Quantitative Finance

Non-parametric calibration of the local volatility surface for European options using a second-order Tikhonov regularization

Jian Geng [a] , I. Michael Navon [b] & Xiao Chen [c]

[a] Department of Mathematics , Florida State University , Tallahassee , FL , USA

[b] Department of Scientific Computing , Florida State University , Tallahassee , FL , USA

[c] Center for Applied Scientific Computing , Lawrence Livermore National Laboratory , Livermore , CA , USA
Published online: 11 Sep 2013.

PLEASE SCROLL DOWN FOR ARTICLE

Routledge
Taylor & Francis Group

Non-parametric calibration of the local volatility surface for European options using a second-order Tikhonov regularization

JIAN GENG†, I. MICHAEL NAVON*‡ and XIAO CHEN§

†Department of Mathematics, Florida State University, Tallahassee, FL, USA
‡Department of Scientific Computing, Florida State University, Tallahassee, FL, USA
§Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA, USA

We calibrate the local volatility surface for European options across all strikes and maturities of the same underlying. There is no interpolation or extrapolation of either the option prices or the volatility surface. We do not make any assumption regarding the shape of the volatility surface except to assume that it is smooth. Due to the smoothness assumption, we apply a second-order Tikhonov regularization. We choose the Tikhonov regularization parameter as one of the singular values of the Jacobian matrix of the Dupire model. Finally we perform extensive numerical tests to assess and verify the aforementioned techniques for both volatility models with known analytical solutions of European option prices and real market option data.

Keywords: Local volatility surface; Second-order Tikhonov regularization; SVD; Large scale non-linear inverse problem

JEL Classification: C61, C

1. Introduction

Local volatility model is an extension of the Black–Scholes constant volatility model (Black and Scholes 1973) aimed at explaining the volatility smiles observed in the market. It assumes the volatility term is a deterministic function of both stock price and time. Dupire in his seminal work (Dupire 1994) established that the local volatility function can be uniquely derived from European option prices given the existence of European options with all strikes and maturities. However, in the market, there is only a limited number of available European options with discrete strikes and maturities. Up to date, there have been quite a number of studies addressing the reconstruction of local volatility function from a limited number of options available in the market.

Lagnado and Osher (1997) first solved the calibration problem in a PDE framework without assuming any shape of the local volatility function, i.e. a non-parametric approach. They used the first order derivatives of the volatility surface to regularize the inverse problem. Most subsequent research followed the same regularization approach such as Bouchouev and Isakov (Bouchouev and Isakov (1997, 1999)), Bodurtha, Jr. and Jermakyan (1999), Jiang and Tao (2001), Jiang *et al.* (2003), Crepey (2003), Egger and Engl (2005), Hein (2005), Achdou and Pironneau (2005) and Turinici (2009). However, the recovered volatility surface is usually very rough and it is hard to discern any patterns. Coleman *et al.* (1999), Achdou and Pironneau (2005) and Turinici (2009) solved the calibration problem using a parametric approach : the volatilities at several specially chosen points on the volatility surface are computed first, and then the volatility surface is constructed from those points using either linear interpolation or cubic splines. By parameterizing the volatility surface this approach reduces the dimension of the calibration problem. It works well when the chosen points can represent well the key regions of true volatility surface. However, it runs the danger of allowing too few degrees of freedom to explain the data. The recovered volatility surface is still either too rough (for the linear interpolation case) or subject to extreme values(for the cubic spline case) especially for the market data.

In this paper, we still use a non-parametric approach for the calibration of the local volatility surface with the only assumption that a smooth volatility surface is more preferable

---

*Corresponding author. Email: inavon@fsu.edu
This article was originally published with errors. This version has been corrected. Please see Erratum (http://dx.doi.org/10.1080/14697688. 2013.844894).

than a non-smooth volatility surface. This assumption is inspired by the 'Occam's razor', a principle from the fourteenth century philosopher William of Ockham, who argued that simpler explanations should be preferred to more complicated explanations. We seek a solution of smaller size in addition to the purpose of matching the market data. The same idea was introduced for solving non-linear inverse problems by Constable *et al.* (1987). We hope that by imposing the smoothness assumption, a simpler solution can be obtained from which some patterns maybe detectable. The smoothness preference assumption has already been adopted in the approach in Coleman *et al.* (1999), Achdou and Pironneau (2005) and Turinici (2009) when cubic splines are used to connect the volatility surface. Another reason for the smoothness assumption is due to the fact that this calibration problem is an underdetermined problem. A second-order regularization, which follows from the smoothness assumptions, adds more constraints to the problem than a first-order regularization. Note we are not trying to obtain a volatility surface as smooth as possible either.

There have been some theoretical studies about the stability, uniqueness and convergence of the calibration problem such as Bouchouev and Isakov (1997, 1999), Navon (1998), Jiang and Tao (2001), Crepey (2003), Egger and Engl (2005) and Hein (2005). However, according to the authors' knowledge, there is no conclusive answer as yet. We will not address these issues in this paper. We will also ignore the shortcomings of the local volatility model in describing the dynamics of the volatility surface. Assuming the local volatility model is perfect, we demonstrate the robustness of our calibration approach by extensive numerical tests with both theoretical local volatility models with known analytical solutions for European option prices and the real market option prices. The novelty of the present paper consists in the use of second-order Tikhonov regularization and the way we choose the Tikhonov regularization parameter. (See also Cordier *et al.* (2010) and Alekseev and Navon (2001)).

This paper is organized in the following manner. Section 1 consists of the introduction. In section 2, the mathematical formulation of the calibration problem is set up and complex issues related to the inverse problem are addressed. In section 3, we address the issue of using automatic differentiation tools to derive the adjoint code required to compute the gradient of the cost function with respect to the volatility surface. In section 4, by analysing how ill-posedness occurs for linear inverse problems, we propose a method to select the Tikhonov regularization parameter. In section 5, numerical results are presented and discussed. Finally, the paper concludes with a summary and conclusions section.

2. Description of the calibration problem

For consistency, the local volatility model is defined as in Lagnado and Osher (1997). The local volatility model assumes that the price s of an underlying follows a general diffusion process:

$$\frac{ds}{s} = (r - q)dt + \sigma(s, t)dW_t \qquad (1)$$

where r is the risk-free continuously compounded interest rate; q is the continuous dividend yield of the asset; W_t is a standard Brownian motion process, and the local volatility σ is a deterministic function that may depend on both the asset price s and the time t. r and q are assumed to be constant in this paper. Let $V(s_0, 0, K, T, \sigma)$ denote the theoretical price of an European option with strike K and maturity T at reference time 0 for an asset with spot price s_0 following the process in (1). Let $T_1, \ldots, T_N$ be the set of maturities of the European options available in the market for the asset. For each maturity T_i, the strikes available range from $K_{i1}, \ldots, K_{iM_i}$.

The calibration of the local volatility surface to the market is to find a local volatility surface $\sigma(s, t)$ such that the theoretical option price computed using this volatility surface comprises between the corresponding bid and ask prices for any option(K_{ij}, T_i), i.e.

$$V_{ij}^b \le V(s_0, 0, K_{ij}, T_j, \sigma) \le V_{ij}^a$$

for $i = 1, \ldots, N$ and $j = 1, \ldots, M_i$. V_{ij}^a and V_{ij}^b denote the bid and ask prices respectively for an option with maturity T_i and strike K_{ij} at the time $t = 0$.

This problem is usually solved by solving the following optimization problem:

$$\min_{0 < \sigma \le 1} \quad G(\sigma) = \sum_{i=1}^{N} \sum_{j=1}^{M_i} [(V(s_0, 0, K_{ij}, T_i, \sigma) - \overline{V}_{ij})w_{ij}]^2 \qquad (2)$$

where $\overline{V}_{ij} = (V_{ij}^b + V_{ij}^a)/2$ is the mean of the bid and ask prices. w_{ij} is a scaling factor to reflect the relative importance of different options. This scaling factor could play an important role in the calibration problem especially when option prices contain 'noises' since a small amount of relative noises from deep-in-the-money option prices could easily mask the 'signals' implied by out-of-money option prices. One way of choosing the above weights is based on the liquidity argument. The weights are computed as inversely proportional to the square of the bid-ask spreads to give more weights to the liquid options. Cont and Tankov (2004) suggested computing the weights as the inverse of square of the Black-Scholes vegas evaluated at the implied volatilities of the market option prices. They showed that it is approximately equivalent to minimizing the differences of the Black–Scholes implied volatilities between the market prices and the model computed prices. The weighting scheme is derived as an efficient way of generating approximation errors proportional to the bid-and-ask spreads and it also works for the case when the bid-and-ask spreads are not available. In this paper, we set w_{ij} to be one for all cases except for one case where we deliberately add enough artificial noises to show the significance of the weighting scheme, when the weighting scheme in Cont and Tankov (2004) is used. The reason we set $w_{ij} = 1$ is that either we know the prices are true prices or $w_{ij} = 1$ was used in other papers for the same market data used in the present paper. We retain the same weighting scheme in order to be able to compare our result with the previous studies.

