

Combinatorial Implications of Nonlinear Uncertain Volatility Models: the Case of Barrier Options

Marco Avellaneda
New York University

Robert Buff*
New York University

August 15, 1998

Abstract

Extensions to the Black-Scholes model have been suggested recently that permit to calculate worst-case prices for a portfolio of vanilla options or for exotic options when no a priori distribution for the forward volatility is known. The Uncertain Volatility Model (UVM) by Avellaneda and Parás finds a one-sided worst-case volatility scenario for the buy resp. sell side within a specified volatility range. A key feature of this approach is the possibility of hedging with options: risk cancellation leads to super-resp. sub-additive portfolio values. This nonlinear behavior causes the combinatorial complexity of the pricing problem to increase significantly in the case of barrier options. In this paper, we show that for a portfolio P of n barrier options and any number of vanilla options, the number of PDEs that have to be solved in a hierarchical manner in order to solve the UVM problem for P is bounded by $O(n^2)$. We discuss how a numerically stable implementation can be achieved and give numerical results.

1 Introduction

This paper discusses combinatorial and numerical aspects of an options pricing system that follows the Uncertain Volatility Model of Avellaneda *et al.* (1995). The system can handle any portfolio consisting of vanilla, single and double barrier options, both with linear and digital payoff functions. It prices these options and produces hedges that combine vanilla options with cash instruments (delta-hedging). An online version of the pricer can be accessed on the WWW under “<http://home.cs.nyu.edu:8080/cgi-bin/vop>”.

The Uncertain Volatility Model (UVM) is an extension of the Black-Scholes framework that incorporates uncertainty in the volatility of the underlying asset

*We are greatly indebted to Richard Holmes for pointing out a number of inconsistencies in early stages of the project.

in the pricing and hedging of derivative securities. In UVM, no statistical distribution for the stochastic volatility is specified; rather, a worst-case volatility scenario is derived for the liability structure under consideration.

The theoretical values of portfolios of derivative securities generated by UVM are super-additive for the buy side and sub-additive for the sell side, due to diversification of volatility risk and “gamma-risk.” This effect is achieved through a nonlinear Hamilton-Jacobi-Bellman equation that generalizes Black-Scholes by adjusting the local volatility, or conditional variance, to the Gamma of the position.

The nonlinearity in the UVM equation leads to negligible computational overhead if the portfolio under consideration contains only vanilla options. Once path-dependent instruments are added, however, the combinatorial complexity can increase quite dramatically, as the past history of an instrument affects not only the instrument itself, but the valuation of the entire portfolio.

Indeed, as outlined in Avellaneda and Parás (1996), the evaluation of the worst-case scenario for a short position in a knockout option hedged with vanilla options requires pricing

- the worst-case scenario price for the vanilla options hedge *without* the barrier option; and
- the worst-case scenario for the entire portfolio (vanillas and barrier).

The reason for this is that the boundary conditions required for evaluating the portfolio at the barrier(s) depend on the worst-case pricing of the vanilla options used in the hedge at that price level (i.e. after the knockout option disappears). The combinatorial complexity increases in the presence of several knockout options because we have to analyze different scenarios, corresponding to one option being alive and the others not, or vice-versa. The number of possible scenarios increases with the number of barrier options considered and hence the number of PDEs that need to be solved.

This paper studies the combinatorial complexity of the pricing algorithm in the case of many barrier options. We show that the computation time needed for portfolios of barrier options can increase by a factor of $O(n^2)$ compared to the vanilla case (here, n is the number of barrier options). Moreover, we describe numerical techniques which lead to a stable and accurate explicit finite difference scheme, in which all barriers are made to coincide with lattice levels (this scheme is implemented in the WWW pricer VOP mentioned earlier).

The main application of the algorithms described is to the pricing and hedging of barrier options under volatility uncertainty. There is an extensive literature on barrier options pricing, including the works of Rubinstein and Reiner (1991), Conze and Viswanathan (1991), Kunitomo and Ikeda (1992), Geman and Yor (1996) and Roberts and Shortland (1997). It is noteworthy that few publications so far have addressed the question of volatility uncertainty in hedging exotics. Perhaps, one notable exception are the works on static hedging of barrier options by Carr and Chou (1997), and Carr *et al.* (1997).

The remainder of the paper is organized as follows. The Uncertain Volatility Model is reviewed in Sect. 2. In Sect. 3, combinatorial issues are discussed and upper bounds on the algorithmic complexity are derived. Section 4 is devoted to numerical issues (barrier alignment and stability). Section 5 describes implementation aspects and experimental results. Section 6 concludes the paper.

2 Review of the Uncertain Volatility Model (UVM)

Following Avellaneda *et al.* (1995), we assume that the spot price S_t of the underlying asset follows a diffusion with non-constant coefficients

$$\frac{dS}{S} = \sigma(S, t)dZ + \mu(S, t)dt \quad (2.1)$$

The spot volatility function $\sigma(S, t)$ is known to fluctuate within a band

$$0 < \sigma_{\min} \leq \sigma(S, t) \leq \sigma_{\max} \quad (2.2)$$

for $0 < S < \infty$ and $0 \leq t \leq T$, where T represents some distant time horizon. The spot drift function $\mu(S, t)$ is known a priori at time $t = 0$ and satisfies

$$\mu_{\min} \leq \mu(S, t) \leq \mu_{\max} \quad (2.3)$$

for some constants μ_{\min} and μ_{\max} . In most cases $\mu(S, t) = r_t - d_t$, where r_t and d_t are the spot riskless rate and the dividend rate of the underlying, respectively. The present day price of the underlying is denoted by S_0 .

