Recovering volatility from option prices by evolutionary optimization

Sana BEN HAMIDA[1,2] & Rama CONT [1]
(1) Centre de Mathématiques Appliquées
CNRS – Ecole Polytechnique, F-91128 Palaiseau, France.
(2) Ecole Supérieure de Technologie et Informatique, Tunis.

May 2004.*

Abstract

We propose a probabilistic approach for estimating parameters of an option pricing model from a set of observed option prices. Our approach is based on a stochastic optimization algorithm which generates a random sample from the set of global minima of the in-sample pricing error and allows for the existence of multiple global minima. Starting from an IID population of candidate solutions drawn from a prior distribution of the set of model parameters, the population of parameters is updated through cycles of independent random moves followed by "selection" according to pricing performance. We examine conditions under which such an evolving population converges to a sample of calibrated models.

The heterogeneity of the obtained sample can then be used to quantify the degree of ill–posedness of the inverse problem: it provides a natural example of a coherent measure of risk, which is compatible with observed prices of benchmark ("vanilla") options and takes into account the model uncertainty resulting from incomplete identification of the model.

We describe in detail the algorithm in the case of a diffusion model, where one aims at retrieving the unknown local volatility surface from a finite set of option prices, and illustrate its performance on simulated and empirical data sets of index options.

Keywords: model calibration, option pricing, inverse problems, volatility, evolutionary algorithms, stochastic optimization.

*This project has benefited from a research grant by Europlace Institute of Finance. Part of this work was developed in the framework of a research project on model calibration at HSBC-CCF, Division of Market and Model Risk. Earlier versions were presented at Humboldt University (Berlin), the AMAM 2003 Congress, the Satellite meeting on Mathematical Finance (Nice), the IPM Workshop on Inverse Problems (Tehran), HSBC Quants seminar (June 2003), Université de Paris X (MODALX) CREST Financial Econometrics seminar and Institute for Mathematics and Applications, Minneapolis (April 2004). We thank Kasra Barkeshli, Randall Douc, Helyette Geman, Pierre DelMoral and Franck Viollet for helpful discussions.

Contents

Stochastic models of financial markets usually represent the evolution of the price of a financial asset as a stochastic process $(S_t)_{t\in[0,T]}$ defined on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. An option on S with maturity T then corresponds to a random variable H_T, whose value is revealed at T and depends on the behavior of the underlying asset S between 0 and T. For example, a call option with maturity T_i and strike price K_i is then a financial contract which pays out $\max(0, S_{T_i} - K_i) = (S_{T_i} - K_i)^+$ to the holder at the maturity date T_i. The main focus of option pricing theory has been to define a notion of value for such options and compute this value. In arbitrage-free markets, the assumption of linearity of prices leads to the existence of a probability measure $\mathbb{Q}$ equivalent to $\mathbb{P}$ such that the value $V_t(H)$ of a terminal payoff H_T is given by:

$$V_t(H_T) \quad = \quad B(t,T)E^{\mathbb{Q}}[H_T|\mathcal{F}_t] \tag{1}$$

where $B(t,T)$ is a discount factor. For option pricing purposes, it is sufficient to know this pricing measure $\mathbb{Q}$. For example the value $C_t(T,K)$ of a call option with maturity T and strike K is given by

$$C_t(T,K) \quad = \quad B(t,T)E^{\mathbb{Q}}[(S_T - K)^+|\mathcal{F}_t] \tag{2}$$

Since the famous Black Scholes model was introduced in 1973, option markets have evolved to become autonomous, organized markets with a fairly high degree of liquidity especially for index options and foreign exchange options. In such markets, the market prices of a series of liquid options, which are often call or put options, are readily observed. These market prices are then used as a benchmark to "mark to market" or *calibrate* an option pricing model, which can then be used to compute prices of more complex ("exotic") options or compute hedge ratios.

The well-known smile and skew patterns in market option prices has led to the development of option pricing models generalizing the Black-Scholes model: local volatility (diffusion) models, stochastic volatility models, models based on jump processes. The price to pay for more realistic models is the increased complexity of model calibration: as noted by [Jacquier & Jarrow (2000)], in presence of complex models "the estimation method becomes as crucial as the model itself".

The availability of market prices for options has also made it feasible to identify such pricing models from market prices of options: this can be done by parameterizing the pricing measure by some parameter $\theta \in E$ and choosing θ to match an observed set $(C_t^*(T_i, K_i), i = 1..I)$ of call option prices:

$$C_t^*(T_i, K_i; \theta) \quad = \quad B(t,T_i)E^{\mathbb{Q}^\theta}[(S_{T_i} - K_i)^+|\mathcal{F}_t] \qquad i = 1..I \tag{3}$$

The parameter θ can be a finite dimensional vector: this is the case for instance in the Heston stochastic volatility model [Heston], the Merton jump-diffusion model or the CEV model. Alternatively, in the *non-parametric* approach, θ is identified with the local characteristics of the stochastic (risk-neutral) process S and is typically an element of an infinite dimensional space: the local volatility

function in the case of diffusion models [Dupire 1994] or the Lévy measure in the case of models with jumps [Cont & Tankov (2004)].

Determining the model parameter θ to match the market prices of a set of benchmark options is known to practitioners as the "model calibration" problem: it is the inverse problem associated to the option pricing problem. One of the difficulties in solving this inverse problem is that in practice the market information is insufficient to completely identify a pricing model: if the model is sufficiently rich, several sets of model parameters may be compatible with the market prices, leading to ill-posedness and model uncertainty.

Because of possible model misspecification, it is neither feasible nor meaningful in practice to match exactly the market prices. Therefore, the calibration problem is often reformulated as an optimization problem, where the goal is to minimize the pricing error or discrepancy between model prices and market prices for a set of liquidly traded options. A common way to measure this discrepancy is to use the (quadratic) difference between market and model prices, which leads to the nonlinear least squares method:

$$\inf_{\theta \in E} G(\theta) \qquad G(\theta) = \sum_{i=1}^{I} |C^{\theta}(t, S_t, T_i, K_i) - C_t^*(T_i, K_i)|^2 w_i \tag{4}$$

where $w_i > 0$ is a weight, $C_t^*(T_i, K_i)$ is the market price of a call option observed at date t and C^{θ} is the model price computed with a parameter θ. However the optimization problem (4) is still not easy to solve. As a function of the parameter θ, the objective function G is neither convex nor does it have any particular structure enabling the use of gradient-based minimization methods to locate the minima. Also, $G(.)$ is not given explicitly: its computation often involves a numerical method -either a finite difference solver, a Fourier transform or a Monte Carlo simulation- and computing its gradient may be even more difficult. More importantly, it is not clear whether there G a unique global minimum and even if this is the case, whether it can be reached by a gradient-based algorithm.

A remedy proposed in the literature [Avellaneda et al (1997), Coleman et al (1999), Cont & Tankov (2004), Crépey (2003), Jackson et al (1999), Lagnado & Osher (1997)] has been to use a *regularization* method, adding to the objective function (4) a convex penalization criterion $F : E \to \mathbb{R}^+$ which makes the problem well-posed and for which a gradient-based optimization procedure can be used:

$$\inf_{\theta \in E} G(\theta) + \alpha F(\theta)$$

Examples of penalization criteria are smoothness norms for volatility functions [Crépey (2003), Jackson et al (1999), Lagnado & Osher (1997)] and relative entropy [Avellaneda et al (1997), Cont & Tankov (2004)] for probability measures. When applied to a given set of market prices, these methods yield a single set of model parameters calibrated to the market but require the extra step of determining the regularization parameter α.

But, with or without regularization, deterministic optimization methods will at best locate one of the (local or global) minima of the fitting criterion, but have

nothing to say about the multiplicity of solutions of the initial (non-regularized) calibration problem (4). In other words, they provide a point estimate but no information about parameter uncertainty. However, the non-uniqueness of the solution of the original calibration problem is not simply a mathematical nuisance: the multiplicity of solutions contains interesting information on *model uncertainty*, which is lost through the process of regularization.

We describe here a probabilistic approach to the model calibration problem, which takes into account the multiplicity of solutions and the reflects the ill-posed character of the problem instead of suppressing it. Our method is based on a stochastic algorithm which generates a random sample from the set of calibrated models. Starting from an IID population of candidate solutions drawn from a prior distribution on the set of model parameters, the population of parameters is updated through cycles of independent random moves followed by "selection" using the calibration criterion. We examine conditions under which such an evolving population converges to the set of global minima of a pricing error such as (4), which may or may not be reduced to a single element.

Our approach naturally leads to a *family* of pricing models compatible with market prices. This family can then be used to quantify model uncertainty and its impact on derivative prices and provides an example of a coherent risk measure [Artzner et al(1999)] compatible with a set of observed option prices.

While most of the existing literature on model calibration has cast it in the framework of a deterministic optimization problem, yielding point estimates of model parameters [Andersen & Andreasen (2000), Avellaneda et al (1997), Coleman et al (1999), Crépey (2003), Cont & Tankov (2004), Jackson et al (1999)], several authors [Ait Sahalia & Lo (1998), Jacquier & Jarrow (2000), Lo (1986)] have cast this problem into a statistical framework, which leads to distributions on model parameters. Nonparametric kernel regression was used in [Ait Sahalia & Lo (1998)] for estimation of state price densities from option prices. This approach assumes IID errors across options and can produce, in addition to SPD estimates, confidence intervals on such estimates in large samples. By contrast with these methods, we do not rely on large sample results nor do we assume IID errors across options. Closer to the spirit of this work, [Jacquier & Jarrow (2000)] propose a Bayesian approach in the framework of the Black Scholes model: starting from a prior distribution on model parameters and an assumption on the distribution of observational errors, a posterior distribution is obtained by taking into account the observed option prices and a Monte Carlo algorithm is proposed for simulating from this posterior. The general idea of our approach is similar: we start from a prior distribution on model parameters, end up with a sample of model parameters and allow for model misspecification. However, our approach is based not on the application of the Bayes formula but on the minimization a pricing error such as (4) and does not rely on a specific distribution for observational errors (which is not known in practice). Also, the scope of application of our method is not limited to the Black Scholes model, as illustrated by the example of diffusion models in Section 2.

Section 1 presents a general methodology for model identification based on a set of observed option prices, applicable to a wide range of models. Section 2

specializes these results to the case of one-dimensional diffusion model ("local volatility" model) and describes in detail the algorithm used in this case. In order to assess the performance of our method, we first perform some numerical experiments on simulated data: these tests are presented in section 3. Section 4 presents the results obtained by applying the method to an empirical data set of DAX option prices. Section 5 discusses the implications of our methodology for measuring model uncertainty and its extensions beyond the setting of diffusion models or European options.

1 Evolutionary algorithms for model calibration

In this section we formulate a general framework for the model calibration problem and describe the use of a class of stochastic optimization methods, known as evolutionary algorithms, for solving it. After presenting the mathematical setting of the problem, we present the idea behind evolutionary optimization methods, their convergence properties and discuss how they apply in the context of model calibration.