When there is more than one maturity, the theoretical option price V can be efficiently computed by solving the Dupire equation (3). The Dupire equation establishes the option prices as a function of strike K and maturity T for a fixed asset price s_0 at reference time $t = 0$. By solving the following Dupire

equation (3) just once, we can obtain the theoretical prices for all the European options of the same underlying at s_0.

$$\frac{\partial V}{\partial T} - \frac{1}{2}K^2\sigma^2(K, T)\frac{\partial^2 V}{\partial^2 K} + (r - q)K\frac{\partial V}{\partial K} + qV = 0 \quad (3)$$

Notice in (3), σ is a function of K and T instead of s and t. We just point out that the function form of σ is not changed, and that the K, T, s, or t are all just dummy variables, details of which are in Dupire (1994).

Before attempting to solve the optimization problem in (2), we want to point out some aspects of the problem that make it complicated. The optimization problem in (2) is a large-scale non-linear underdetermined inverse problem. (a) The number of parameters to estimate is very large. To estimate the volatility surface, we want to find the volatility at each grid point. While similar to other archival material as well as our research found, only the section of volatility surface near the money can be estimated from market prices, the number of parameters to estimate is still quite large. (b) The Dupire or Black–Scholes equation is a non-linear operator in σ or σ^2. (c) The total number of options available is usually much less than the number of parameters to be estimated. Thus, it is also an under-determined problem. (d) As for most inverse problems, it is ill-posed in the sense that small changes in the option prices may lead to big changes in the volatility surface. When noises are included in option prices, which is usually the case in reality, the reconstructed volatility surface tends to be unstable. To resolve the issues of (c) and (d), we propose use of a second-order Tikhonov regularization, details of which will be introduced in later sections.

To deal with the issues of (a) and (b), a gradient-based optimization routine is usually used. Most papers Bouchouev and Isakov (1997, 1999), Jiang and Tao (2001), Jiang *et al.* (2003), Crepey (2003), Egger and Engl (2005), Hein (2005), Achdou and Pironneau (2005) and Turinici (2009) derived the gradient of cost function G in (2) with respect to σ by solving the adjoint model of the Dupire model. By using an adjoint approach, the gradient can be computed by solving the adjoint model just once. In all of these papers, the adjoint model of the Dupire model was derived first and then solved numerically. This way of using the adjoint belongs to the *differentiate-then-discretize* approach, i.e. one differentiates the partial differential equations (along with initial and boundary conditions), takes the adjoint of the results and then discretizes the continuous system of adjoint equations.

There is an alternative way of deriving the adjoint, namely the *discretize-then-differentiate* approach, see for example Giering (2000), in which one first discretizes the original model and then obtains a system of adjoint equations of the discretized model. Both approaches yield a set of discrete equations for the adjoint variables. But, the discretization and differentiation steps do not commute. Gunzburger (2000) found that the gradient derived using the *differentiate-then-discretize* approach can be inconsistent with the true gradient. The inconsistency can result in a serious difficulty for minimizing the cost function. In this paper, we will adopt the *discretize-then-differentiate* approach: we first discretize the Dupire model using a finite difference method and then differentiate the discrete version of

the Dupire model to obtain its adjoint model. In the step of differentiation of the discrete Dupire model, automatic differentiation in reverse mode can be utilized to generate the discrete adjoint model. In the following section, we set up the derivation of the gradient in a general framework so that the same technique can be used for calibration of other models or with respect to exotic options.

3. Gradient of the cost function

Algorithmic differentiation has already been utilized in the quantitative finance field. For example, Giles and Glasserman (2005) and Capriotti and Giles (2010) used it to speed up the calculation of Greeks. It has long been established in other studies such as computational fluid dynamics that the gradient of a cost function in the form of (2) can also be computed by using automatic differentiation, such as Giering and Kaminski (1998). We will just list some results for the completeness of this paper. For a more general formulation, see Castaings *et al.* (2007) and Navon (1998).

Let M be a general model such that

$$\frac{\partial X}{\partial t} = M(X, \alpha) \quad (4)$$

where $X \in \mathbf{R}^m$ is a vector containing the state variables of the model $\alpha \in \mathbf{R}^n$ denotes the model parameters. A typical cost function in parameter calibration assumes the form

$$J(X, \alpha) = \frac{1}{2}\int_{t_0}^{t_\tau}\langle W(X - X^{obs}), W(X - X^{obs})\rangle \, dt \quad (5)$$

where $[t_0, t_\tau]$ is the observation window and W is a weighting factor to reflect the relative importance of each observation. X^{obs} is the observation vector. It can be shown (Lions 1968) that the gradient is given by

$$\nabla_\alpha J = \int_{t_0}^{t_\tau}\left(-\left[\frac{\partial M}{\partial \alpha}\right]^T P\right) dt \quad (6)$$

where $P \in \mathbf{R}^m$ is adjoint variable of the state variables and is governed by the following system:

$$\begin{cases} \dfrac{\partial P}{\partial t} + \left[\dfrac{\partial M}{\partial X}\right]^T P = W(X - X^{obs}) \\ P(t_\tau) = 0 \end{cases} \quad (7)$$

where $\left[\frac{\partial M}{\partial X}\right]^T$ and $\left[\frac{\partial M}{\partial \alpha}\right]^T$ represent the transpose of the Jacobian matrix of the model with respect to state variables and model parameters respectively in the discrete case. When P is known by integrating backward in time the system described by (7), all the components of the gradient J with respect to α can be computed using equation (6).

Equations (6) and (7) show that we can compute the gradient of cost function $J(\alpha)$ by running the adjoint model only once. Griewank (1989) shows that the required numerical operations will require only 2–5 times the computation required for the forward cost function.

In this paper, $\left[\frac{\partial M}{\partial X}\right]^T$ and $\left[\frac{\partial M}{\partial \alpha}\right]^T$ are obtained using automatic differentiation tools. A complete detailed discussion of

the rationale of automatic differentiation is beyond the scope of this paper. See Griewank and Walther (2008) for details.†

4. Tikhonov regularization

4.1. Second-order Tikhonov regularization

To deal with the ill-posedness of the calibration problem, regularization is usually required. Tikhonov regularization is one of the most popular regularization methods for ill-posed inverse problems. In addition to minimizing the cost function, it seeks to minimize some measure of the solution, for example, the size of the solution or the norm of the first and second derivative of the solution. It usually assumes the following form.

$$J(\sigma) = G(\sigma) + \lambda \, \|\mathbf{L}\sigma\|_2^2 \qquad (8)$$

where $G(\sigma)$ is as defined in (2) and λ is the regularization parameter. $\mathbf{L}$ is an operator on σ. When $\mathbf{L}$ is the identity matrix, it is called the zeroth-order Tikhonov regularization. When $\mathbf{L}$ is an operator approximating the first or second derivative of σ with respect to s and t, it is called the first- or second-order Tikhonov regularization respectively. As mentioned in the introduction, most papers on the calibration of local volatility surfaces used the following first order Tikhonov regularization, see Lagnado and Osher (1997), Bouchouev and Isakov (1997, 1999), Bodurtha,Jr. and Jermakyan (1999), Jiang and Tao (2001), Jiang *et al.* (2003), Crepey (2003), Egger and Engl (2005), Hein (2005), Achdou and Pironneau (2005) and Turinici (2009).

$$J(\sigma) = G(\sigma) + \lambda \left(\left\| \frac{\partial \sigma}{\partial s} \right\|_2^2 + \left\| \frac{\partial \sigma}{\partial t} \right\|_2^2 \right) \qquad (9)$$

However, the volatility surface generated by the first-order Tikhonov regularization is usually rough. Assuming the volatility surface is smooth, we propose to use a second-order Tikhonov regularization: the regularization term $\|\mathbf{L}\sigma\|_2^2$ would be a measure of the norm of the second derivatives of σ with respect to s and t. Since the calibration problem is underdetermined, a second-order Tikhonov regularization also imposes more constraints on the calibration problem than the first-order Tikhonov regularization. Our regularization term $\|\mathbf{L}\sigma\|_2^2$ is an approximation of the following:

$$\left\| \frac{\partial^2 \sigma}{\partial s^2} \right\|_2^2 + \left\| \frac{\partial^2 \sigma}{\partial t^2} \right\|_2^2 + \left\| \frac{\partial^2 \sigma}{\partial t \partial s} \right\|_2^2 \qquad (10)$$

If σ were just a one-dimensional vector of size n, the exact form of $\mathbf{L}$ could be written as follows:

---

†There are several free automatic differentiation tools available, whose details are to be found on the website www.autodif.org. Automatic differentiation can help speed up the process of developing the numerical code of an adjoint model especially for complicated models. However, some debugging and verification is usually necessary for checking the validity of the code generated by the free automatic differentiation tools. For a method to verity the correctness of the adjoint code, please see the gradient test in Navon *et al.* (1992).

$$\begin{bmatrix} 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \cdots & & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \end{bmatrix}_{(n-2) \times n} \qquad (11)$$

At each grid point, the second derivative of σ is approximated by a second-order accurate finite difference scheme up to a constant. Due to the fact that σ is a two-dimensional surface rather than a vector, the explicit matrix form of operator $\mathbf{L}$ could not be easily written down since we have to approximate three-second derivatives at each grid point. For our computation, we actually do not need the explicit form of $\mathbf{L}$, since we just need the term $\| \mathbf{L}\sigma \|_2^2$. The following simple algorithm (1) describes the computation of $\| \mathbf{L}\sigma \|_2^2$ and the update of gradient of J with respect to σ when the regularization part is added. In the algorithm (1), $\partial^2\sigma/\partial s^2$, $\partial^2\sigma/\partial t^2$, $\partial^2\sigma/\partial t\partial s$ are all approximated by a second-order accurate finite difference scheme up to a constant.

