, \quad (60)$$

where $B_T = e^{-\int_0^T r_t dt}$ is the zero-coupon bond of maturity T .

Quoted prices are not always arbitrage free, and before using our data we remove quotes that induce arbitrage opportunities. It is well established that absence of arbitrage implies that call options of a given maturity have prices that are decreasing and convex with respect to the strike. We therefore wish to remove any option from our data set that makes price quotes either non-decreasing or concave. However, quotes are given as bid- and ask prices so the question is if for a given maturity there is a decreasing, concave curve that fits within the spread (5) of all observable quotes. To test this criterion we use a spline representation constrained to be convex and decreasing by the technique in [18] (we do not use any smoothness penalty; see also Appendix C). We remove any prices such that the obtained spline interpolation deviates from the mid-price by more than $\min(\frac{1}{2} \text{spread}, 20 \cdot 10^{-4} S_0)$.

For the condition of no-arbitrage for the relation between call options of different maturities, we first write S_T from (1) as

$$S_T = S_t e^{\int_t^T \sigma(u, S_u) dW_u - \int_t^T \sigma^2(u, S_u) du + \int_t^T (r_u - q_u) du} \quad (61)$$

for $t \leq T$, and then use the conditional Jensen's inequality to obtain

$$E[(S_T - K)_+] | \mathcal{F}_t \geq (E[S_T | \mathcal{F}_t] - K)_+, \quad (62)$$

where \mathcal{F}_t is taken to be the natural filtration of S_t . From (61) and (62) we then have

$$E[(S_T - K)_+] \geq e^{\int_t^T (r_u - q_u) du} E[(S_t - K e^{-\int_t^T (r_u - q_u) du})_+] \quad (63)$$

and by multiplying both sides with $e^{-\int_0^T r_u du}$ we get

$$e^{-\int_0^T r_u du} E[(S_T - K)_+] \geq e^{-\int_t^T q_u du} e^{-\int_0^t r_u du} E[(S_t - K e^{-\int_t^T (r_u - q_u) du})_+]. \quad (64)$$

If $\text{Call}(T, K)$ is the price of a call option written on S with strike K and maturity T , then (60) gives that (64) can be written as

$$\frac{B_t F_t}{B_T F_T} \text{Call}(T, K) \geq \text{Call}(t, K \frac{F_t}{F_T}). \quad (65)$$

In order to remove arbitrage opportunities between a call option of strike K and maturity T and call options of a shorter maturity $t < T$, we therefore need to verify that (65) is satisfied. However, if K is among the strikes for which prices are quoted at T , there is not in general a quote for $K \frac{F_t}{F_T}$ at t . To test if our data satisfies (65), we therefore again use the spline representation obtained according to [18] to approximate prices at the required strikes for which market prices are not available.

B Constructing a local volatility from an affine stochastic volatility model

So called affine stochastic volatility models are extensively used in finance because they provide analytical tractability that allows the prices of standard options to be expressed as improper integrals that can be efficiently evaluated numerically. This tractability has two components. The first is that the characteristic function of the log-process can be expressed as a function of the solution to a Riccati ordinary differential equation (ODE). These Riccati equations can in certain cases be solved explicitly, which was first done by Heston [22] for his continuous diffusion model. The technique has later been extended to a wider class of models with a certain affine structure by Duffie, Pan and Singleton [14] and to models with time-dependent parameters in Elice [16] and Mikhailov, Nögel [28]. The second component is the calculation of prices of call options as indefinite integrals in which the characteristic function appears in the integrand. Such a formulation of option prices in affine models was first done by Carr and Madan [9]. Below we will use a later version of this Fourier inversion integration which is due to Attari [3].

We will make use of an affine stochastic volatility model to approximate the prices of all observable options in the market place and use the analytical tractability of the model to calculate a local volatility function that reproduces the model's prices. The model will not perfectly match the market prices, but we will use the resulting local volatility as our initial guess for the algorithms in Sections 4 and 2.