1.1 Model calibration as an optimization problem

Consider an underlying asset modeled as stochastic process $(S_t)_{t\in[0,T]}$ on some probability space $(\Omega, \mathcal{F}, \mathcal{F}_t, \mathbb{P})$. An arbitrage-free pricing rule can be represented as a measure $\mathbb{Q} \sim \mathbb{P}$ such that the discounted price $\hat{S}_t = B(t,T)S_t$ is a martingale under $\mathbb{Q}$. Let $M(S)$ denote the set of such pricing rules. An option pricing model is given by a (parameterized) family of pricing measures

$$\begin{aligned} \theta \in E &\rightarrow M(S) \quad (5) \\ \theta &\rightarrow \mathbb{Q}^\theta \quad (6) \end{aligned}$$

Consider now a set of benchmark options, with terminal payoffs denoted by $H_i, i = 1..I$. Typically these benchmark options are European calls and puts in the case of index or foreign exchange option markets, or caps and floors in the case of interest rate markets. However for the sequel, we need not assume this is the case; the benchmark options may have path-dependent features, for example.

Denote by $C_i(\theta), i = 1..I$ the values of benchmark options at $t = 0$ under the pricing rule given by the parameter θ:

$$C_i(\theta) = B(t,T)\ E^{\mathbb{Q}_\theta}[\ H_i\ |\mathcal{F}_0]$$

where $B(t,T)$ is a discount factor. We will assume the parameter-to-price map

$$\begin{aligned} C : E &\mapsto \mathbb{R}^I \\ \theta &\mapsto C(\theta) = (C_i(\theta), i = 1..I) \quad (7) \end{aligned}$$

is continuous; this is the case for all option pricing models of interest. We assume now that the prices of these benchmark options are observed on the

market at $t = 0$; denote these prices by $C_i^*, i = 1..I$. The pricing rule $\mathbb{Q}^\theta$ is said to be compatible with the market prices $(C_i^*)_{i=1..I}$ if

$$\forall i = 1..I,\ C_i(\theta) = C_i^* \tag{8}$$

Eq. (8) should be seen as a system of implicit constraints for the model parameter θ and the calibration problem consists in determining (the set of parameters) θ satisfying (8). However, (8) may have no solution at all. Typically, if the model is misspecified, the observed option prices may not lie within the range of prices attainable by the model. Also, observed option prices are "noisy" estimates, defined up to a bid–ask spread: although they may be compatible, in the sense of bid-ask spreads, with the model prices for some parameter θ, they may not verify the equality (8) exactly for any given $\theta \in E$. For these reasons, it is a common approach to replace (8) by its least squares version:

$$\inf_{\theta \in E} G(\theta) \qquad\qquad G(\theta) = \sum_{i=1}^{I} w_i |C_i(\theta) - C_i^*|^2 \tag{9}$$

where $w_i > 0$ are a set of weights. In cases where the option prices $C_i(\theta)$ depends continuously on θ and when E is a compact subset of a finite dimensional space (i.e. there are a finite number of bounded parameters), the least squares formulation always admits a solution.

However, the solution of (9) need not be unique: G may in fact have several *global* minima, when the observed option prices do not uniquely identify the model. Even in the case where there is a unique global minimum, it may be difficult to compute using gradient-based optimization methods commonly used for this purpose. Besides the fact that the gradient of G is often not known explicitly, in most cases G is a continuous but non-convex function: figures 1–3 show examples of the function G for some popular parametric option pricing models, computed using a data set of DAX index options prices on May 11, 2001. Figure 1 corresponds to the quadratic pricing error in a lognormal-mixture diffusion model, described in [Brigo & Mercurio]: it displays a flat profile, many parameter combinations yielding equivalent fits. The quadratic pricing error in the Heston stochastic volatility model [Heston], shown in figure 2 as a function of the "volatility of volatility" and the mean reversion rate, displays a line of local minima. Finally, the pricing error for the Variance Gamma model [Madan & Milne (1991)] in figure 3 displays a strongly nonconvex profile, with two distinct minima in the range of observed values.

From figures 1–3 it is readily observed that, given the prices of the benchmark call and put options, there are several – sometimes a full range of – model parameters which are equally compatible with these market prices. In any of these (real) examples, gradient-based algorithms will converge to local minima of G or, at best, to *one* of the global minima of G, leaving us uninformed about the other possible solutions and their multiplicity. The existence of multiple solutions is not due to a specific numerical optimization method, but inherent to the ill-posedness of the problem at hand: instead of ignoring or bypassing

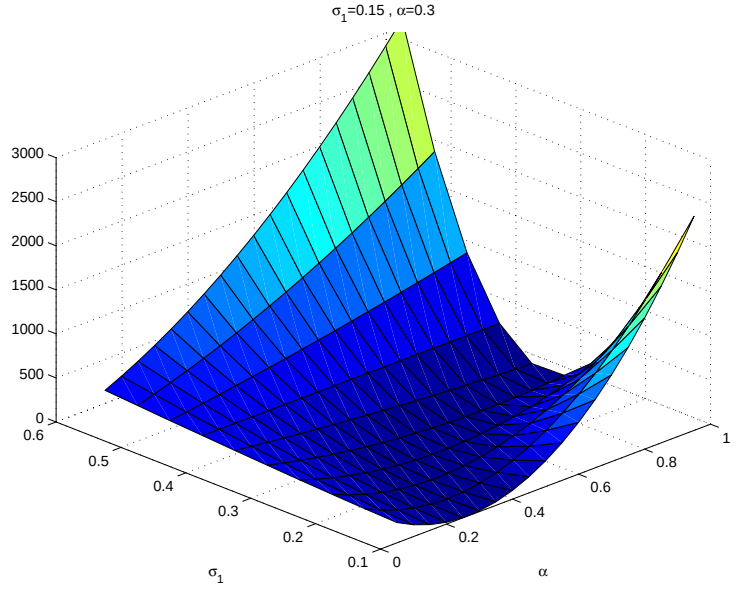

Figure 1: Error surface for lognormal density mixture model, DAX options.

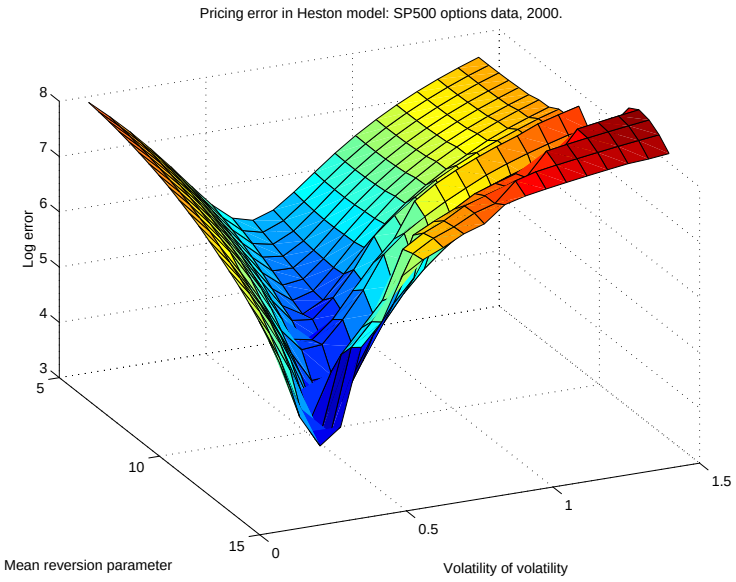

Figure 2: Error surface for the Heston stochastic volatility model, DAX options.

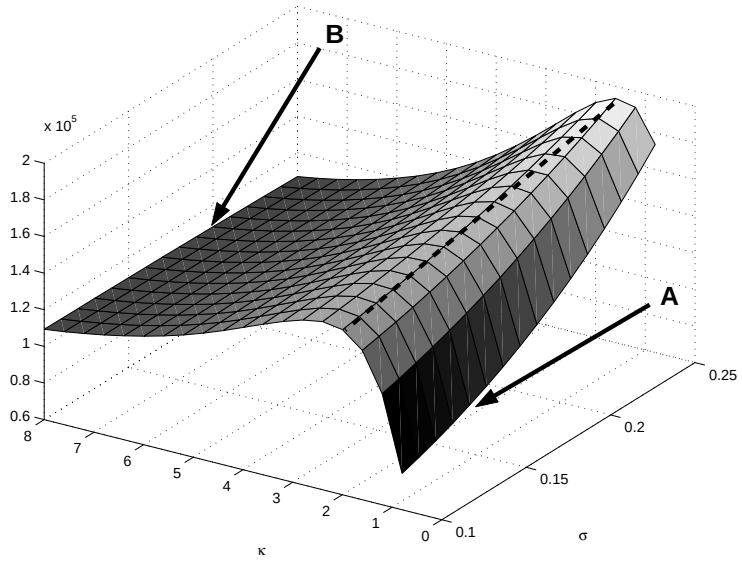

Figure 3: Error surface for Variance Gamma (pure jump) model, DAX options.

it using an exogenous criterion, it is thus interesting to explore the various solutions and try to quantify the associated model uncertainty. We will now describe a probabilistic approach, based on *evolutionary optimization*, which attempts at *sampling* from the set of solutions of (9), instead of selecting one of them using an exogenous criterion.

1.2 Evolutionary optimization: a brief overview

Evolutionary algorithms, introduced in [Holland], are order-zero stochastic optimization methods: they require neither differentiability nor convexity of the functions being optimized. Evolutionary algorithms are based on a random search of the parameter space by a population of optimizers undergoing "evolutionary pressure" based on an analogy with Darwinian selection of species [Holland, Bäck]. They are widely used to solve complex – high dimensional and nonconvex – optimization problems [Kallel et al.]. This section gives an overview of evolutionary algorithms and their convergence properties.

Consider a search space E and a continuous "fitness" function $G : E \mapsto [0, \infty[$ to be minimized on E. In practice E will be taken to be a compact set but this condition is not strictly necessary. An evolutionary algorithm with objective function G is based on the evolution of a population of candidate solutions (*individuals*), denoted by $X_n^N = (\theta_n^i, i = 1 \cdots N)$, where n is the current step (*generation*), N is the population size and θ_n^i is a candidate minimizer of G. For $X = (\theta^1, ..., \theta^N) \in E^N$ denote $[X] = \{\theta^1, ..., \theta^N\}$.

The idea is to "evolve" the population $\theta_n^i, i = 1 \cdots N$ through cycles of

modification (mutation) and "Darwinian" selection in order to improve the performance of its individuals, as measured by the fitness function G. At each iteration n, the population undergoes three transformations:

$$X_n^N \underset{mutation}{\longrightarrow} V_n^N \underset{crossover}{\longrightarrow} W_n^N \underset{selection}{\longrightarrow} X_{n+1}^N \tag{10}$$

During the mutation stage, individuals undergo independent random transformations, as if performing independent random walks in E, resulting in a randomly modified population V_n^N. In the crossover stage, pairs of individuals chosen from the population to "reproduce': each pair gives birth to a new individual, which is then added to the population. This results in a new diversified population W_N^n, regrouping parents and children, with $> N$ elements. This new population is now evaluated using the fitness function $G(.)$: we compute $G(x)$ for every $x \in [W_n^N]$. Elements of the population are now selected for survival with according to their fitness: those with a lower value of G have a higher probability of being selected. One such selection rule is the following: each individual $x \in [W_n^N]$ is selected with probability proportional to $\exp[-\beta_n G(x)]$. Here $\beta_n > 0$ is a parameter called the selection pressure: $\beta_n \to \infty$ leads to elitist selection, retaining only the individuals with lowest values of $G(.)$ while $\beta_n \to 0$ means we select individuals at random, regardless of their performance. The N individuals thus selected then form the new population, X_{n+1}^N.