Algorithm 1 Compute $\lambda \left(\left\| \frac{\partial^2 \sigma}{\partial s^2} \right\|_2^2 + \left\| \frac{\partial^2 \sigma}{\partial t^2} \right\|_2^2 + \left\| \frac{\partial^2 \sigma}{\partial t \partial s} \right\|_2^2 \right)$ and update gradient

---

// Compute $\left\| \frac{\partial^2 \sigma}{\partial s^2} \right\|^2$
$norm1 = 0.0$
for $i = 1$ **to** n_t **do**
  for $j = a + 1$ **to** $b - 1$ **do**
    $temp = \sigma_{j+1,i} + \sigma_{j-1,i} - 2\sigma_{j,i}$
    $\text{norm1} = \text{norm1} + temp * temp$
    $g(j+1, i) = g(j+1, i) + temp * \lambda$
    $g(j-1, i) = g(j-1, i) + temp * \lambda$
    $g(j, i) = g(j, i) - 2 * temp * \lambda$
  end for
end for
// Compute $\left\| \frac{\partial^2 \sigma}{\partial t^2} \right\|^2$
$norm2 = 0.0$
for $i = a$ **to** b **do**
  for $j = 2$ **to** $n_t - 1$ **do**
    $temp = \sigma_{i,j+1} - 2\sigma_{i,j} + \sigma_{i,j-1}$
    $\text{norm2} = \text{norm2} + temp * temp$
    $g(i, j+1) = g(i, j+1) + temp * \lambda$
    $g(i, j) = g(i, j) - 2.0 * temp * \lambda$
    $g(i, j-1) = g(i, j-1) + temp * \lambda$
  end for
end for
// Compute $\left\| \frac{\partial^2 \sigma}{\partial t \partial s} \right\|^2$
$norm3 = 0.0$
for $i = a$ **to** b **do**
  for $j = 2$ **to** $n_t - 1$ **do**
    $temp = \sigma_{i+1,j+1} + \sigma_{i-1,j-1} - \sigma_{i+1,j-1} - \sigma_{i-1,j+1}$
    $\text{norm3} = \text{norm3} + temp * temp$
    $g(i+1, j+1) = g(i+1, j+1) + temp * \lambda$
    $g(i-1, j+1) = g(i-1, j+1) - temp * \lambda$
    $g(i+1, j-1) = g(i+1, j-1) - temp * \lambda$
    $g(i-1, j-1) = g(i-1, j-1) + temp * \lambda$
  end for
end for
$\text{Lsigma} = \text{norm1} + \text{norm2} + \text{norm3}$
$f = f + \lambda * \text{Lsigma}$

Since only the section of the volatility surface that is near the money is sensitive to option prices and can be recovered, the regularization is just applied to the part of volatility surface $\sigma(s,t)$ for which the ratio between s and spot s_0 lies within the interval [0.8, 1.2].

For the regions of the volatility surface outside the interval defined above, no regularization is performed. Since the components of the gradient vector corresponding to volatilities at these regions are zero, the volatilities at these regions cannot be updated by a gradient based optimization routine and are thus kept constant throughout the optimization. The constant is the initial guess of the local volatility surface.

In the algorithm (1), $\sigma(n_x, n_t)$ is a two-dimensional matrix representing $\sigma(s,t)$ where n_x, n_t are the number of intervals along the s and t direction, respectively. a and b are the indices that correspond to $0.8s_0$ and $1.2s_0$ along the s direction, respectively. f and G are inputted respectively as the cost function and gradient before any regularization, and then returned as the regularized cost function and the gradient of the regularized cost function with respect to σ, respectively.

The calibration problem now assumes the form of a constrained minimization problem:

$$\min_{0<\sigma(s,t)\leq 1} \quad J(\sigma) = G(\sigma) + \lambda \parallel \mathbf{L}\sigma \parallel_2^2 \qquad (12)$$

4.2. Strategy for selecting the Tikhonov regularization parameter λ

A Tikhonov regularization solution of an inverse problem depends critically on a suitable selection of the regularization parameter λ. How to suitably choose a regularization parameter is still at the stage of active research. For linear inverse problems, λ is usually selected by either the L-curve method or by generalized cross validation theory, see Hansen (1998), Alekseev and Navon (2001), Aster *et al.* (2005) and Cordier *et al.* (2010). For non-linear inverse problems, the L-curve method is still applicable to select the optimal λ. The L-curve method plots the cost function $G(\sigma)$ with respect to $\parallel \mathbf{L}\sigma \parallel_2^2$. This plot usually assumes an L shape. The corner of the L-curve is considered the best compromise point between the size of the solution and the magnitude of the cost function. The λ at the corner of L-curve is considered the optimal regularization parameter λ. However, we found from our numerical tests that we cannot generate an L-shaped curve for this non-linear inverse problem since λ chosen close to any sort-of L-corner generates a volatility surface far away from the true volatility surface.

As many non-linear problems are solved iteratively by solving a linear problem at each iteration, we will adopt an iterative regularization strategy to solve the non-linear inverse problem, in which a suitable regularization parameter λ is selected at each iteration. By linearizing the problem at each iteration, some of the analysis for linear inverse problems can be applied.

To determine how to select a suitable regularization parameter at each iteration, we carry out the following analysis to see how ill-posedness occurs.

We are actually solving for a vector X from

$$\mathbf{F}X = \tilde{Y} \qquad (13)$$

where $\mathbf{F}$ is a non-linear model; X is the input of model $\mathbf{F}$; and $\tilde{Y}$ consists of observation data.

This problem cannot be solved directly due to its non-linearity. Instead, it is solved by minimizing a cost function of the form (5). Usually, a gradient-based optimization scheme is used to minimize the cost function in (5) iteratively. At each iteration, it attempts to find a better estimate X_{k+1} from the current estimate X_k using the gradient information of the cost function at X_k. This process is in fact equivalent to solving the following linearized problem:

$$\mathbf{F}X_{k+1} = \mathbf{F}X_k + \mathbf{A}(\delta X) + o(\delta X) = \tilde{Y} \qquad (14)$$

where $\mathbf{A}$ is the Jacobian matrix of non-linear operator $\mathbf{F}$ at X_k and δX represents the changes from X_k to X_{k+1}. X_{k+1} is then updated by $X_{k+1} = X_k + \delta X$. If equation (14) is not well posed, then the optimization routine is much likely to find an unstable solution. If equation (14) is well posed, the optimization routine has a better chance to find stable solutions.

Equation (14) can be reformulated as:

$$\mathbf{A}(\delta X) = \tilde{Y} - \mathbf{F}X_k \qquad (15)$$

Considering $X_{k+1} = X_k + \delta X$, (15) is equivalent to:

$$\mathbf{A}X_{k+1} = \tilde{Y} - \mathbf{F}X_k + \mathbf{A}X_k \qquad (16)$$

Let $B = \tilde{Y} - \mathbf{F}X_k + \mathbf{A}X_k$, then

$$\mathbf{A}X_{k+1} = B \qquad (17)$$

Let matrix $\mathbf{A}$ be an m by n matrix. In our case, n is the number of parameters to estimate, and m is the number of options. $\mathbf{A}$ can be reduced to the following form using Singular Value Decomposition(SVD).

$$\begin{aligned} \mathbf{A}_{mn} &= \mathbf{U}_{mm}\mathbf{S}_{mn}\mathbf{V}_{nn}^T \\ &= [\mathbf{U}_p, \mathbf{U}_0]\begin{bmatrix} \mathbf{S}_p & 0 \\ 0 & 0 \end{bmatrix}[\mathbf{V}_p, \mathbf{V}_0]^T \\ &= \mathbf{U}_p\mathbf{S}_p\mathbf{V}_p^T \end{aligned}$$

where p is the number of non-zero singular values s_i of matrix $\mathbf{A}$. Since m is less than n in our problem, $p \leq m$. $\mathbf{U}_{mm}$ and $\mathbf{V}_{nn}$ are orthogonal matrices. $\mathbf{S}_{mn}$ is a diagonal matrix. $\mathbf{U}_p$ and $\mathbf{V}_p$ are the first p columns of matrices $\mathbf{U}$ and $\mathbf{V}$ respectively. $\mathbf{S}_p$ is a diagonal matrix containing all the non-zero singular eigenvalues s_i. The singular values s_i are all positive and gradually decrease to zero.