We will use a stochastic volatility (under our chosen pricing measure \mathbb{Q}) with spot jumps according to

$$\begin{aligned} \frac{dS_t}{S_{t-}} &= \left(r_t - q_t - \eta_\zeta(-i) - \eta_\xi(-i) - \frac{1}{2}y_t \right) dt + \sqrt{y_t} dW_t^S + (e^{d\zeta_t} - 1) + (e^{d\xi_t} - 1) \\ dy_t &= \kappa(\eta - y_t) dt + \theta \sqrt{y_t} dW_t^y, \end{aligned} \quad (66)$$

where the short rate r_t and the dividend yield q_t are deterministic and (W^S, W^y) is a Brownian motion with correlation parameter ρ . The constants κ, η, θ are all positive. The jumps are defined by two mutually independent Lévy-process ζ_t and ξ_t such that the couple (ζ_t, ξ_t) is independent of (W^S, W^y) . The Lévy symbol of ζ and ξ are η_ζ and η_ξ respectively, i.e., the characteristic function of ζ_t is given by $\Phi_\zeta(u, t) = E[e^{iu\zeta_t}] = e^{tu\eta_\zeta}$, and likewise for ξ . Then with $i = \sqrt{-1}$, the component $\eta_\zeta(-i)$ that appears in the drift is the logarithm of the Laplace transform of ζ_1 . In other words, we use the Heston model [22] with Lévy-jumps in the log-spot process. Specifically, we will let our jump process ζ_t be the Meixner process [33] and ξ_t be a compound Poisson process with intensity λ and jumps following an asymmetric Laplace distribution (or double exponential distribution), i.e., the jump density ν is given by

$$\nu(z) = \frac{1}{\mu_- + \mu_+} \left(1_{\{z < 0\}} e^{\frac{z}{\mu_-}} + 1_{\{z \geq 0\}} e^{-\frac{z}{\mu_+}} \right), \quad (67)$$

for positive constants μ_- , μ_+ . (This choice of jump distribution is similar - but not identical - to the one used by Kou [25]. In particular, the density (67) is always continuous whereas the one in [25] can be discontinuous.) As mentioned above, it is well established that the characteristic function for the log-process $\ln S_t$ can be explicitly calculated.

Attari's result for the price of a European option when the characteristic function of the underlying's dynamics is known reads as follows. Let

$$F(t, T) = S_t e^{\int_t^T (r_s - q_s) ds} \quad (68)$$

be the forward price at t with maturity T of the underlying asset S from (66), and let $P(t, T)$ be the corresponding zero-coupon bond. At time t , the price of a European call option on S with

strike K and maturity T is given as

$$\text{Call}(t, T, K) = P(t, T) \left\{ F(t, T) - \frac{1}{2}K - K \left[\frac{1}{\pi} \int_0^\infty \frac{\left(\text{Re}_\varphi + \frac{\text{Im}_\varphi}{\omega} \right) \cos(l\omega) + \left(\text{Im}_\varphi - \frac{\text{Re}_\varphi}{\omega} \right) \sin(l\omega)}{1 + \omega^2} d\omega \right] \right\}. \quad (69)$$

We use the abbreviations $\text{Re}_\varphi = \text{Re } \varphi(\omega)$, $\text{Im}_\varphi = \text{Im } \varphi(\omega)$ where $\varphi(\omega) = E^{\mathbb{Q}} [e^{i\omega x(t, T)} | F_t]$ is the characteristic function of $x(t, T) = \ln \frac{F(T, T)}{F(t, T)}$ under the pricing measure \mathbb{Q} and $l = \ln \frac{K}{F(t, T)}$ and $F(t, T)$ is as in (68).

A control on the numerical error in the evaluation of the integral in (69) is crucial for our purposes. Firstly, the calibration of the parameters of our chosen model will involve the iterative solving of an optimization problem where the prices obtained (69) appear in the objective function calculation. We therefore need to calculate model prices for many different choices of parameters and it is important that we can control the error in the numerical integration in a fashion that does not depend on the parameters. Secondly, we want to exploit Attari's pricing formula together with Dupire's formula (3) to calculate the local volatility function that replicates our model prices. In order to use (3) we will also need to differentiate (69), as we describe below. This step will not yield satisfactory results without a high numerical accuracy in the evaluation of the pricing integral.

The calculation of indefinite integrals like (69) by numerical quadrature involves two kinds of errors: truncation error (from replacing the infinite domain by a finite domain) and quadrature error (from the approximation of the integrand by simpler functions on a grid in the integration domain). We can, however, eliminate the truncation error in the evaluation of (69) by a change of variables that transforms our integral to a summation on the interval $[0, 1]$. Once we have a definite integral to evaluate, the discretization error can be handled by choosing an adaptive quadrature rule that calculates the integral to desired precision.