The following flow chart summarizes the structure of an evolutionary algorithm.

Structure of an Evolutionary Algorithm

$n \leftarrow 0$

Draw N IID points in E from prior distribution μ_0: $\theta_0^i \sim \mu_0$.
Initialize population $X_0^N = (\theta_0^1, \theta_0^2 \cdots, \theta_0^N)$

Evaluate X_0^N: compute $G(\theta_0^i)$, $i = 1 \cdots N$

while (**not** termination-condition)**do**

begin
$n - 1 \to n$
Mutation: randomly modify individuals in $X_{n-1}^N \mapsto V_n^N$
Generate new individuals by crossover $\mapsto W_n^N$
Evaluate individuals in W_n^N
Select X_n^N from W_n^N
end

The role of mutation is to explore the parameter space and the optimization is done through selection. The idea is that, by analogy with Darwinian

evolution, cycles of mutation followed by selection will globally improve the population's performance and lead it to regions where $G(.)$ is minimal. Note that the gradient of G is not used at any time: movements are totally random, corrected only through the selection procedure. Crossover is used to enhance the search in parameter space. Aside from the crossover step, the mutation–selection cycle is similar to the prediction–correction step in the EM algorithm or in filtering problems. In mathematical terms, the evolutionary algorithm described above corresponds to simulating an (inhomogeneous) Markov chain $(X_n^N)_{n\in\mathbb{N}}$ in E^N.

- The initial population is an IID sample drawn from a prior distribution μ_0 on E.
- Mutation: each individual θ^i evolves independently following a transition kernel $M_n(x, dy)$ on E.
- Selection: at the n-th iteration, given the population $\theta_n^i, i = 1..N$ each individual θ_n^i is selected with probability $\exp[-\beta_n G(\theta_n^i)]$; if not selected, it is replaced by another individual θ_n^j selected according to the distribution:

$$\frac{\exp[-\beta_n G(\theta_n^j)]}{\sum_{k=1}^N \exp[-\beta_n G(\theta_n^k)]}$$

  The selection pressure β_n is progressively increased: $\beta_n \to \infty$ as $n \to \infty$.
- Crossover: each pair (θ^i, θ^j) is selected according the selection mechanism above and then evolves following a transition kernel C_n on E^2.

Asymptotics and concentration properties of mutation/selection algorithms have been studied in the recent [Cerf (1996), Cerf (1998), Del Moral & Miclo (2000), Del Moral & Miclo (2001), Del Moral & Miclo (2003), Löwe (1996)] using "small noise" asymptotics and large deviation techniques. Most studies are limited to mutation–selection algorithms, not including crossover, which complicates the picture. In the sequel we will give convergence results in the framework where there is no crossover. These results give various conditions on the mutation and selection kernel under which, for large n the population becomes concentrated on the set of minima of G.

A first set of results [Cerf (1996), Cerf (1998)] states that, when E is finite and for a large enough population size $N \geq N_0$, if the mutation allows to explore sufficiently the parameter space and the mutation noise is gradually decreased to zero as $n \to \infty$ then the population gradually settles down on the set of global optima of G.

Another approach is to study the distribution of individuals in large populations ($N \to \infty$) [Del Moral & Miclo (2000), Del Moral & Miclo (2001)]. In this framework, one considers as primary object not the positions (θ_n^i) of the individuals but the population distribution $\mu_n^N = N^{-1} \sum_{i=1}^N \delta_{\theta_n^i}$, which then defines a flow $(\mu_n^N)_{n\geq 0}$ on $M_1(E)$, the space of probability measures on E. Under some technical conditions on the mutation and selection kernels [Del Moral (2004)],

the flow μ_n^N weakly converges as $N \to \infty$ to a distributional flow $(mu_n)_{n\geq 0}$ given by the measure–valued dynamical system

$$\mu_{n+1} \quad = \quad M_n \; S^n_{\mu_n} \mu_n \tag{11}$$

where mutation and selection steps are represented by (nonlinear) transformations acting on μ_n: M_n is the mutation kernel at the n-th generation and S^n_μ is a (state-dependent) transition kernel corresponding to the selection rule described above:

$$S^n_\mu(x, dy) = e^{-\beta_n G(x)} \delta_x(dy) + (1 - e^{-\beta_n G(x)}) \frac{e^{-\beta_n G(y)} \mu(dy)}{\int \mu(dz) \exp[-\beta_n G(z)]} \tag{12}$$

The Markov chain $(X_n^N)_{n\geq 0}$ described above then corresponds to a N-particle approximation of the measure-valued dynamical system (11). Conditions under which this convergence is uniform with respect to $n \geq 0$ are given in [Del Moral et al. (2001)]: the distribution of the individuals in the evolutionary algorithm X_n^N then behaves like a sample from μ_n as $n \to \infty$. The goal is then to choose the mutation kernels M_n and the selection parameter β_n such that when $n \to \infty$ the distribution μ_n concentrates on the minima of G. By contrast with the situation described in [Cerf (1998)], finiteness of E is not needed. Also, one can keep the mutations time-homogeneous and concentrate the population on the set of optima by gradually increasing the selection rate β_n as $n \to \infty$ [Del Moral & Miclo (2003)].

Denote by $H(Q|P)$ the relative entropy of a probability distribution Q with respect to P and define

$$I(\mu) = \inf_K \int_E \mu(dx) \quad H(K(x, .)|M(x, .)) \tag{13}$$

where the infimum is taken over all Markov kernels with stationary measure μ. The following result [Del Moral & Miclo (2003), Prop.4.2.] shows that if the mutation kernel verifies a mixing condition and the selection pressure is progressively increased, the population concentrates in the level set corresponding to the "minimum" of G in the following sense:

Proposition 1 (Concentration when $n \to \infty$) *Let G be a function of bounded oscillation, choose $\beta_n = n^a,\ a \in]0, 1[$ and assume the mutation kernel $M_n = M$ verifies the following*

$$\text{Mixing condition} \qquad \exists \epsilon > 0, \forall (x, y) \in E^2, M(x, .) \geq \epsilon M(y, .) \tag{14}$$

Then the population concentrates in the level set corresponding to G^*:

$$\forall \delta > 0, \qquad \mu_n(G(\theta) \geq G^* + \delta) \to 0 \text{ as } n \to \infty.$$

where G^* is the solution of the variational problem

$$G^* \quad = \quad \inf\{\int_E G(\theta)\mu(d\theta), \mu \in M_1(E), I(\mu) < \infty\} \tag{15}$$

Under appropriate conditions on the kernel M, G^* coincides with the essential infimum of G with respect to the invariant measure of the kernel M.

Properties of the interacting particle system $(X_n^N)_{n\geq 1}$ has also been investigated when the population size N becomes large. One interesting property, usually known under the (obscure) name of "propagation of chaos", states the following: under some technical conditions and if the initial population is IID, when population size $N \to \infty$, the joint law of any finite subset $(\theta_n^1, ..., \theta_n^k)$ (k fixed) of individuals converges weakly to a product law.[1] This result can be interpreted as follows: when the population size N is large, the individuals in each finite subpopulation $(\theta_n^1, ..., \theta_n^k)$ will behave as if they were independent. This suggests the following statistical interpretation : if N is large then, for large n a finite subpopulation $(\theta_n^1, ..., \theta_n^k)$ can be viewed as a sample of *independent* draws from the level set $\{x \in E, G(x) \leq G^* + \delta\}$: one can use the evolutionary algorithm to obtain a sample of independent δ-optima from the function G.

1.3 Evolutionary algorithms for model calibration

Consider now the setting of option pricing models described in Section 1.1. We consider the pricing error $G : E \mapsto [0, \infty[$ defined by:

$$G(\theta) = \sum_{i=1}^{I} w_i |C_i(\theta) - C_i^*| \tag{16}$$

where $C_i(\theta)$ are model prices and C_i^* are observed (transaction or mid-market) prices for the benchmark options. We will assume that the parameter space E is compact: in most cases, it is simply a compact subset of $\mathbb{R}^d$. If the model is capable of perfectly fitting the data, then $\min G = 0$ otherwise $\min G > 0$. However, even if $\min G = 0$ we are not necessarily interested in computing the zeros of G. Recall that mid-market or transaction prices C_i^* are defined up to a bid-ask interval $[C_i^{\text{bid}}, C_i^{\text{ask}}]$. Thus, it is not meaningful for a model to retrieve the value of C_i^* with a precision much higher than $|C_i^{\text{bid}} - C_i^{\text{ask}}|$. Define now the a priori error level δ by:

$$\delta = \sum_{i=1}^{I} w_i |C_i^{\text{bid}} - C_i^{\text{ask}}| \tag{17}$$

Given the uncertainty on option values due to bid-ask spreads, one cannot meaningfully distinguish a "perfect" fit $G(\theta) = 0$ from any other fit with $G(\theta) \leq \delta$. Therefore, all parameter values in the level set $\{\theta \in E, G(\theta) \leq \delta\}$ correspond to models which are compatible with the market data $(C_i^{\text{bid}}, C_i^{\text{ask}})_{i=1..I}$. As noted before, in general the evolutionary algorithm described above allows to reach the level $G^* \geq \min G$ given by (15). We will later see that under some conditions G^* coincides with the (essential) infimum of G. If $G^* > \delta$ then *no*

[1] We refer to [Del Moral & Miclo (2000)] for a precise formulation of results on propagation of chaos for evolutionary algorithms and other particle systems.

model in the class considered is capable of reproducing market prices with the required accuracy and the model is clearly misspecified. Thus, we will assume in the sequel that

$$G^* \leq \delta \tag{18}$$

We will see that this assumption can indeed be verified in all empirical examples below. In this case the level set $\{\theta \in E, G(\theta) \leq \delta\}$ is not empty : some parameter values will satisfy the calibration requirement within the desired accuracy, and our objective will be to sample from this set.

When (18) is satisfied, under the conditions of proposition (1), the evolutionary algorithm yields a population of points which converges to a sample of model parameters compatible with the market data $(C_i^{\text{bid}}, C_i^{\text{ask}})_{i=1..I}$ in the sense that $G(\theta) \leq \delta$. Note that this means that, as $n \to \infty$, *all* elements of the population converge end up in the level set $\{\theta \in E, G(\theta) \leq \delta\}$ so we obtain not only one solution but a *population* of model parameters calibrated to market data. Using the "propagation of chaos" concept described above, one can sample a subpopulation of such parameters and regard it as a sample of statistically independent draws from the set of calibrated model parameters.