The solution to (17) then can be written as in Aster *et al.* (2005):

$$\begin{aligned} X_{k+1} &= \mathbf{X}_\dagger + \tilde{X} = \mathbf{V}_p\mathbf{S}_p^{-1}\mathbf{U}_p^T B + \tilde{X} = \sum_{i=1}^{p} \frac{(\mathbf{U}_{.,i})^T B}{s_i}\mathbf{V}_{.,i} \\ &\quad + \sum_{i=p+1}^{n} \alpha_i \mathbf{V}_{.,\mathbf{i}} \qquad (18) \end{aligned}$$

where $\mathbf{V}_{.,\mathbf{i}}$ is the ith column of matrix $\mathbf{V}$.

From (18), we can see that the solution X_{k+1} is composed of two parts: $\mathbf{X}_\dagger$ and $\tilde{X}$. $\mathbf{X}_\dagger$ is the solution obtained from solving $\mathbf{U}_p\mathbf{S}_p\mathbf{V}_p^T X = B$ while $\tilde{X} = \sum_{i=p+1}^{n} \alpha_i \mathbf{V}_{.,\mathbf{i}}$ is a vector that lies in the null space of matrix $\mathbf{A}$. The existence of $\tilde{X}$ shows the underdetermined nature of this inverse problem.

For the solution $\mathbf{X}_\dagger = \Sigma_{i=1}^{p} (\mathbf{U}_{.,i})^T B/s_i \mathbf{V}_{.,i}$, if $(\mathbf{U}_{.,i})^T B$ does not decay as fast as s_i, X_{k+1} will become unstable as s_i

tends to zero, since a small amount of noise from $\boldsymbol{B}$ will be amplified by the small singular values.

After diagnosing where the ill-posedness originates from, we propose to regularize the ill-posedness by eliminating the effects of the small singular values s_i. The addition of Tikhonov regularization at each iteration is equivalent to solving the following overdetermined linear problem:

$$\begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{L} \end{bmatrix} X_{k+1} = \begin{bmatrix} \boldsymbol{B} \\ \mathbf{0} \end{bmatrix} \qquad (19)$$

When $\mathbf{L}$ represents a higher order Tikhonov regularization operator, as in our case, the analytical solution of (19) can be obtained by applying a generalized singular value decomposition(GSVD) of the matrix pair $[\mathbf{A}^T, \mathbf{L}^T]^T$, see Hansen (1998) and Aster *et al.* (2005) for details. But, GSVD of the matrix pair is computationally expensive especially since the dimension of our problem is large. In addition, most GSVD packages require the explicit form of matrices $\mathbf{A}$ and $\mathbf{L}$, neither of which is generated explicitly in our method. Extra computation and storage are necessary to generate and store the matrices $\mathbf{A}$ and $\mathbf{L}$ in order to use the GSVD packages. Furthermore, if we need to carry out a GSVD to find the regularization parameter λ at each iteration, the total computational cost of the minimization of (12) becomes very expensive.

To avoid using GSVD, we consider the special case when $\mathbf{L}$ is the identity matrix in order to gain insight of the problem. When $\mathbf{L}$ is the identity matrix, the regularized solution of (19) assumes the following analytical form of (20), see Hansen (1998) for the derivation.

$$X_\lambda = \sum_{i=1}^{m} \frac{s_i{}^2}{\lambda^2 + s_i{}^2} \frac{(\mathbf{U}_{.,i})^T \boldsymbol{B}}{s_i} \mathbf{V}_{.,i} \qquad (20)$$

When $\lambda \gg s_i$, the weighting factor $f = \frac{s_i{}^2}{\lambda^2 + s_i{}^2}$ is about 0. When $\lambda \ll s_i$, f is about 1. Choosing a λ that is smaller than the leading singular values and greater than the smallest singular values eliminates the ill-posedness caused by small singular values, yet does not affect the information represented by the large singular values. The under-determined part $\tilde{X}$ is also eliminated. Inspired by this insight, we choose our regularization parameter λ to be one of the singular values of $\mathbf{A}$ determined by the truncation level defined in the following:

$$\frac{\sum_{k=1}^{i} s_i}{\sum_{k=1}^{m} s_i} = \text{truncation level} = 50\% \qquad (21)$$

where the singular values are sorted in order of magnitude such as $s_1 \geq s_2 \geq \ldots \geq s_m \geq 0$, where m is the total number of singular values.

Now, the question is how to compute the singular values of $\mathbf{A}$ at each iteration? We need an algorithm that can compute the singular values without requiring the explicit form of matrix $\mathbf{A}$. For this purpose, we will use the package ARPACK which meets this requirement. All it requires is the product of matrices $\mathbf{A}$ and $\mathbf{A}^T$ with a vector. For our problem, the tangent linear code and adjoint code derived from the automatic differentiation tools readily compute these two products. ARPACK is based upon an algorithmic variant of the Arnoldi process called the Implicitly Restarted Arnoldi Method (IRAM). See Lehoucq *et al.* (1998) for details.

Computing the singular values at each iteration to determine λ is the most computationally expensive part of our algorithm.

From our numerical tests carried out in the following section, we found out that the λ selected in this manner does not change much throughout the minimization. By using a fixed λ, the reconstructed volatility surface remains the same as the one reconstructed by repeatedly updating λ at each iteration. But, the computational time is significantly reduced by using a fixed λ. If we assume that λ selected according to (21) is almost a constant during the minimization of (12), an alternative and efficient strategy of choosing λ consists in using a constant λ selected according to (21) throughout the minimization process of (12). This assumption is valid when the Jacobian matrix $\mathbf{A}$ does not change significantly during the minimization process. If we assume that the initial guess X_0 is not far away from the optimal solution X^*, then we can assume $\mathbf{A}$ is almost constant. In our calibration problem, the initial guess X_0 is set as a constant volatility surface obtained by averaging the Black–Scholes implied volatilities of the ATM options across different maturities. If we assume that the true local volatility surface does not deviate much from the average of the Black–Scholes implied volatility surface of ATM options, then the assumption that $\mathbf{A}$ is constant is reasonable. In this case, we can assume λ is constant. However, for a general model when X_k changes significantly across iterations, we have to choose a λ at each iteration. For this reason, we still present the pseudo-algorithm for the general case in algorithm (2) on the following page.

With the gradient obtained from the previous section and the regularization parameter λ ready, we can use a constrained optimization routine to find the optimal σ of (12). We use the algorithm L-BFGS-B to carry out the optimization. For details of L-BFGS-B, see Zhu *et al.* (1997). This is a robust algorithm for bound-constrained minimization. Prior to discussing our numerical tests, we summarize our pseudo-algorithm description in the following.

Algorithm 2 Main algorithm to reconstruct the local volatility surface

---

1. Initialize volatility surface $\sigma_0(s, t)$.
2. Use (3) to compute option prices V_{cmpt} and cost function G in (2).
3. Feed the difference between V_{cmpt} and V_{obs} into the adjoint model $\mathbf{A}^T$, using (6) and (7), to compute the gradient of G with respect to $\sigma(s, t)$.
4. Use ARPACK to compute the singular values of Jacobian matrix $\mathbf{A}$ and select the regularization parameter λ according to (21).
5. Compute the regularized cost function J of (12) and update the gradient after the regularization.
6. Insert the cost function J and its gradient into L-BFGS-B routine to obtain the next estimate $\sigma_{k+1}(s, t)$. $k = 0, 1, 2, \cdots$
7. When either the stopping criterion of L-BFGS-B is satisfied or the number of function calls of the cost function exceeds a preset limit, stop. Otherwise, go back to step 2.

For theoretical volatility models, the limit of the number of function calls is 1500 while for the case of real market data the limit is 250. We allow more iterations for the theoretical volatility models since the true volatility surfaces are known. The recovered volatility surface actually displays the general

features of the true volatility surface after 250 function calls, which is why we set the upper limit of function calls for the real market data as 250.

5. Numerical tests

For all of our numerical tests, the initial guess σ_0 is the average of Black–Scholes implied volatilities for the ATM options across different maturities. We scale the spot price of the underlying to 100 and then the option prices are scaled accordingly. The scaling reduces the calibration problem for different underlying instruments into the same problem. It has the additional benefit that λ can be precomputed and applied to different problems when we assume the regularization parameter λ is constant and r and q do not change significantly.

The Dupire equation (3) is solved using the backward Euler scheme in time and a centered finite difference scheme in space direction. The computation domain $\begin{bmatrix} 0 & \bar{T} \end{bmatrix} \times \begin{bmatrix} 0 & \bar{K} \end{bmatrix}$ is set as $\bar{K} = 2s_0$ as in Lagnado and Osher (1997) while $\bar{T}$ is the longest maturity. The space and time domain are divided into $n_x = 200$ and $n_t = 100$ intervals respectively. Since only the section of volatility surface $\sigma(s, t)$ for which the ratio s/s_0 lies in [0.8, 1.2] can be recovered, the total number of parameters to calibrate is $0.2 \times 200 \times 100 = 4000$. The lower and upper bound for σ is set to be 0.00001 and 1, respectively. We perform two kinds of numerical tests: one for volatility models, whose analytical solution for European options are known; and the other one for the real market options data.