We choose the simple change of variables $\omega = \frac{v}{1-v}$ to transform the integral in (69) into a definite integral,

$$\text{Call}(t, T, K) = P(t, T) \left\{ F(t, T) - \frac{1}{2}K - K \left[\frac{1}{\pi} \int_0^1 \frac{\left(\text{Re}_\varphi + \frac{(1-v)}{v} \text{Im}_\varphi \right) \cos \left(l \frac{v}{1-v} \right) + \left(\text{Im}_\varphi - \frac{(1-v)}{v} \text{Re}_\varphi \right) \sin \left(l \frac{v}{1-v} \right)}{(1-v)^2 + v^2} dv \right] \right\}, \quad (70)$$

where l is as in (69) and

$$\text{Re}_\varphi = \text{Re } \varphi \left(\frac{v}{1-v} \right), \quad \text{Im}_\varphi = \text{Im } \varphi \left(\frac{v}{1-v} \right). \quad (71)$$

To calculate a given price to a desired precision we apply the Gauss-Kronrod quadrature rule to the definite integral in (70). A detail worth mentioning for the purpose of computational efficiency is that the parts of the integrand that contain the characteristic function of x_T do not depend on the strike K . If we want to obtain the prices for options of different strikes K but the same maturity T , we can therefore calculate the characteristic function once for all strikes and apply the adaptive Gauss-Kronrod scheme simultaneously for all contracts. This means that we calculate the integral in (70) on a common grid for several different integrands that only differ through their respective value of K . Since the characteristic function is the most numerically costly part of the integrand, this considerably speeds up calculations, especially if the strikes are fairly close to each other, which means that the integrands are close and therefore well resolved by the same grid.

We now return to the Dupire formula (3). In order to find the local volatility function σ that reproduces the same call option prices as our model (66), we need to differentiate our prices $\text{Call}(t, T, K)$ from (70) once in T and twice in K . The derivatives of our call prices $\text{Call}(t, T, K)$ is calculated through direct differentiation of (70) where we differentiate the integral under the integral sign. We then obtain $\partial_T C$, $\partial_K C$ and $\partial_{KK} C$ numerically by use of the same quadrature rule as before. (Note that the time-derivative of the price will involve the time-derivative of the characteristic function φ , which is known since the characteristic function is calculated as the solution to a Riccati ODE in the time-variable T . (See [22], [14].). Both the numerator and the denominator in (3) approach zero as K approaches zero or infinity. Especially for short maturities T , the price C as function of K is close to its initial value $(S_0 - K)_+$ and the second derivative $\partial_{KK} C$ will be small for most K . For these reasons, high numerical accuracy is necessary in order to calculate σ as the quotient of derivatives (3), and especially so for short maturities. The technique described above to handle the error terms in the integral that yields the prices allows us to achieve the desired precision.

Nonetheless, for very short maturities, it is difficult to obtain the local volatility this way except for strikes K close to S_0 . We therefore perform the calculations for an interval of strikes - centered around the spot S_0 - that increases with the maturity T . We also monitor the numerical values obtained for the derivatives $\partial_T C$, $\partial_K C$ and $\partial_{KK} C$ and only calculate σ from (3) if their absolute values are significantly above the precision prescribed for the adaptive quadrature. We extrapolate the calculated values to cover the full desired range of strikes and maturities.

Recall that the purpose for which we wanted to exploit the affine model (66) is to create a smooth local volatility surface that will give a fair approximation of the observable call prices. To this end, we first search for model parameters that best approximate observed market data in a least-square sense. Once a set of parameters are found from (72), the corresponding local volatility is calculated by the use of (3) and differentiation of (70) as described above. Now let α denote a vector containing the scalar parameters in our model (66). We can let α be defined by $\alpha = [y_0, \kappa, \eta, \theta, \rho, \lambda, \mu_-, \mu_+, a, b, d]$ where y_0 is the initial value of the variance, ρ is the correlation between the Brownian motions, λ, μ_-, μ_+ are the parameters of the compound Poisson process with Laplace distributed jumps, a, b and d are the parameters of the characteristic function of the Meixner process [33] and κ, η, θ are defined in (66). We let $C^\alpha(T, K)$ be the model's price of a call option for a given α . Now, as in (4), let \mathcal{I} be the set maturity-strike pairs for which market prices are available. We try to find the parameters α^* that give the resulting model prices that best reproduce the market mid-prices (33) in a weighted quadratic norm according to

$$\alpha^* = \underset{\alpha \leq \bar{\alpha}}{\operatorname{argmin}} \sum_{i=1}^M \sum_{j=1}^{N_i} \omega_j^i \left[C^\alpha(\bar{T}_i, \bar{K}_j^i) - \bar{C}(\bar{T}_i, \bar{K}_j^i) \right]^2 \quad (72)$$