Compared to existing calibration algorithms, evolutionary optimization methods present the following advantages:

- No interpolation of option prices is required: contrarily to implied tree [Derman et al (1996)] methods or methods based on the Dupire formula [Dupire 1994], we does not require call/put prices for all strikes or maturities nor any ad-hoc interpolation of observed prices. They than therefore be applied to index options where the number of observations is large ($I \simeq 100 - 200$) but also to equity options for which data is scarcer ($I \simeq 20 - 30$).

- They avoid computing the (high dimensional) gradient of the objective function, an essential but difficult step in other algorithms.

- They do not require convexity of the objective function being minimized. As seen in figures 1–2, this property may actually not hold in many commonly used models.

- Evolutionary algorithms provide a *population* of solutions instead of a single model uncertainty/ degree of ill-posedness is reflected in the heterogeneity of this population.

We will now present a case study to show how this methodology can be used to estimate parameters of an option pricing model from market data.

2 Recovering diffusion coefficients from option prices

The methodology outlined above is quite general and is not tied to a certain class of option pricing models. However, to illustrate its performance in a specific example, we will now specialize it to the class of Markovian diffusion models, popular in option pricing applications, where the underlying is modeled as a diffusion process:

$$\frac{dS_t}{S_t} = \mu(t, S_t)dt + \sigma(t, S)dW_t \tag{19}$$

The model parameter here is the *local volatility function* $\sigma(.,.)$. The calibration problem then consists of identifying the local volatility function $\sigma(S,t)$ from prices of call or put options observed in the market. This is usually done by parameterizing the volatility function with some parameter $\theta \mapsto \sigma_\theta$ and minimizing the quadratic pricing error over parameter values $\theta \in E$:

$$G_2(\theta) = \sum_{i=1}^{I} w_i |C^{\sigma_\theta}(t, S_t, T_i, K_i) - C_t^*(T_i, K_i)|^2 \tag{20}$$

here $C_t^*(T_i, K_i)$ is the market price of a call option with strike K and maturity T and $C^\sigma(.,., T_i, K_i)$ is the model price. The model price is obtained as the solution of the partial differential equation:

$$\begin{cases} \forall S > 0, t \in [0, T[, \quad \frac{\partial C^\sigma}{\partial t} + \frac{\sigma^2(t,S)S^2}{2}\frac{\partial^2 C^\sigma}{\partial S^2} - rS\frac{\partial C^\sigma}{\partial S} = rC(t, S; T, K). \\ C^\sigma(T, S, T, K) = (S - K)^+. \end{cases} \tag{21}$$

Various numerical methods have been proposed for estimating local volatility functions from call option prices. Dupire [Dupire 1994] presents a formula for reconstructing local volatility functions from a continuum of call option prices; however this formula involves taking derivatives from discrete data and is numerically unstable. A discretized version of the Dupire formula is the implied tree method of Derman & Kani [Derman et al (1996)], which is prone to similar instabilities leading to "negative probabilities". Other methods, based on nonlinear least squares [Coleman et al (1999)] or regularized versions of it [Achdou & Pironneau, Avellaneda et al (1997), Crépey (2003), Jackson et al (1999), Lagnado & Osher (1997), Samperi (2002)] all lead to minimization problems which are solved using gradient based algorithms.[2] In the non-parametric case, the minimization variable is the function $\sigma(.,.)$. After discretization, this leads to a (nonconvex) minimization problem in a high dimensional space, which presents the following difficulties:

- The gradient of G is costly to compute. This makes gradient-based optimization methods rather time consuming.

[2] See [Crépey (2003)] for a review.

- The objective function G is not convex and typically presents many (local or global) minima. As a consequence, gradient based methods tend to get trapped in local minima. Also, lack of convexity creates problems when using duality techniques as in [Avellaneda et al (1997), Samperi (2002)] and may lead to a duality gap.

We will now describe how the evolutionary algorithm described above can be adapted to the reconstruction of local volatility functions from option prices and allows to overcome these problems.

2.1 An algorithm for reconstructing local volatility

Evolutionary optimization algorithms are often criticized for their slow convergence; however, much of this criticism is due to the use of black-box evolution operators embedded in general-purpose libraries, which may have no relation with the problem at hand. In our case, a priori knowledge of the structure of local volatility functions will allow us to design the Markov chain in a way that improves considerably the convergence and accuracy of the algorithm.

The ingredients of the algorithm are the following:

1. A pricing algorithm (in this case, a finite difference solver) which takes as input a local volatility function, a strike and a maturity and returns the value of the option in the model (19): $[\sigma(.,.), K, T] \rightarrow C^{\sigma}(T, K)$

2. A parameterization of the set of admissible local volatility surfaces:

$$\begin{array}{ccc} E & \mapsto & H \\ \theta & \mapsto & \sigma_{\theta}(.,.) \end{array} \tag{22}$$

   where H is a space of smooth functions $[0, T] \times \mathbb{R} \rightarrow \mathbb{R}^+$ which satisfy our a priori assumptions on the behavior of a local volatility function.

3. A prior distribution μ_0 on E summarizing our prior information on the local volatility function $\sigma(.,.)$: its level, its degree of smoothness, etc.

4. A set of market prices of call options: $C^* = (C^*(T_i, K_i), i = 1..I)$

5. An objective function to be minimized on E: while (20) is a common choice, it should be noted that the only reason for using squared errors is to obtain differentiability. Since the evolutionary algorithm does not require differentiability of the objective function, instead of the quadratic pricing error we can also use the absolute pricing error which yields more stable numerical values:

$$G(\theta) = \sum_{i=1}^{I} w_i |C^{\sigma_\theta}(t, S_t, T_i, K_i) - C_t^*(T_i, K_i)| \tag{23}$$

We now describe each of these ingredients in more detail.

2.2 Representation of the local volatility surface

We consider a representation similar to the one adopted in [Jackson et al (1999)]. Consider a tenor $\mathbb{T} = \{T_1, ..., T_n\}$ of maturities and let $\Delta x > 0$ be a positive number representing the resolution of the representation in the log-strike/price dimension and $x_k = k\Delta x, k = -K..0..K$. For each maturity $T \in \mathbb{T}$ we represent the local volatility function $\sigma(T, .)$ as a function of log-price $x = \ln(S/S_0)$ in the following way[3]: for any maturity $T \in \mathbb{T}$, we represent $\sigma(T, .)$ as a cubic spline:

$$\forall i = 1..n, \sigma_\theta(T_i, .) = \sum_{m=0}^{M} \theta(i, m)\phi_m(.) \tag{24}$$

where $\phi_0(S) = 1$ is the constant function equal to 1 and $(\phi_m, m = 1..M)$ is the B-spline basis associated with the "knots" $(x_k, k = -K, .., 0, .., K)$, described in the appendix. For $t \notin \mathbb{T}$, we define $\sigma(t, x)$ by linear interpolation: if $t \in [T_i, T_{i+1}]$ with $t = \alpha T_i + (1 - \alpha)T_{i+1}$ with $\alpha \in [0, 1]$ we set

$$\sigma(t, x) \quad = \quad \alpha\sigma(T_i, x) + (1 - \alpha)\sigma(T_{i+1}, x) \tag{25}$$

For $t > T_n$ or $t < T_1$ we extrapolate by:

$$\forall t > T_n, \qquad \sigma(t, x) = \sigma(T_n, x) \tag{26}$$

$$\forall t < T_1, \qquad \sigma(t, x) = \sigma(T_1, x) \tag{27}$$

In practice, all coefficients $\theta(i, m)$ are taken to be bounded: they are positive and bounded from above by some positive number, say c. Denote by H the space of local volatility functions generated in this way. H is a subset of a finite dimensional space of smooth functions $[0, T] \times \mathbb{R} \mapsto \mathbb{R}$. A smooth local volatility surface $\sigma(., .) \in H$ is thus represented by the matrix of nd coefficients, which we will denote by $\theta = [\theta(i, m), m = 0..M, i = 1..n] \in [0, c]^d$. $\sigma(., .)$ will be positive if these coefficients are positive. This representation separates the "Black-Scholes" component, represented by the coefficients $\theta(., 0)$ from the other coefficients $\theta(i, m), m = 1..M$ which represent the implied volatility smile or skew.

2.3 Generating the initial population

The first step in the evolutionary algorithm is to generate a family of local volatility functions $\sigma_0^j(., .), j = 1..N$ which constitute the initial population of candidate solutions. As described in Section 1, we generate this initial population using some prior distribution on the local volatility surface, capturing the information available on its level and shape.

What prior information do we possess on local volatility functions? First, the at-the-money implied volatilities can give a good idea of the level of local

[3]By abuse of notation, we continue to denote by $\sigma(t, x)$ the volatility in the log-price variable.

volatilities: by analogy with a Black-Scholes model with time-varying volatility, we can set the constant (level) component $\theta(.,0)$ according to:

$$\forall T_i \in \mathbb{T}, \theta(i,0) = \frac{\Sigma_{ATM}(T_{i+1}) - \Sigma_{ATM}(T_i)}{T_{i+1} - T_i} \tag{28}$$

where $\Sigma_{ATM}(T)$ is the at-the-money implied volatility for the maturity T. Apart from this, the only other prior requirement we might have is to require the local volatility to be "smooth" in (t,x): in practice, this means we want to avoid highly oscillatory behavior. To integrate this requirement we use a *smoothness prior* i.e. a prior distribution on H which generates surfaces which are typically smooth as a function of price (and to a lesser extent, in the time variable). In order to quantify the smoothness we can use for instance the (semi)-norm:

$$||\sigma||^2_{1,2} = \sum_{T\in\mathbb{T}} \int_0^T dt \int_{-L}^{L} dx [|\frac{\partial^2 \sigma}{\partial x^2}(t,x)|^2 + |\frac{\partial \sigma}{\partial t}(t,x)|^2] \tag{29}$$

A possible choice of smoothness prior is then to choose a Gaussian measure on E with density:

$$\frac{1}{(2\pi)^{d/2}} \exp[-\frac{||\sigma||^2_{1,2}}{2\gamma^2}] \tag{30}$$

where d is the dimension of the parameter space. A typical draw from this Gaussian measure will thus be a surface σ with $||\sigma||_{1,2}$ of the order γ so the parameter γ can be used to control the smoothness of surfaces generated from this prior.

To simulate from this prior we use the representation (24). Each surface $\sigma_0^j(.,.)$ is represented by its coefficients $\theta_0^j = [\theta_0^j(i,m), i = 1..n, m = 0..M]$. We set the reference level using the at the money implied volatility according to (28). We compute the "stiffness" matrix A defined by

$$A_{ij} = \int dx \phi_i''(x) \phi_j''(x) \tag{31}$$

and use a Cholesky decomposition to find a matrix B such that $A = BB'$. We then set:

$$\theta(i,.) = B.\epsilon^i \tag{32}$$

where $\epsilon^i \in \mathbb{R}^M$ and $(\epsilon^i(T), i = 1..N)$ are IID $N(0, I_M)$ vectors.[4] Note that smoothness norms such as (29) have also been used in [Lagnado & Osher (1997), Crépey (2003)] as regularization terms in order to penalize $G(.)$. Using a smoothness prior as defined here allows to introduce information on the smoothness of the volatility surface via the prior, without modifying the objective function G.