5.1. Tests with theoretical volatility models

We start with the constant elasticity of variance (CEV) model, for which the analytical form of European option prices can be found in Cox and Ross (1976). The CEV model assumes the following form:

$$ds(t) = \mu s dt + \kappa s^p dW(t)$$

According to our definition of local volatility in (1), the local volatility for the CEV model is:

$$\sigma(s, t) = \kappa s^{p-1} \tag{22}$$

We will test three cases: $p = 0$, $p = \frac{1}{2}$ and $p = 2$. When $p = 0$, it corresponds to the Bachelier model. When $p = \frac{1}{2}$, it corresponds to the square root process. When $p = 2$, it is a special case of quadratic volatility model. The first case was used as a test case in both Lagnado and Osher (1997) and Coleman *et al.* (1999). Specifically, we test the following three cases:

$$\sigma(s, t) = \frac{15}{s} \tag{23}$$

$$\sigma(s, t) = \frac{2}{\sqrt{s}} \tag{24}$$

$$\sigma(s, t) = 0.002s \tag{25}$$

The constant κ in (22) is chosen in order that $\sigma(s, t)$ is contained in the interval of (0, 1) for all s and t. Twenty-two
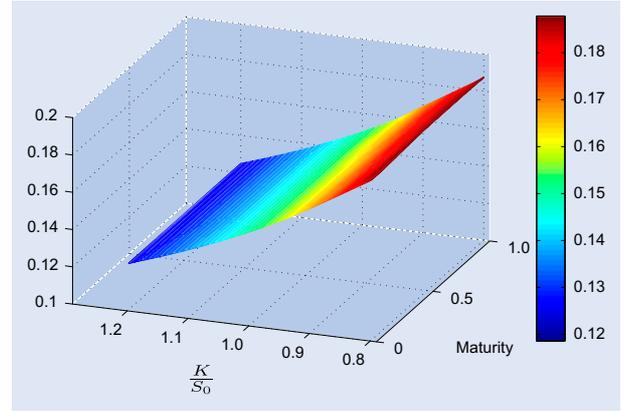
Figure 1. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s, t) = \frac{15}{s}$.
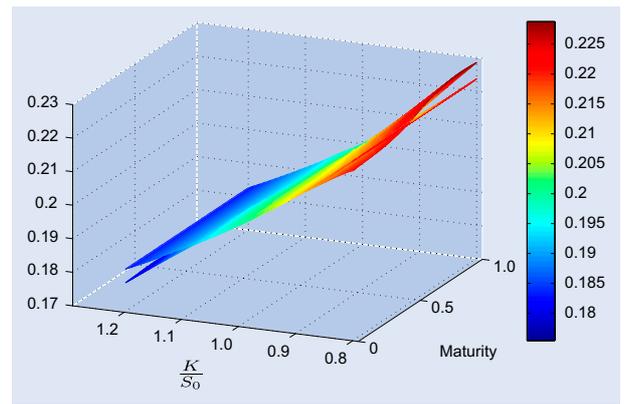
Figure 2. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s, t) = \frac{2}{\sqrt{s}}$.

European call option prices are generated using the closed-form solution for two maturities $T = 0.5$ and $T = 1.0$. For each maturity, we select eleven options whose strikes range from 90.0 to 110.0 with an increment of 2.0. These option prices are used to recover the volatility surface for (23)–(25). Similar to the study of Lagnado and Osher (1997) and Coleman *et al.* (1999), $s_0 = 100$, the risk free interest rate $r = 0.05$ and the dividend yield $q = 0.02$ for all three cases.

Figures 1–3 show the recovered volatility surface and the true volatility surface. For all the three cases, the recovered volatility surfaces approximate the true volatility surfaces very well. The relative errors of the computed option prices with respect to the true option prices are of the order of 10^{-4}. Figure 4 shows the plot of relative errors and option prices with respect of the number of options for the case (25). For the other two cases, the plots of relative errors exhibit similar patterns.

To compare the difference between the first-order Tikhonov regularization and the second-order Tikhonov regularization, Figures 5 and 6 show the recovered volatility surface by using the first-order Tikhonov regularization for two CEV models. We can see that even for these two simple CEV models, the first-order Tikhonov regularization could not match the true volatility surface as precisely as the second order Tikhonov regularization.

For all of the above CEV models, $\sigma(s, t)$ are monotonic functions of s. Next, we deliberately choose a quadratic
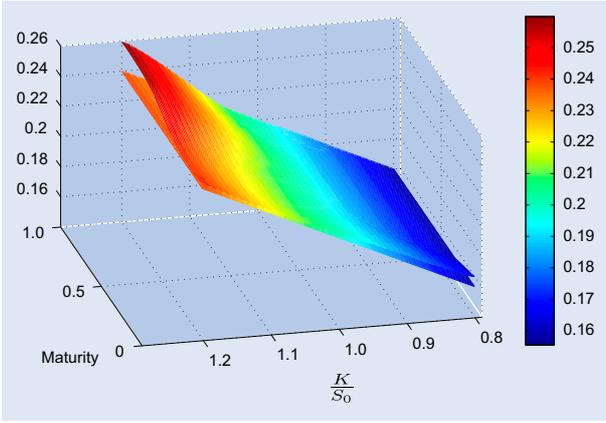
Figure 3. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s,t) = 0.002s$.
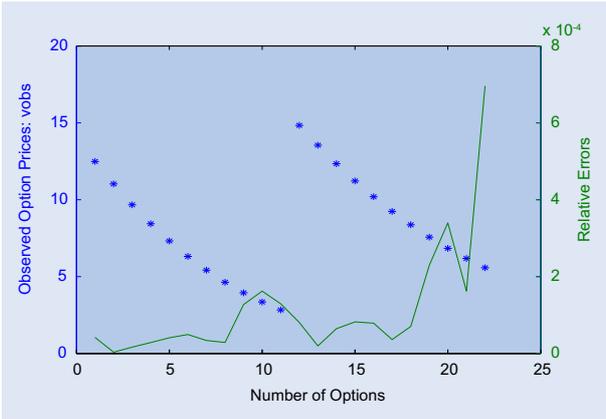
Figure 4. $*$: option prices; $\_$: relative errors = |vobs-vcmpt|/vobs. Left axis: the true option prices. Right axis: the relative errors of the computed option prices using optimal volatility surface with respect to the true prices for the volatility model $\sigma(s,t) = 0.002s$.
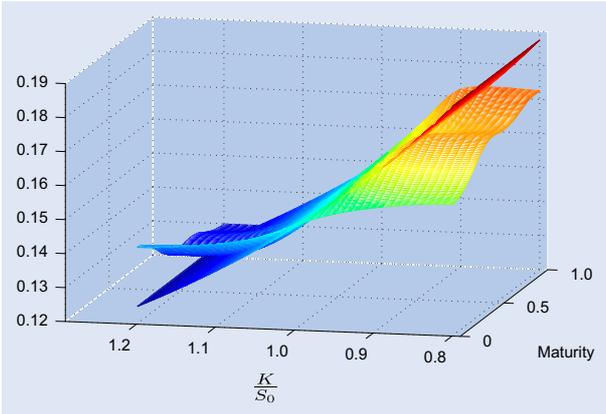
Figure 5. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s,t) = \frac{15}{s}$ using the first order Tikhonov regularization.

volatility model that is not monotonically changing as our test case. Andersen (2011) summarizes the analytical solution of European option prices for different quadratic volatility models. The following quadratic volatility model is taken from his paper.

$$\sigma(s,t) = 0.1 \left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s} \right) \qquad (26)$$

A total of 22 European put options with the same set of maturities and strikes as the previous tests are computed as
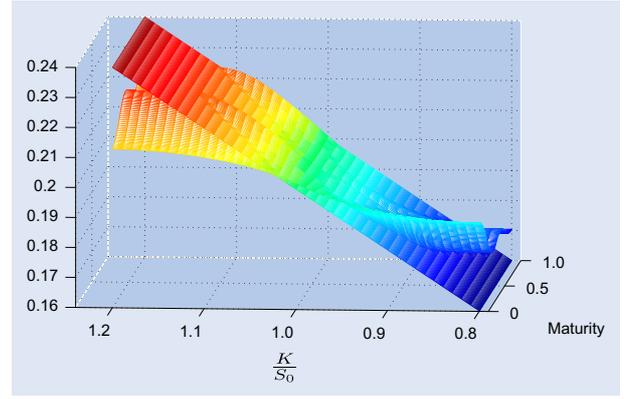
Figure 6. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s,t) = 0.002s$ using the first order Tikhonov regularization.
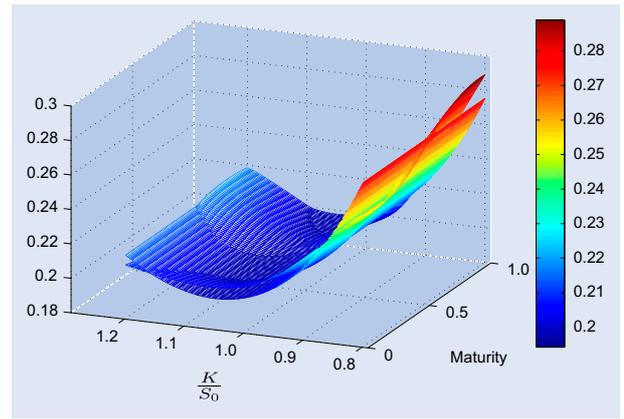
Figure 7. The true volatility surface and optimal volatility surface for the volatility model $\sigma(s,t) = 0.1 \left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s} \right)$.

market data. s_0 is set to 100 and the risk-free interest rate r and the dividend yield q are both zero. (When the drift is not zero, a change of measure can reduce the drift to zero). Figure 7 plots both the true volatility surface and the recovered volatility surface. We can see that the recovered volatility surface approximates the true volatility surface fairly well. Figure 8 displays the relative errors of computed option prices with respect to the true prices. We can see that the relative errors are of order of 10^{-4}. Figure 9 displays the cost function J with respect to iteration number. Figure 10 shows the decrease of the norm of the projected gradient as the number of iterations increases.