for some vectors $\alpha, \bar{\alpha} \in \mathbb{R}^d$ that define upper and lower boundaries for the parameters. For the weights ω in (72) we use

$$\omega_j^i := \frac{N \bar{T}^{i\gamma}}{N_i \sqrt{\operatorname{spread}(\bar{T}^i, \bar{K}_j^i)}} \quad (73)$$

with $N = \#\mathcal{I}$ and the price spread as in (5). We thus choose a higher weight for those prices that are given with a smaller spread. The factor $\frac{N}{N_i}$ reflects that we want to be accurate on options at a given maturity even if there are few prices observable for this particular date. We use $\bar{T}^{i\gamma}$ with $\gamma = \frac{1}{\log_2(\bar{T}^1)}$ to give a higher weight to options with short maturities, since these are the most difficult ones to match. We thus choose a higher weight for those prices that are given with a smaller spread. The factor $\frac{N}{N_i}$ reflects that we want to be accurate on options at a given maturity even if there are few prices observable for this particular date. We use $\bar{T}^{i\gamma}$ with $\gamma = \frac{1}{\log_2(\bar{T}^1)}$ to give a higher weight to options with short maturities, since these are the most difficult ones to match.

C Interpolating option prices in strike

The optimal control algorithm in Section 2 is based on a finite difference discretization in which the grid-points in strike range will not necessarily coincide with observable market data. We therefore replace observed market prices by a continuous and arbitrage free representation in the strike range for each observable maturity. In addition to providing prices at non-observable strikes, we aim to find a representation that is smoother than the market prices, thus increasing the chances of obtaining a smooth local volatility from our algorithm.

The practice of smoothing and interpolating prices in the strike dimension is quite common in finance. Two commonly used techniques are the parameterized approximations of implied volatility with the SABR model **hagan**, **andreasen**, and the so called SVI approach **gatheral**, constructed to resemble the implied volatilities produced by stochastic volatility models and often used for equity options.

We will use a combination of two different techniques. First, we will fit one set of parameters per maturity from our model (66) to observable data and use the prices produced by those parameters as a smooth interpolant of data. The use of this interpolation technique usually gives good results in terms of smoothness in the local volatility resulting from the algorithm we develop, and this is therefore our preferred method. Although this interpolation usually yields good results, we do not always come as close as we would like to the mid-prices when prices are given with tight spreads. We therefore make use of the method from [18] to obtain an arbitrage free cubic spline approximation of prices in the strike range. Since the model based method is our method of choice, we use a convex combination of the two representations where we chose as high a weight as possible on the model representation.

Suppose that $\{T^i\}_1^M$ are the maturities with available market data and let $\bar{C}(T^i)$ be a vector with the corresponding prices for different strike at maturity T^i and suppose we have N_i observable prices at maturity T^i . Let $C^\alpha(T^i)$ be the corresponding model prices for a given choice of parameters α . To approximate market prices for maturity T^i with our affine model, we search for parameters α^i that satisfy

$$\alpha^i = \underset{\alpha \leq \bar{\alpha}}{\operatorname{argmin}} \sum_{j=1}^{N_i} \omega_j^i [C^\alpha(\bar{T}_i, \bar{K}_j^i) - \bar{C}(\bar{T}_i, \bar{K}_j^i)]^2, \quad (74)$$

for weights ω given by

$$\omega_j^i := \frac{1}{\sqrt{\operatorname{spread}(\bar{T}_i, \bar{K}_j^i)}}. \quad (75)$$

The cubic spline representation g from [18] of market prices $\bar{C}(\bar{T}_i, \bar{K}_j^i)$ solves

$$\hat{g}^i = \min_g \left(\sum_{j=1}^{N_i} (\bar{C}_j - g(K_j)) + \lambda \int_{K_1}^{K_{N_i}} g''(K) dK \right) \quad (76)$$

under linear constraints that guarantee absence of arbitrage. For our purposes, we use this implementation with the smoothness penalty $\lambda = 0$.

Our convex combination \hat{C}^i of the model interpolation C^{α^i} from (74) and the spline \hat{g} will then be

$$\hat{C}^i = (1 - \theta)\hat{g}^i + \theta C^{\alpha^i}, \quad (77)$$

where we choose θ as close to 1 as is possible while still giving an accuracy such that \hat{C}^i does not deviate from mid-prices by more than $0.15 \times \operatorname{spread}$.