[4]Note that, in fact, since the coefficients $\theta(i,m)$ are bounded the actual distribution corresponds to a truncated Gaussian distribution, values larger than the bounds being rejected and replaced by new ones.

2.4 Computation of the objective function

We choose as weights in the objective function

$$w_i = \max(\frac{1}{Vega(T_i, K_i)}, 100)$$

where $Vega(T_i, K_i)$ is the Black-Scholes Vega of the option computed using the market implied volatility. This weight 'converts' errors in price into errors in implied volatility, thus rescaling all terms in the sum defining G to the same order of magnitude. Thresholding by 100 avoids overweighting of options very far from the money. In order to compute the objective function at each step, the option prices $C^\sigma(t_0, S_0; T_i, K_i)$ have to be computed from the volatility surface $\sigma(.,.)$ and substituted in (23). This can be done in principle by solving the partial differential equation (21) to compute $C^\sigma(t, S; T, K)$. The price we are interested in is then given by $C^\sigma(t_0, S_0; T, K)$. This method is in fact quite heavy because it implies solving (21) at each time step for *each* option in the observation set. In the case where the calibration instruments are European call options, this procedure can be speeded up by a factor equal to the number of options being calibrated, by remarking that the call option price $C(t_0, S_0, T, K)$, as a function of the strike and maturity (K, T), verifies the Dupire equation [Dupire 1994] :

$$\begin{aligned} &\frac{\partial C}{\partial T} + Kr\frac{\partial C}{\partial K} - \frac{K^2\sigma^2(T,K)}{2}\frac{\partial^2 C}{\partial K^2} = 0. \\ &\forall K \geq 0, \quad C(t_0, S, t_0, K) = (S-K)^+. \end{aligned} \tag{33}$$

Solving this equation will then give us the whole range of call prices for all strikes and maturities in a single sweep. Using a logarithmic change of variable $x = \ln(K/S_0)$ and $u(T, x) = C(t_0, S_0; T, S_0 e^x)$, (33) is equivalent to:

$$\begin{aligned} &\frac{\partial u}{\partial T} = \frac{\sigma^2}{2}\frac{\partial^2 u}{\partial x^2} - \left(\frac{\sigma^2}{2} + r(t)\right)\frac{\partial u}{\partial x}(T, x) \\ &\forall x \in]-\infty, \infty[, \quad u(0, x) = (S_0 - S_0 e^x)^+ \end{aligned} \tag{34}$$

Equation (34) is then localized to a bounded domain $x \in [-L, L]$ and discretized using an implicit finite difference scheme on a uniform grid. Unconditional stability of the implicit scheme reduces the number of time steps and allows to use coarser grids in the first stages of the evolution and refine progressively as the need for precision appears. For the localized problem we use the numerical boundary conditions:

$$u(T, L) = 0 \qquad u(T, -L) = S_0(1 - e^{-L}) \tag{35}$$

which correspond to the asymptotic behavior of call prices for small and large strikes. The instantaneous discount rate $r(t)$ is modelled as a piecewise constant function computed from a set of discount factors with maturities $\{T_1, ..., T_n\}$.

2.5 The evolution scheme

As described in Section 1, the population of parameters undergoes a cycle of mutation, crossover and selection at each iteration. In the case of volatility functions, we need to ensure that these transformations allows to explore sufficiently the parameter space but avoids generating pathological volatility functions with non-smooth or oscillating features.

- Mutation: the mutation step amounts to randomly modifying each volatility function by adding a noise term to the spline coefficients. Noise terms are IID across individuals in the population but in order to conserve the smoothness of the surfaces, they cannot be IID across spline components. We choose them to have the covariance A computed in (31), so as to generate smooth surfaces:

$$\forall j \in \{1, \cdots, N\}, \forall i = 1..n \;\; \theta^j(i,.) \longleftarrow \theta^j(i,.) + B.\epsilon^j$$

  where $\epsilon^j \sim N(0, I_M)$ are IID across individuals and B is as described in section 2.3. More precisely, since E is bounded, moves beyond the imposed bounds are rejected: denoting by $V_n^N = (\gamma_n^i, i = 1..N)$ the modified population:

$$\begin{aligned} \gamma_n^j(i,.) = \theta_n^j(i,.) + B.\epsilon^j \quad &\text{if} \quad \theta_n^j(i,.) + B.\epsilon^j \in [0,c]^d \\ \gamma_n^j(i,.) = \theta_n^j(i,.) \quad &\text{if} \quad \theta_n^j(i,.) + B.\epsilon^j \notin [0,c]^d \end{aligned}$$

  The mutation kernel M is thus a "truncated" Gaussian kernel:

$$\begin{aligned} M(x,dy) &= M(x,x)\delta_x(dy) + 1_{y \in E} \frac{\exp[-\frac{1}{2}{}^t(y-x)A^{-1}(y-x)]}{\sqrt{(2\pi)^d |A|}} dy \\ \text{where} \;\; M(x,x) &= 1 - N(x,A)([0,c]^d) \end{aligned} \tag{36}$$

  is the probability that the mutation leaves the point x unmodified and $N(x,A)$ is the normal distribution centered at x with covariance A.

- Selection: at the n-th iteration, given the population $\theta_n^i, i = 1..N$ each individual σ_n^i is selected with probability $\exp[-\beta_n G(\theta_n^i)]$; if not selected, it is replaced by another individual θ_n^j selected according to the distribution:

$$\frac{\exp[-\beta_n G(\theta_n^j)]}{\sum_{k=1}^N \exp[-\beta_n G(\theta_n^k)]}$$

  The selection pressure is increased with the number of iterations: $\beta_n = n^a$ with $0 < a < 1$.

- Crossover: this extra step is not strictly required for convergence but enhances the search procedure in practice. First, we select, using the selection procedure outlined above, two individuals σ^j and σ^k from the population. The probability of an individual being selected is therefore

proportional to its calibration performance. Then, we generate an independent uniform random variable $\alpha \in [0,1]$ and create a new individual σ^l by convex combination : $\sigma^r = \alpha\sigma^j + (1-\alpha)\sigma^k$. σ^r is then added to the current population. Note that this operation preserves smoothness of the surfaces since

$$||\sigma^r||_{1,2} \leq \alpha||\sigma^k||_{1,2} + (1-\alpha)||\sigma^j||_{1,2}$$

Neglecting the effect of the crossover step in the convergence analysis, we obtain the following convergence result, a proof of which is outlined in the appendix:

Proposition 2 (Convergence to global minima) *Let μ_n denote the distribution of the population after n mutation-selection cycles described above. Then, as $n \to \infty$, μ_n concentrates on the set of global minima of the pricing error G:*

$$\forall \epsilon > 0,\ \mu_n(G(\theta) \leq \min_E G + \epsilon) \stackrel{n\to\infty}{\longrightarrow} 0.$$

3 Numerical experiments

To assess the performance of the algorithm we perform a series of numerical tests in which the algorithm is used to retrieve a known diffusion coefficient (i.e. local volatility function) from a set of option prices generated from it.

Figure 4 shows the local volatility function $\sigma_0(.,.)$ used for the test.[5] A set of call option prices $C^*(0, S_0, T_i, K_i)$ was computed from $\sigma_0(.,.)$ by solving the Dupire partial differential equation (33). In order for the test set to be similar to the empirical data sets available to us, we used 70 unevenly spaced values of strikes and maturities for $(T_i, K_i), i = 1..70$. To each of these prices, we add independent noise components with standard deviation equal to 0.1% of the option price; this observational noise simulates the effect of bid-ask spreads in real data.

These prices are then used as inputs for the calibration algorithm described in Section 2. A population of $N = 50$ solutions was used in this case. Volatility functions were parameterized as described in section 2.2, using a tenor of 3 maturities and 10 spline nodes per maturity. Figure 5 illustrates the evolution of the population: the performance (measure with G) of the average individual, the best individual as well as the standard deviation of the population performance converge after 15 iterations to values very close to zero, which indicates a population of local volatility functions with very good calibration performance. The a priori error defined in (17) is normalized here to $\delta = 1$: we therefore see that in this case $G^* \ll \delta$ and the level set $\{\theta \in E, G(x) \leq \delta\}$ is attained. These local volatility functions, although very similar with regard to calibration performance, can be quite different in their actual values: figure 6 shows some

[5] This function was actually obtained by calibrating a diffusion model to DAX option prices, as described in Section 4.

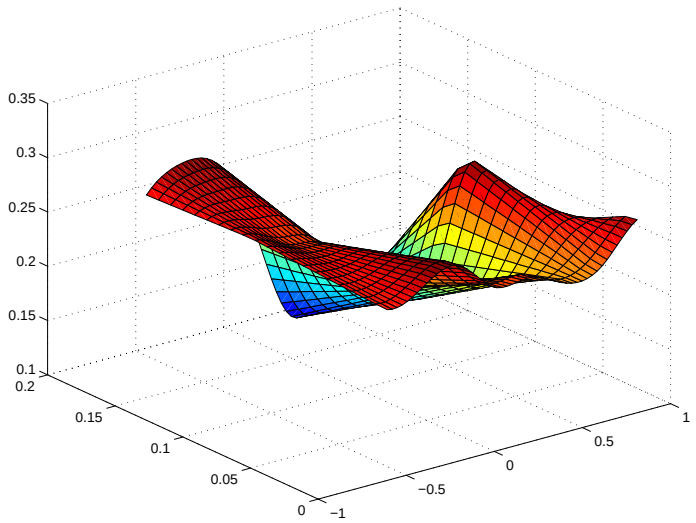

Figure 4: Local volatility function $\sigma_0(.,.)$ used to generate option prices in the numerical experiments.

of the best performing ones. Figure 3 shows the retrieval error in terms of implied volatility. Each point represents the calibration error in implied volatility units for one option in the data set. As shown in the figure, the input implied volatilities are retrieved with a precision of a few basis points, which is quite acceptable by comparison to market bid-ask spreads.

To summarize, these numerical tests show that the evolutionary algorithm is capable of retrieving a local volatility function from a data set of option prices with realistic size and features. Not only does the algorithm retrieve the input local volatility function and its associated implied volatilities with a precision of a few basis points, it also identifies *other* local volatility function which are different but which lead to similar prices for the benchmark options in the calibration set: they correspond to volatility functions which are not distinguishable from σ_0 given only the observed option prices.