To test the stability of our methods, we add noises to the true option prices to assess whether we can still recover the volatility surface. The noise is introduced as in (Coleman *et al.* 1999):

$$\tilde{v}_i = v_i + 0.02\epsilon_i$$

where v_i is the true price of the ith option, ϵ_i is a uniformly distributed random number between 0 and 1. The noises are introduced as absolute errors rather than relative errors. The plot of the reconstructed volatility surfaces using noisy option prices and noise-free option prices is shown in Figure 11 for the quadratic volatility model. We can see that the two volatility surfaces are indistinguishable from each other with the maximum absolute difference of the order of 10^{-3}. It means that our method is stable with respect to a small amount of perturbation.
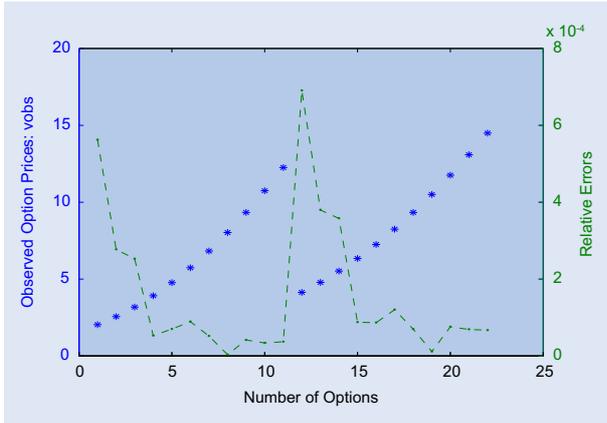
Figure 8. ∗ : option prices; _ : relative errors. Left axis: The true option prices. Right axis: The relative errors of the computed option prices using optimal volatility surface with respect to the true prices for the volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

Figure 9. The evolution of the regularized cost function versus the number of minimization iterations for the volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

Figure 10. Variation of the norm of the projected gradient of the cost function vs the number of iterations for the volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

Next, we test the case when the noises are introduced as relative errors:

$$\tilde{v}_i = v_i(1 + 0.02\epsilon_i)$$

Figure 11. The optimal volatility surfaces obtained from the non-noisy and noisy option prices(2% uniformly distributed absolute noise) for the volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

Figure 12. The true volatility surface and the optimal volatility surface obtained from the noisy option prices(2% uniformly distributed relative noise) for the volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

Figure 13. The regularization parameter λ computed at each iteration for volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.

2% of uniformly distributed noises are added as relative errors to the option prices. A direct calibration without any weighting of the noisy option prices fails to recover the true volatility surface. When relative errors are introduced, a proper weighting scheme needs to be assigned in order to reflect the relative
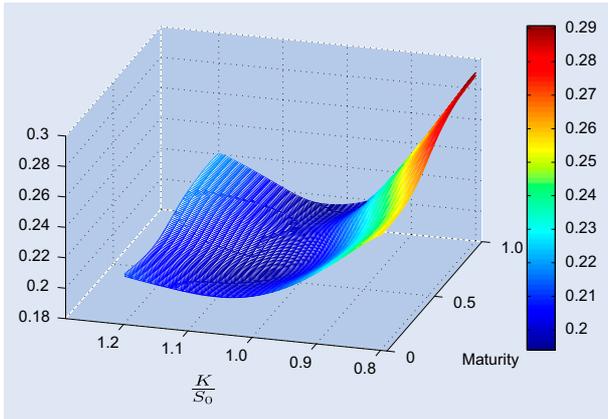
Figure 14. The optimal volatility surfaces obtained by using a constant $\lambda = 0.94$ and a λ updated at each iteration for volatility model $\sigma(s,t) = 0.1\left(1 + \frac{s_0}{s} + \frac{(s-s_0)^2}{100s}\right)$.
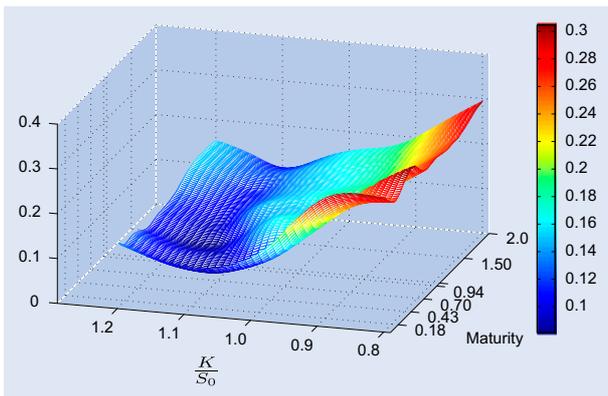
Figure 15. The optimal volatility surface obtained for S&P 500 index European call options in October 1995. $s_0 = \$ 590$, r=0.06, q=0.0262. Note: the available maturities are plotted on the T axis in unit of years.

importance of different options. We adopt the weighting method as in Cont and Tankov (2004) to scale the noisy prices in this case, which is defined as:

$$w_i = \frac{1}{vega(I_i)^2}$$

where I_i is the Black–Scholes implied volatility of the ith noisy option price, $vega()$ is the Black–Scholes vega evaluated as a function of implied volatility. Figure 12 displays the recovered the volatility surface with respect to the true volatility surface. We can see that the recovered volatility surface approximates the true volatility surface very well.

The total CPU time for each of the previous six tests lies between 332 and 480 s using a Dell Vostro 1720 with Intel Core Duo CPU @2.2G HZ and 2GB RAM.

The above calibration updates the regularization parameter λ at each iteration. Figure 13 displays λ against the number of iterations for the quadratic volatility model. At the beginning, λ is set to zero. We can see that λ does not vary much throughout iterations and that it almost stays constant after a number of iterations. The same phenomenon is observed for other test cases as well as the real market data cases in the following section. Based on this observation, we use as a constant λ during the optimization. Figure 14 shows the recovered volatility
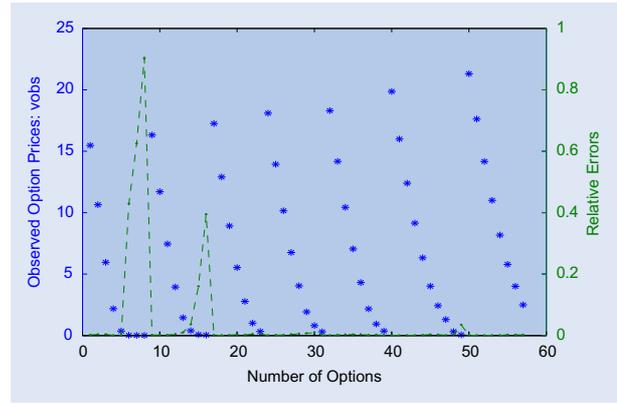
Figure 16. $*$: scaled option prices; $\_$: relative errors. Left axis: the scaled prices of S&P 500 index European call options in October 1995 (Andersen and Brotherton-Ratcliffe 1998). Right axis: the relative errors of computed option prices with respect to observed price. Option prices are plotted in an order of increasing maturities.

surface by using a constant λ vs an updated λ for the quadratic volatility model with noise free prices. The two constructed volatility surfaces are indistinguishable from each other with the maximum absolute difference being of the order of 10^{-3}. By using a constant λ the total CPU time for each of the previous six tests is now just between 13 and 20 s.

5.2. Tests with market data

Our market data are all obtained from previous studies on the calibration of local volatility surface. Our first test uses option prices as in Coleman *et al.* (1999), Andersen and Brotherton-Ratcliffe (1998) and Turinici (2009). The options are European call options on S&P 500 index in October 1995. There are a total of 57 options with seven maturities. The initial index, interest rate, and dividend yield are provided in the footnotes of Figure 15. Figure 15 shows the optimal volatility obtained. Contrary to previous studies, the volatility surface has an obvious skew structure as expected for the equity market. The volatility surface is also smoother. Furthermore, the recovered volatility surface is in a range between 0.08 and 0.30 without local extreme values. The relative errors of computed prices with respect to observed prices are plotted in Figure 16. The relative errors are mostly close to zero except for options whose prices are close to zero. This is acceptable since the bid and ask spreads for out of money options are usually much higher than or comparable with the option prices. In other words, out of money option prices allow a much higher degree of approximation errors. The mean absolute relative error is 4.7%. Excluding the seven options with big absolute relative errors, the mean absolute relative error is as small as 0.2%.