4 Empirical results: DAX index options

Motivated by the positive results in the numerical experiments, the algorithm described above was applied to a data set of market prices of European call and put options on the DAX index, quoted on June 13, 2001. Around 100 option prices were available to us on this date, with different strikes and maturities running from a week up to one year. Figure 4 shows the implied volatility of these options as a function of strike and maturity. Options with a Black–Scholes vega less than 0.01 were discarded from the calibration set: such options are too far from the money to be liquid and the uncertainty on implied volatility is

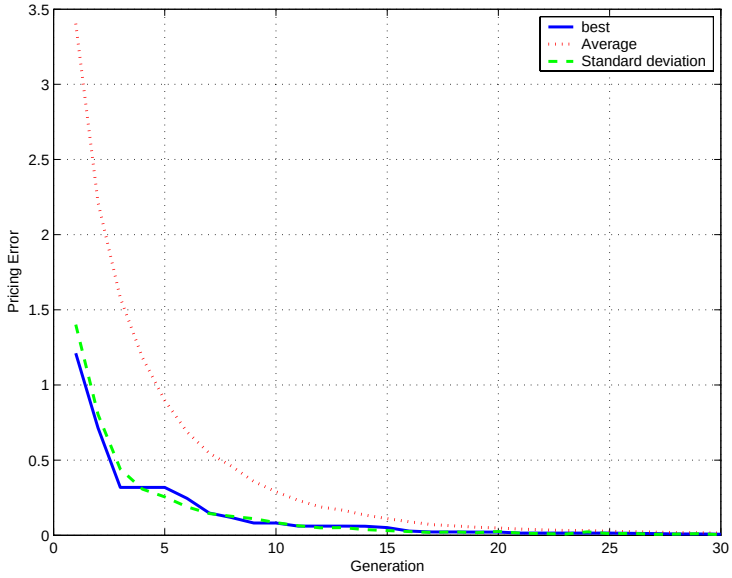

Figure 5: Numerical test: population mean, standard deviation of fitness and fitness of best individual.

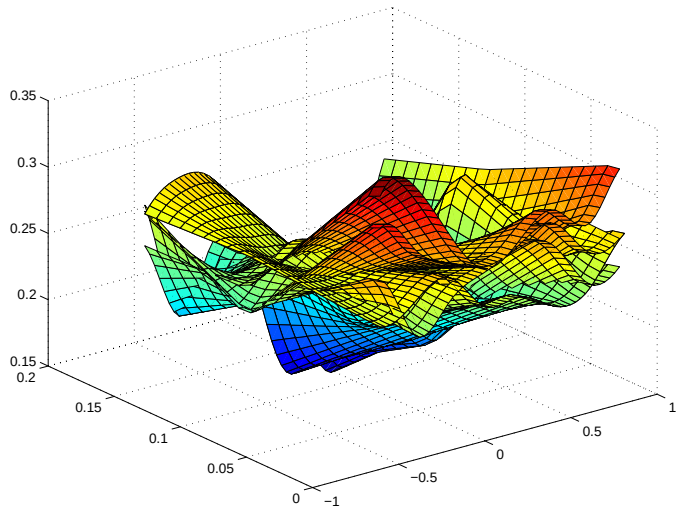

Figure 6: Local volatility functions retrieved from option prices generated by $\sigma_0(.,.)$.

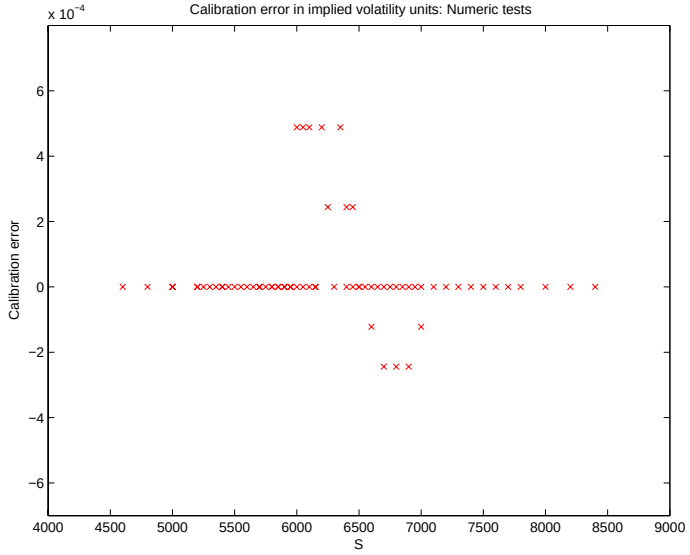

Figure 7: Numerical test: retrieval error in terms of implied volatility. Each point represents the calibration error in implied volatility units for one option in the data set. Scale: $10^{-4} = 1$ basis point.

too large to be useful in our context. The weight w_i was chose to be the inverse of the Black-Scholes Vega for the i-th option. In this way, i-th error term in (23) is, up to first order in the pricing error, equivalent to the error in implied volatility units.

The procedure described in Section 2 was applied to this data set to generate a population of 50 candidate solutions, which were then evolved using an evolutionary algorithm. Figure 4 illustrates the convergence of the algorithm: contrarily to the numerical tests shown above where the model was known to be well specified, here convergence is not monotonous, indicating that the minimization is more difficult. The a priori error defined in (17) is given in this case by $\delta \simeq 40$: again, we observe that the level set $\{\theta \in E, G(x) \leq \delta\}$ has been attained. Note that fitness in the population is fairly homogeneous as indicated by a standard deviation of $G(\theta^i)$ much lower than its population average. This means that most of the models in the population have a similar (and good) calibration performance. It is therefore interesting to see if they correspond to similar volatility functions. Figure 10 shows two of the best performing volatility functions obtained. As seen in the figure, they have different term structures, distinguishable by the eye. However, their calibration performance, as indicated by the absolute pricing error (23), is quite similar. Figure 11 gives the decomposition of the pricing errors for these two local volatility models: each point on the graphs corresponds to the calibration error, in implied volatility units, for one of the options in the data set. Since there is a one-to-one map between local

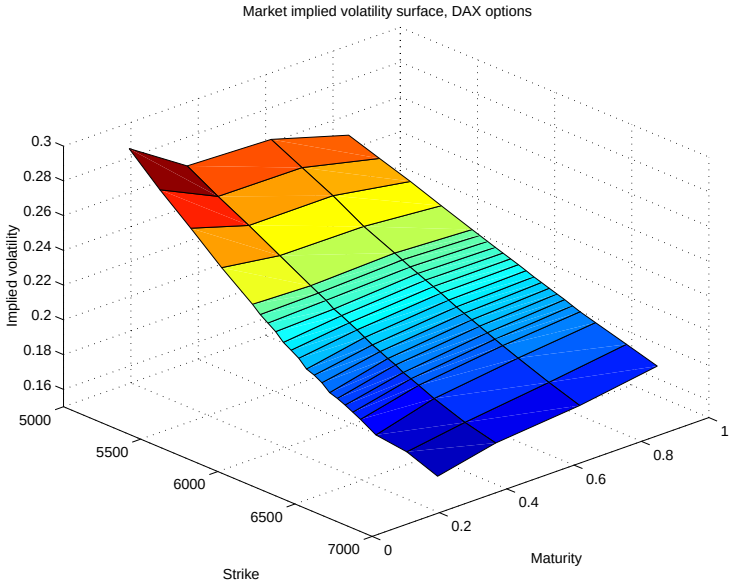

Figure 8: DAX options implied volatilities: 13 June 2001.

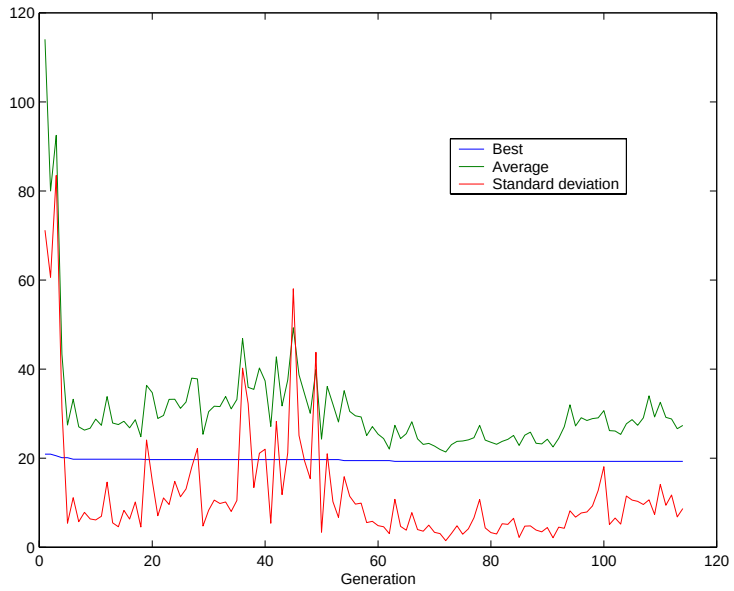

Figure 9: Evolution of pricing error.

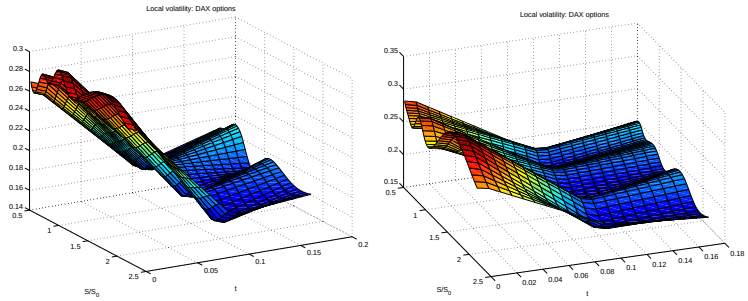

Figure 10: Local volatility surfaces: DAX options.

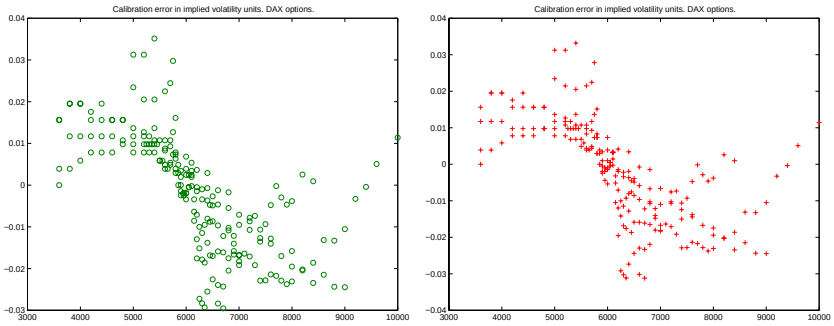

Figure 11: Calibration error in implied volatility units: DAX options.

volatility functions and call option prices [Dupire 1994], if *all* strikes and maturities had been included in the data sets we would have been able to distinguish these two models; however, as can be observed from figure 11, the two diffusion models display quite similar patterns of implied volatility for the quoted strikes and maturities. Other local volatility functions, compatible with the same data set of call options, are given in figures 12–14. Recall that the final population has a relatively small standard deviation of fitness do the calibration performance of the corresponding diffusion models are quite similar, even though the actual values of local volatilities and their term structures differ.

Figure 15 regroups all these candidate solutions on the same graph: they can be regarded is a sample drawn from the set of calibrated local volatility functions. What is striking is the pattern of their dispersion, which decreases notably as a function of maturity. For the sake of clarity we have represented the pointwise upper and lower envelopes of these surfaces in figure 16: the widening of this band as maturity decreases reflects the parabolic nature of the inverse problem. In financial terms, it simply means that short term options are not affected much by the value of the underlying's (local) volatility so their price cannot give much information on short term volatility, which therefore leads to a large uncertainty on short term volatility. Conversely, longer term options

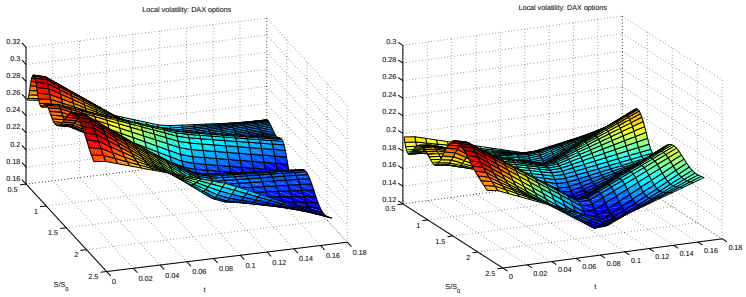

Figure 12: Local volatility surfaces: DAX options.

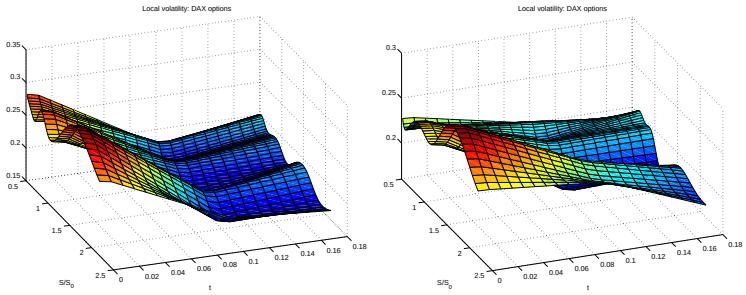

Figure 13: Local volatility surfaces: DAX options.

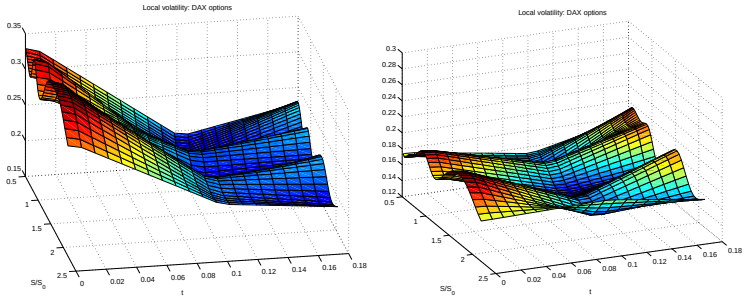

Figure 14: Local volatility surfaces: DAX options.

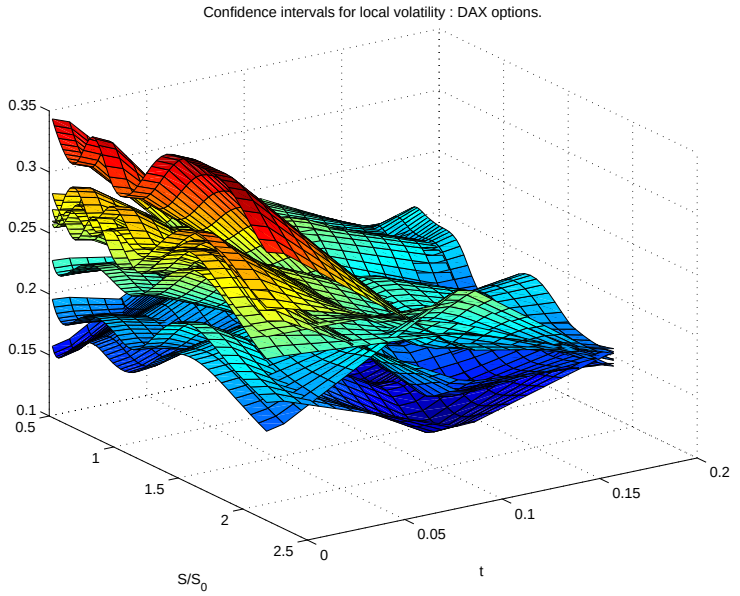

Figure 15: Uncertainty on local volatility surfaces: DAX options.

are strongly affected by the value of volatility so one can use them to get more precise estimates for longer term volatility.

Similar results were obtained on data sets of DAX options from other dates in our data set. These examples illustrate that the possibility of being able to reconstruct local volatility from call option prices in a precise manner is at best illusory: even in the case of index options where data sets are relatively large, the parameter uncertainty which prevails is too important to be ignored. They also show that this uncertainty prevails even more for short maturities: in our example, short term volatility hovers anywhere between 15% and 35%! These observations casts a doubt on the information content of short term options in terms of volatility and questions the basis of short maturity asymptotics as a method for exploring volatility patterns.

5 Applications and extensions

We have described a new approach to the estimation (risk-neutral) dynamics of an underlying asset from cross-sectional observations of option prices. Our approach is based on an evolutionary algorithm in which a population of optimizers performs a random search in the parameter space of the model, evolving through cycles of random mutations followed by selection. We give conditions under which the algorithm converges to a sample of models calibrated to the market prices with a given precision.

The proposed algorithm takes as input market prices for a set of benchmark

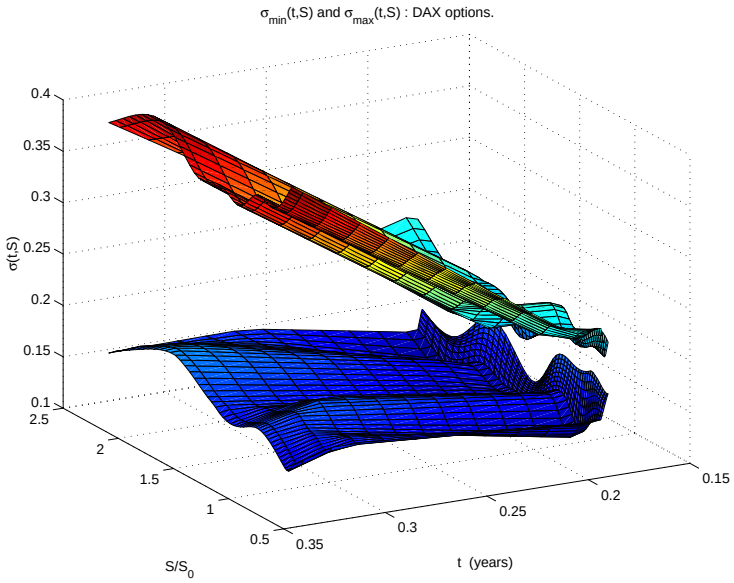

Figure 16: Pointwise lower and upper bounds for the local volatility $\sigma_{\min}(t, S)$ and $\sigma_{\max}(t, S)$ computed from 10 best solutions, DAX options.

options and produces a family of models which are marked-to-market on these options. We have given a detailed example of how this algorithm can be implemented in the case of diffusion models, where the aim is to reconstruct the local volatility surface from prices of call options and we have illustrated its implementability by applying it to an empirical data set of index options.

Evolutionary algorithms require neither differentiability nor convexity of the objective function to be minimized and hence allow a wide range of fitting criteria to be used. They avoid computation of gradients, which are the main computational burden in high-dimensional optimization problems typical of non-parametric calibration methods. In fact, their only requirement is being able to price the options in the calibration set which makes them easy to adapt to a wide variety of models and payoffs.

5.1 Quantifying model uncertainty

Apart from the numerical advantages detailed above, it also yields, as by-product, a way to analyze model uncertainty. Calibration algorithms based on deterministic optimization yield a *point estimate* for model parameters: they point out to *one* model in the class of models considered (here, the Markovian diffusion class) which approximates well the observed prices of benchmark options. By contrast, an evolutionary approach yields an entire *population* of solutions to the inverse problem, many of which price the benchmark options with equivalent precision. The heterogeneity of this population reflects the

uncertainty in model parameters, which are left undetermined by the benchmark options. This idea can be exploited to produce a quantitative measure of model uncertainty compatible with observed market prices of benchmark instruments [Cont (2003)], in the following manner. Suppose that we have calibrated a model $(\mathbb{Q}_\theta)_{\theta \in E}$ to the benchmark payoffs H_i with market prices $C_i^*, i = 1..I$ using the evolutionary algorithm outlined above. This yields a population of parameter values $\theta_1, .., \theta_k$ corresponding to calibrated models. Denote by $\mathcal{Q} = \{\mathbb{Q}_{\theta_1}, ..., \mathbb{Q}_{\theta_1}\}$ the associated pricing rules. Then for $\mathbb{Q} \in \mathcal{Q}$ we have $E^{\mathbb{Q}}(H_i) \simeq C_i^*$ within bid-ask bounds. Consider now an exotic or illiquid option on the same underlying, with payoff X. Define the upper and lower price bounds by:

$$\overline{\pi}(X) = \sup_{\mathbb{Q} \in \mathcal{Q}} E^{\mathbb{Q}}[X] \qquad \underline{\pi}(X) = \inf_{\mathbb{Q} \in \mathcal{Q}} E^{\mathbb{Q}}[X] = -\overline{\pi}(-X) \tag{37}$$

then $\overline{\pi}(X)$ is a coherent risk measure [Artzner et al(1999)], compatible with the market prices in the sense that for any of the traded options with payoffs $H_i, i = 1..I$, $\overline{\pi}(H_i) \simeq C_i^*$ within bid-ask bounds. This is in contrast with superreplication costs associated with uncertainty in volatility, which are typically way out of market bid-ask bounds. Furthermore

$$\mu(X) \quad = \quad \overline{\pi}(X) - \underline{\pi}(X) = \overline{\pi}(X) + \overline{\pi}(-X) \tag{38}$$

quantifies the impact of model uncertainty on the value of the derivative X. Once the family $\mathcal{Q}$ is obtained from the evolutionary calibration procedure, computation of $\overline{\pi}(X), \underline{\pi}(X)$ and $\mu(X)$ simply amounts to pricing the claim X in each of the models; moreover, the model realizing the supremum can be identified as the "worst case" model for the claim X. Therefore, our model calibration procedure enables to quantify the uncertainty on values of contingent claims associated with parameter uncertainty, without much further computational effort. Properties of such measures of model uncertainty are further discussed in [Cont (2003)].

5.2 Extensions

As noted above, since the only ingredient necessary in the calibration algorithm is the computation of option prices, one can also include in this setting prices of options other than European calls or puts: American options, barrier options or any other available exotic options can also be included in the calibration set as long as an efficient numerical algorithm is available for pricing them in the model. To our knowledge, this is not feasible using alternative calibration algorithms.

On the implementation side, the methodology described in section 1 is not specific to diffusion models: the same approach can be used for other families of models such as stochastic volatility models, interest rate models, multivariate models and models with jumps. We believe these extensions are of interest for applications and will be the object of our future work.

References

[Achdou & Pironneau] Achdou, Y. & Pironneau, O. (2002) Volatility calibration by multilevel least squares, *International Journal of Theoretical and Applied Finance*, Vol. 5, No. 6, 619–643.

[Ait Sahalia & Lo (1998)] Ait Sahalia, Y. and Lo, A. (1998) Nonparametric Estimation of State-Price Densities Implicit in Financial Asset Prices, *Journal of Finance*, Vol. 53, 499-547.

[Andersen & Andreasen (2000)] Andersen & Andreasen (2000) Jump diffusion models: volatility smile fitting and numerical methods for pricing, *Review of Derivatives Research*, 4, 231262.

[Artzner et al(1999)] Artzner P., Delbaen F., Eber J.M., Heath D. (1999) Coherent measures of risk, *Mathematical Finance*, **9**, No. 3, pp 203–228.