The second test uses data-set from Andreasen and Huge (2011), which contained 155 European options on the Eurostoxx(SX5E) index spanning 12 maturities. The shortest maturity was about one week ($T = 0.025$) and the longest maturity was about 5.8 years ($T = 5.778$). Since the original data only had market data in terms of implied volatilities without the interest rate structure, we computed the option prices just using these implied volatilities under the assumption the interest rates were zero. This assumption is reasonable since the local
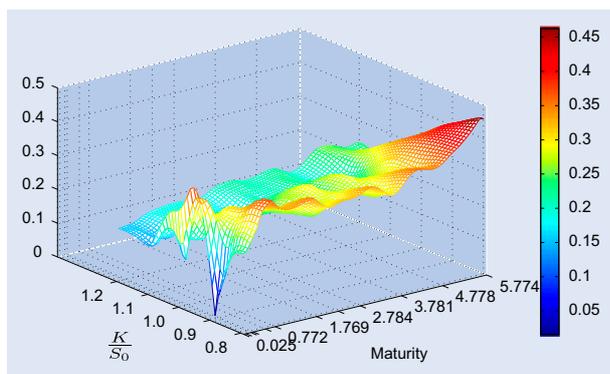
Figure 17.   The optimal volatility surface obtained for Eurostoxx 50(SX5E) equity options on March 1, 2010, as studied in Andreasen and Huge (2011). Note: the available maturities are plotted on the T axis in unit of years.

Figure 18.   ∗ : option prices in terms of implied volatilities; _ : absolute difference of the implied volatilities between market data and reconstructed option prices. Left axis: the absolute difference of implied volatilities between market data and reconstructed data. Right axis: the option prices for SX5E in terms of implied volatilities on March 1, 2010.
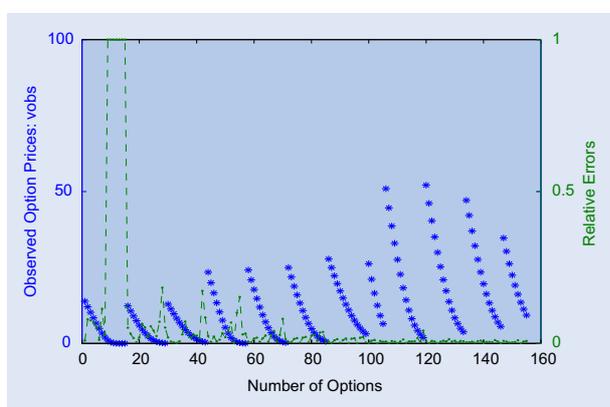
Figure 19.   ∗ : scaled option prices ; _ : relative error between market data and reconstructed option prices. Left axis: the scaled option prices for SX5E on March 1, 2010. Right axis: the relative error between market data and reconstructed data. Option prices are plotted in an order of increasing maturities.

volatility model (1) can be changed into a driftless process by a change a measure while the local volatility term keep the same during the change. The recovered volatility surface is shown in Figure 17. Figure 18 shows the absolute errors and the option

Figure 20.   The optimal volatility surface obtained when $nt = 500$ for Eurostoxx 50(SX5E) equity options on March 1, 2010, as studied in Andreasen and Huge (2011). Note: the available maturities are plotted on the T axis in unit of years.

prices in terms of implied volatilities as in Andreasen and Huge (2011). Figure 19 plots the relative errors and the option prices in terms of prices. Figure 17 displays an obvious skew structure although there is a lot of fluctuations of the local volatility surface close to the region when $T = 0.025$. The computed data do not match the market data very well either for that maturity. However, one of the possible reasons is that our finite difference scheme does not have enough resolution at $T = 0.025$. By setting $\bar{T} = 5.778$, $nt = 100$ and using a uniform grid, $\Delta t = 0.058 > 0.025$. A finer mesh grid should be used to improve the accuracy of the computed the option prices at this maturity. Ignoring the 15 options with maturity $T = 0.025$, Figures 18 and 19 demonstrate a very good fit of the market prices. Again, high relative errors occur when the option prices are close to zero (Figure 19). For the remaining 140 options, the mean relative error is 2% and the mean absolute difference in terms of implied volatility is 0.6%. Next, we refine the grids by setting $nt = 500$. Figure 20 shows the recovered volatility surface. Compared to Figure 17, the volatility surface does not change significantly. Anyway, the region when $T \leq 0.025$ just occupies a small section of the volatility surface. Figure 21 exhibits the absolute difference in terms of implied volatility. Compared to Figure 18, we can see there are some improvements in terms of matching the prices for the options with $T = 0.025$. The CPU time in this case is very high, 32 minutes when λ is selected iteratively. A non-uniform grid should be used to reduce the computational cost when it is necessary to resolve cases like this with the maximum maturity $\bar{T}$ large yet the soonest maturity being very small.

The last example is for European call options in the foreign exchange market. The option data were studied by both Avellaneda *etal.* (1997) and Turinici (2009). There are 15 European call options for the US dollar/Deutsche mark with five maturities, which are computed from 20, 25 and 50 delta risk-reversals quoted on 23 Aug 1995. The spot price and interest rates are shown in the captions of Figure 22. The optimal volatility surface and relative errors are plotted in Figures 22 and 23 respectively. The volatility surface has a shape similar to the smile shape as expected for volatilities in the foreign exchange market. The mean absolute relative error is as small as 1.9%.
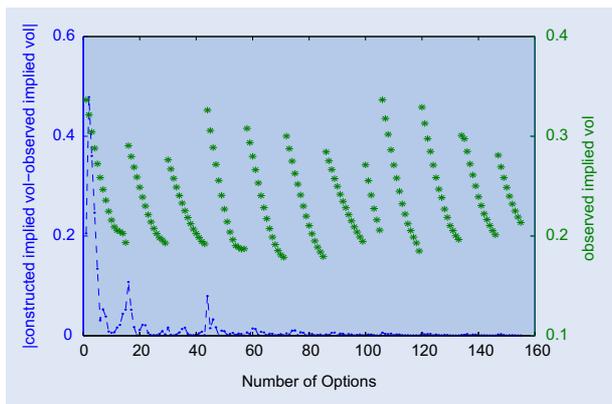
Figure 21. $*$: option prices in terms of implied volatilities; $\_$: absolute difference of the implied volatilities between market data and reconstructed option prices. Left axis: the absolute difference of implied volatilities between market data and reconstructed data when nt=500. Right axis: the option prices for SX5E in terms of implied volatilities on March 1, 2010. Option prices are plotted in an order of increasing maturities.
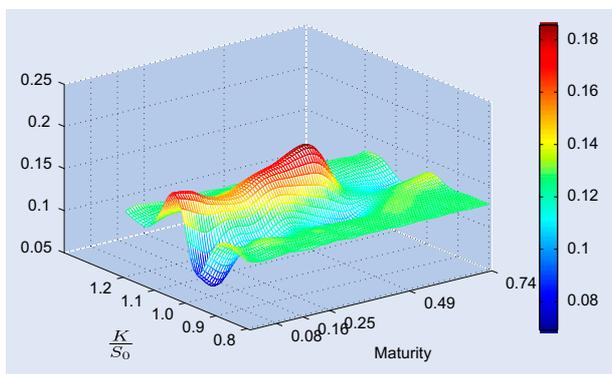
Figure 22. The optimal volatility surface obtained for European call options of US dollar/ Deutsche mark rate. The spot price was s_0 =1.48875; US dollar interest rate was r_{USdollar} = 5.91%; Deutsche mark rate was $r_{\text{Deutschemark}}$= 4.27%. Note: the available maturities are plotted on the T axis in unit of years.
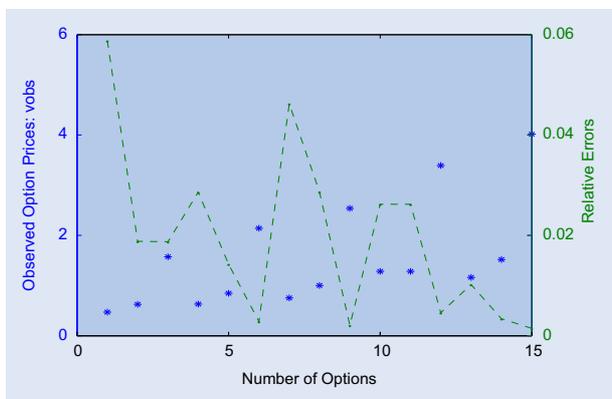
Figure 23. $*$: scaled option prices; $\_$: relative errors. Left axis: the scaled prices of European call options on US dollar/Deutsche mark rate recovered from 20, 25, and 50 delta risk reversals (Avellaneda *etal.* 1997). Right axis: the relative errors of computed option price with respect to observed price. Option prices are plotted in an order of increasing maturities.