[Avellaneda et al (1997)] Avellaneda M., Friedman C, Holmes & D. Samperi (1997) Calibrating volatility surfaces via relative entropy minimization. *Applied Mathematical Finance*, March 1997.

[Bäck] T. Bäck (1995) Evolutionary Algorithms in theory and practice, Oxford University Press.

[BenHamida & Petrowski] Ben Hamida, S. and A. Petrowski (2000) The need for improving the exploration operators for constrained problems, Congress on Evolutionary Computation, pp 1176-1183, Congress on Evolutionary Computing 2000, Springer.

[Brigo & Mercurio] Brigo, D. & Mercurio, F. (2002) Lognormal mixture dynamics and calibration to market volatility smiles, *International Journal of Theoretical and Applied Finance*, Vol. 5, No. 4, 427–446.

[Cerf (1996)] Cerf, R. (1996) The dynamics of mutation-selection algorithms with large population sizes, *Annales de l'Institut Henri Poincaré*, **32** (4), 455–508.

[Cerf (1998)] Cerf, R. (1998) Asymptotic convergence of genetic algorithms,*Advances in Applied Probability*, **30** (2), 521–550.

[Coleman et al (1999)] Coleman T., Li & Verma, A. (1999): Reconstructing the unknown volatility function, *Journal of Computational Finance*, Vol. 2, Number 3, 1999, 77-102.

[Cont (2003)] Cont, R. (2003) Model uncertainty and its impact on derivative instruments, Working Paper.

[Cont & Tankov (2004)] Cont, R. & Tankov, P. (2004): Nonparametric calibration of jump-diffusion option pricing models, *Journal of Computational Finance*, Vol. 7, No. 3, 1–49.

[Crépey (2003)] Crépey, S. (2003) Calibration of the local volatility in a trinomial tree using Tikhonov regularization, *Inverse problems*, 19, 91–127.

[Del Moral (2004)] Del Moral. P. (2004): Feynman-Kac formulae: genealogical and interacting particle systems with applications, Berlin: Springer.

[Del Moral & Miclo (2000)] P. Del Moral & L. Miclo (2000): Branching and interacting particle system approximations of Feynman-Kac formulae with applications to nonlinear filtering, *Séminaire de Probabilités XXXIV*, Lecture Notes in Mathematics 1729, 1–145, Berlin: Springer.

[Del Moral & Miclo (2001)] P. Del Moral and L. Miclo (2001): Asymptotic results for genetic algorithms with applications to nonlinear estimation. in: Kallel et al. (Eds.) Theoretical aspects of evolutionary computing, pp 439–493, Nat. Comput. Ser., Springer, Berlin.

[Del Moral & Miclo (2003)] P. Del Moral and L. Miclo (2003): Annealed Feyman-Kac models, *Communications in mathematical physics*, Vol. 235, No. 2, 191–214.

[Del Moral et al. (2001)] Del Moral P., Kouritzin, M. and Miclo, L. (2001): On a class of discrete-generation interacting particle systems, *Electronic Journal of Probability*, Vol. 6 (16), 1–26.

[Derman et al (1996)] Emanuel Derman, Iraj Kani and Neil Chriss: Implied Trinomial Trees of the Volatility Smile, The Journal of Derivatives, Summer 1996.

[Dupire 1994] Dupire, B. (1994) Pricing with a smile, *RISK*, Vol. 7, January, 18–20.

[Heston] Heston, S. (1993) A closed-form solution for options with stochastic volatility with applications to bond and currency options, *Review of Financial Studies*, Vol. 6, pp. 327–343.

[Holland] Holland, J.H. (1975) *Adaptation in natural and artificial systems*, Ann Arbor: University of Michigan Press.

[Jackson et al (1999)] Jackson, Süli, E. & Howison, S. (1999) Computation of deterministic volatility surfaces, *Journal of Computational Finance*, Vol. 2, Number 2, 1998-99, 5-32

[Jacquier & Jarrow (2000)] Jacquier, E. & Jarrow, R. (2000): Bayesian analysis of contingent claim model error, *Journal of Econometrics*, Vol. 94, 145–180.

[Kallel et al.] Kallel L., Naudts, B. and Rogers, A. (Eds.): Theoretical aspects of evolutionary computing, Nat. Comput. Ser., Springer-Verlag, Berlin, 2001.

[Ladyzhenskaya et al.] Ladyzhenskaya O.A., Solonnikov V.A. & Uralceva (1968) *Linear and Quasilinear equations of parabolic type*, English Translation, AMS.

[Lagnado & Osher (1997)] Lagnado & Osher (1997): A technique for calibrating derivative security pricing models: numerical solution of an inverse problem. *Journal of Computational Finance*, 1(1), Fall 1997.

[Lo (1986)] Lo, Andrew (1986) Statistical tests of contingent claims asset-pricing models, Journal of Financial Economics, 143–173.

[Löwe (1996)] Löwe, Matthias (1996) On the convergence of genetic algorithms, Exposition. Math. 14, no. 4, 289–312.

[Madan & Milne (1991)] Madan, D. and Milne, F. (1991) *Option pricing with variance gamma martingale components*, Math. Finance, 1 (1991), pp. 39–55.

[Mosegaard & Tarantola (1995)] Mosegaard & Tarantola (1995): Monte Carlo sampling of solutions to inverse problems, *Journal of Geophysical Research*, Vol. 100, No.B7, 431–447.

[Nix & Vose (1991)] Nix, A. & Vose, M.D. (1991): Modelling genetic algorithms with Markov chains, *Annals of Mathematics and Artificial Intelligence*, Vol. 5, 79–88.

[Samperi (2002)] Samperi, D. (2002) Calibrating a diffusion model with uncertain volatility, *Mathematical Finance*, 12, 71–87.

[Schwefel (1981)] H.-P. Schwefel (1981) Numerical Optimization of Computer Models, John Wiley & Sons.

A B splines

Define the functions:

$$\psi_0(x) = \frac{1}{6}(1-x)^3 \tag{39}$$

$$\psi_1(x) = \frac{1}{6}(4-6x^2+3x^3) \tag{40}$$

$$\psi_2(x) = \frac{1}{6}(1+3x+3x^2-3x^3) \tag{41}$$

$$\psi_3(x) = \frac{x^3}{6} \tag{42}$$

Given a set of nodes $X_0 < X_1 < ... < X_n$ the functions $(\psi_i, i = 0..3)$ may be translated and dilated appropriately by substituting for x the new variable

$$\xi_{[X_{i-1},X_i]} = \frac{x - X_{i-1}}{X_i - X_{i-1}} \tag{43}$$

The *B-spline* basis $(\phi_i, i = 0..n)$ associated with a set of nodes $X_0 < X_1 < ... < X_n$ is defined by:

$$\phi_i(x) = \begin{cases} 0 & x \leq X_{i-2} \\ \psi_3(\xi_{[X_{i-2},X_{i-1}]}(x)) & X_{i-1} \geq x \geq X_{i-2} \\ \psi_2(\xi_{[X_{i-1},X_i]}(x)) & X_i \geq x \geq X_{i-1} \\ \psi_1(\xi_{[X_i,X_{i+1}]}(x)) & X_{i+1} \geq x \geq X_i \\ \psi_0(\xi_{[X_{i+1},X_{i+2}]}(x)) & X_{i+2} \geq x \geq X_{i+1} \\ 0 & x \geq X_{i+2} \end{cases}$$

Smooth functions may then be built by using linear combinations of B-splines:

$$f(x) \quad = \quad \sum_{i=1}^{n} f_i \quad \phi_i(x) \tag{44}$$

Due to the smoothness properties of B-splines, f is smooth (C^2). Moreover since ϕ_is have compact support the "zone of influence" of f_i on f is limited to $[X_{i-2}, X_{i+2}]$. Finally, since $\phi_i \geq 0$ the expansions enables to construct positive functions by simply imposing that $f_i \geq 0$. It is important to note that the B-spline expansion (44) is not an interpolation of f at points (X_i, f_i): $f(X_i) \neq f_i$.

B Proof of Proposition 2

The mapping $\theta \mapsto \sigma_\theta$ given by (24)-(25) is a continuous mapping from $E \subset \mathbb{R}^d$ to $H \subset H^{1,2}([0,T] \times \mathbb{R})$. Also, from general results on continuity of solutions of parabolic PDEs with respect to coefficients [Ladyzhenskaya et al.], the mapping $\sigma \mapsto C^\sigma(.,.;T_i,K_i)$ is a continuous mapping from $H^{1,2}([0,T] \times \mathbb{R})$ to $C^0([0,T] \times \mathbb{R})$. Therefore, by composition of continuous maps

$$\begin{aligned} G : E \quad &\mapsto \quad \mathbb{R}^+ \\ \theta \quad &\mapsto \quad C^{\sigma_\theta}(t, S; T_i, K_i) \end{aligned}$$

is continuous. Using standard arbitrage bounds on call prices,

$$\forall \theta \in E, 0 \leq G(\theta) \leq 2S_0 \sum_{i=1}^{I} w_i \leq 200 I S_0$$

so $G(.)$ is bounded, therefore of bounded oscillation. Furthermore, it is easily verified that since E is bounded, the mutation kernel defined by (36) verifies the mixing condition (14). Thus, the ingredients of the evolution scheme verify the hypotheses of Proposition 1, thus entailing the convergence of the population to the level set G^*:

$$\forall \epsilon > 0, \qquad \mu_n(G(\theta) \geq G^* + \epsilon) \ \rightarrow 0 \text{ as } n \rightarrow \infty.$$

where G^* is defined by (15). From (36) it is verified that M is an irreducible kernel. Furthermore, from (36) we see that

$$\sup_{x \in E} M(x, x) \quad < \quad 1 \tag{45}$$

Denote by π the (unique) invariant distribution of M: π verifies $\int_E \pi(dx)M(x,dy) = \pi(dy)$. Since the kernel M defined by (36) verifies $M(x,x) > 0$ for any $x \in E$, by [Del Moral & Miclo (2003), Proposition 4.4] G^* coincides with the essential infimum of G with respect to π. Let us show that π is absolutely continuous with respect to the Lebesgue measure on E. Consider a subset $U \subset E$ with Lebesgue measure 0. Using the expression of the mutation kernel (36), we have

$$\begin{aligned}
\pi(U) &= \int_E \pi(dx)M(x,U) \\
&= \int_U \pi(dx)M(x,x) + \int \pi(dx) \int_{y\in U} \frac{\exp[-\frac{1}{2}{}^t(y-x)A^{-1}(y-x)]}{\sqrt{(2\pi)^d|A|}} dy \\
&= \int_U \pi(dx)M(x,x) + 0 \qquad (46)
\end{aligned}$$

since U has measure zero. Therefore using (45) we obtain

$$\pi(U) \leq \sup_{x\in E} M(x,x)\pi(U) < \pi(U) \qquad (47)$$

which entails that $\pi(U) = 0$: π is thus absolutely continuous with respect to the Lebesge measure. Therefore, the essential infimum of G with respect to π coincides with its global minimum, which gives the result.