There may still be some instability in the volatility surface recovered, for example the reconstructed volatility surface for

the last example. We attribute this partially to the assumption that every option is equally important. The amount of noises in the market option prices is unknown. A proper weighting scheme is necessary to reflect the relative importance of different options. This will constitute an interesting follow-up future research area.

For the above three numerical tests, the CPU time is 158, 232, and 12 s respectively, using a Dell Vostro 1720 with Intel Core Duo CPU @2.2G HZ and 2GB RAM. Again, when we use a constant λ, the CPU time is just as small as 3.4, 3.6 and 3.6 s respectively. The changes of the relative errors and recovered implied volatility surface are again very small compared to those obtained using an updated λ. From here, we can see that when using a constant λ, the CPU time is independent of the number of the options. When $nt = 500$, the CPU time for the data set from Andreasen and Huge (2011) is 18.9 seconds. From this example, we can see when using a constant λ the CPU time grows linearly as the number of parameter increases, which results from the linear dependence of computational cost of a adjoint model on the number of parameters, as mentioned by Giles and Glasserman (2005).

The only parameter that is subject to change in our algorithm is the *truncation level*. It is fixed at 50% throughout our numerical tests. Other truncation levels were also tested. The relative error and the general shape of the optimal volatility surface did not change significantly overall when the truncation level was less than 0.9 although as the truncation level gets lower the volatility surface tends to be smoother. This means this method is fairly robust for different choices of truncation levels as long as the regularization parameter selected is not close to the smallest singular values at the end of the singular values spectrum. The fact that we used the same truncation level for all numerical tests also serves as an indication that the calibration is not very sensitive to the truncation levels.

6. Summary and conclusions

Our present research addresses solving the calibration of the local volatility surface for European options in a non-parametric approach by using a second-order Tikhonov regularization. We select one of the singular values of the Jacobian matrix of the Dupire model as the regularization parameter. For the theoretical volatility models with known analytical solution for European option prices, the proposed method recovers almost exactly the true volatility surface. This method was also tested and proves to be stable for a small amount of noises in the option prices. We also show the significance of the weighting of option prices when the option prices contain a significant amount of noises.

This method also performs reasonably well for real market data. The observed option prices can be matched very well. The obtained volatility surface lies in a reasonable range with nice general pattern, for example, the skew structure in the equity market and the smile structure in the foreign exchange market. Some instability may still persists in the volatility surface recovered. We attribute this partially to the noises in market data and our assumption that every option is equally important in the market data. A proper weighting scheme may prove to be necessary to reflect the relative importance of

different options when they exhibit different amount of noises. When using a constant regularization parameter, the total CPU time is as small as 3–4 seconds for market data.

Last, although this paper focuses on calibration for local volatility model for European options, the calibration technique proposed here is developed in a very general framework so that it can be generalized to explore the calibration of other models such as the hybrid local-stochastic volatility models or calibration with respect to other options, such as American options.

Acknowledgements

References

Achdou, Y. and Pironneau, O., *Computational Methods for Option Pricing*, 2005 (Society for Industrial and Applied Mathematics (SIAM): Philadelphia).

Alekseev, A.K. and Navon, I.M., The analysis of an ill-posed problem using multi-scale resolution and second-order adjoint techniques. *Comput. Methods Appl. Mech. Eng.*, 2001, **190**, 1937–1953.

Andersen, L. and Brotherton-Ratcliffe, R., The equity option volatility smile: an implicit finite difference approach. *J. Comput. Finan.*, 1998, **1**, 37–64.

Andersen, L., Option pricing with quadratic volatility: A revisit. *Finance Stochast.*, 2011, **15**, 191–219.

Andreasen, J. and Huge, B., Volatility interpolation. *Risk*, 2011, **24**, 76–79.

Aster, R.C., Borchers, B. and Thurber, C., *Parameter Estimation and Inverse Problems*, 2005 (Elsevier Academic Press: Burlington).

Avellaneda, M., Fridman, C., Holems, R. and Samperi, D., Calibrating volatility surface via relative entropy minimization. *Appl. Math. Finan.*, 1997, **4**, 667–686.

Black, F. and Scholes, M., The pricing of options and corporate liabilities. *J. Polit. Econ.*, 1973, **81**, 637–659.

Bodurtha, J.N. Jr and Jermakyan, M., Nonparametric estimation of an implied volatility surface. *J. Comput. Finan.*, 1999, **2**, 29–61.

Bouchouev, I. and Isakov, V., The inverse problem of option pricing. *Inv. Problems*, 1997, **13**, L11–L17.

Bouchouev, I. and Isakov, V., Uniqueness, stability and numerical methods for the inverse problem that arises in financial markets. *Inv. Problems*, 1999, **15**, R95–116.

Capriotti, L. and Giles, M., Fast correlation Greeks by adjoint algorithmic differentiation. *Risk*, 2010, **23**, 79–83.

Castaings, W., Dartus, D., Le Dimet, F.X. and Saulnier, G.M., Sensitivity analysis and parameter estimation for the distributed modeling of infiltration excess overland flow. *Hydrology Earth Syst. Sci. Discuss.*, 2007, **4**, 363–405.

Coleman, T.F., Li, Y. and Verma, A., Reconstructing the unknown local volatility function. *J. Comput. Fin.*, 1999, **2**, 77–100.

Constable, C., Parker, L. and Constable, G.C., Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data. *Geophysics*, 1987, **52**, 289–300.

Cont, R. and Tankov, P., Non-parametric calibration of jump-diffusion option pricing models. *J. Comput. Finan.*, 2004, **7**, 1–49.

Cordier, L., Abou El Majd, B. and Favier, J., Calibration of POD reduced-order models using Tikhonov regularization. *Int. J. Numer. Methods Fluids*, 2010, **63**, 269–296.

Cox, J.C. and Ross, S.A., The valuation of options for alternative stochastic processes. *J. Finan. Econ.*, 1976, **3**, 145–166.

Crepey, S., Calibration of the local volatility in a generalized Black–Scholes model using Tikhonov regularization. *J. Math. Anal. SIAM*, 2003, **34**, 1183–1206.

Dupire, B., Pricing with a smile. *Risk*, 1994, **7**, 18–20.

Egger, H. and Engl, H.W., Tikhonov regularization applied to the inverse problem of option pricing: Convergence analysis and rates. *Inv. Problems*, 2005, **21**, 1027–1045.

Giering, R., Tangent linear and adjoint biogeochemical models. In *Inverse Methods in Global Biogeochemical Cycles*, edited by P. Kasibhatla, M. Heimann, P. Rayner, N. Mahowald, R.G. Prinn and D.E. Hartley, pp. 33–47, 2000 (American Geophysical Union: Washington DC).

Giering, R. and Kaminski, T., Recipes for adjoint code construction. *ACM Trans. Math. Softw.*, 1998, **24**, 437–474.

Giles, M. and Glasserman, P., Smoking adjoints: Fast calculation of Greeks in Monte Carlo calculation. Technical Report, 2005, NA-05/15.

Griewank, A., On automatic differentiation. In *Mathematical Programming: Recent Developments and Applications* edited by M. Iri and K. Tanabe, pp. 83–108, 1989 (Kluwer Academic Publishers: Dordrecht).

Griewank, A. and Walther, A., *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, 2nd ed., 2008 (Society for Industrial and Applied Mathematics (SIAM): Philadelphia).

Gunzburger, M., Adjoint equation-based methods for control problems in incompressible, viscous flows. *Flow Turbulence Combust.*, 2000, **65**, 249–272.

Hansen, P.C., *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*, 1998 (Society for Industrial and Applied Mathematics (SIAM): Philadelphia).

Hein, T., Some analysis of Tikhonov regularization for the inverse problem of option pricing in the price dependent case. *J. Anal. Appl.*, 2005, **24**, 593–609.

Jiang, L., Chen, Q., Wang, L. and Zhang, J.E., A new well-posed algorithm to recover implied local volatility. *Quant. Finance*, 2003, **3**, 451–457.

Jiang, L. and Tao, Y., Identifying the volatility of underlying assets from option prices. *Inv. Problems*, 2001, **17**, 137–155.

Lagnado, R. and Osher, S., Reconciling difference. *Risk*, 1997, **10**, 79–83.

Lehoucq, R.B., Sorensen, D.C. and Yang, C., *ARPACK Users' guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, 1998 (Society for Industrial and Applied Mathematics (SIAM): Philadelphia).

Lions, J., *Optimal Control of Systems Governed by Partial Differential Equations*, 1971 (Springer-Verlag: Berlin).

Navon, I.M., Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. *Dynam. Atmos. Oceans*, 1998, **27**, 55–59.

Navon, I.M., Zou, X., Derber, J. and Sela, J., Variational data assimilation with an adiabatic version of the NMC spectral model. *Mon. Weather Rev.*, 1992, **120**, 1433–1446.

Turinici, G., Calibration of local volatility using the local and implied instantaneous variance. *J. Comput. Finan.*, 2009, **13**, 1–18.

Zhu, C., Byrd, R.H. and Nocedal, J., L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. *ACM Trans. Math. Softw.*, 1997, **23**, 550–560.