Each volatility process $\{\sigma(S, t)\}$ that satisfies (2.2) induces together with $\mu(S, t)$ a unique probability measure $\mathbf{P} = \mathbf{P}(\{\sigma(S, t)\}, \{\mu(S, t)\})$ on the price paths $\{S_t\}$. Let \mathbb{P} denote the set of all measures that can be induced within the constraints (2.2).

2.1 Portfolios of vanilla options

Consider a portfolio P of n vanilla options with expiration dates $t_1 \leq t_2 \leq \dots \leq t_n \leq T$ and payoff functions $F_1(S_{t_1}), F_2(S_{t_2}), \dots, F_n(S_{t_n})$. Avellaneda *et al.* (1995) show that the present day worst-case scenario estimate for the buy side is the value $V(S_0, 0)$ of the function

$$V(S_t, t) = \sup_{\mathbf{P} \in \mathbb{P}} \mathbf{E}^{\mathbf{P}} \left\{ \sum_{i: t_i \geq t}^n e^{-\int_t^{t_i} r_t dt} F_i(S_{t_i}) \right\} \quad (2.4)$$

where $\mathbf{E}^{\mathbf{P}}$ is the expectation operator with respect to the measure \mathbf{P} and the spot price process (2.1).

In (2.4), the *pricing equation* for P under UVM is defined. It can be shown that V can be computed by solving the nonlinear partial differential equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}S^2 \Sigma^2 \left\{ \frac{\partial^2 V}{\partial S^2} \right\} \frac{\partial^2 V}{\partial S^2} + \mu S \frac{\partial V}{\partial S} = r_t V \quad (2.5)$$

where

$$\Sigma^2\{C\} = \begin{cases} \sigma_{\max}^2 & \text{if } C \geq 0 \\ \sigma_{\min}^2 & \text{if } CX < 0 \end{cases} \quad (2.6)$$

In Sect. 4, an explicit finite-difference solver will be described that is capable of solving a hierarchy of pricing equations simultaneously on the same lattice.

2.2 Adding barrier options

The situation becomes more complex if vanilla and barrier options are combined in a portfolio, because once one or several of the barrier options have knocked out¹, the holder of the portfolio is left with a reduced portfolio $P' \subset P$ whose exposure to volatility risk can differ a great deal from the exposure to the volatility risk associated with P .

To illustrate this point, assume that the k th option in portfolio P is an up-and-out single barrier option with constant barrier $b > S_0$. The worst-case residual liability of the holder of P along the barrier at time t is then

$$V(S = b, t) = \sup_{\mathbf{P} \in \mathbb{P}} \mathbf{E}^{\mathbf{P}} \left\{ \sum_{\substack{i: t_i \geq t \\ i \neq k}}^n e^{-\int_t^{t_i} r_t dt} F_i(S_{t_i}) \right\} \quad (2.7)$$

where \mathbb{P} is the set of the probability measures of all spot price paths starting at $(S = b, t)$ and induced by (2.1) and (2.2).

This pricing equation is different from the original pricing equation for P and must be solved separately. Its solution is used as boundary data for P 's pricing equation, which therefore turns into a boundary value problem.

If P contains more than one barrier option, this procedure repeats itself in a hierarchical manner. The number of pricing equations that have to be solved can grow quadratically in the number of barrier options, as examples in Sect. 3 will show. The reader may verify that, for instance, a portfolio consisting of one up-and-out barrier option and one down-and-out barrier option demands the solution of three separate pricing equations.

3 Combinatorial issues

The combinatorial complexity will be examined more closely in the following paragraphs. We will show that the maximum number of pricing equations that

¹We discuss only knockout barrier options. Barrier options of many other types can in principal be replicated by suitable combinations of vanilla and knockout barrier options.

needs to be solved in order to arrive at the UVM value of a portfolio with n barrier options is $O(n^2)$. This is much smaller than the naive estimate $2^n - 1$, that obtains by considering all subsets of the portfolio. This improvement is due to the fact that the barriers must be crossed in a specific way. For example, if the portfolio contains only up-and-out barrier options with the same expiration date, then the total number of portfolios that need to be evaluated is at most n . The consideration of double barrier options or up-and-out options combined with down-and-out options makes the analysis (and the algorithm) more complex.

3.1 Problem specification

Given is a portfolio of $n \geq 1$ options, each characterized by an up-and-out barrier u_i and a down-and-out barrier d_i , $1 \leq i \leq n$. Let L be a very large number. Vanilla options are modeled by setting $d_i = 0$ and $u_i = L$. For a single up-and-out barrier option with barrier y , $d_i = 0$ and $u_i = y$. For a single down-and-out barrier option with barrier z , $d_i = z$ and $u_i = L$. For double barrier options, both d_i and u_i are set to the respective barriers.²

Index sets, denoted by the letters P and Q and also called “portfolios”, represent subsets of the original portfolio of n instruments. The index set $\mathcal{P} = \{1, \dots, n\}$ stands for the original portfolio. The empty portfolio is simply denoted by \emptyset .

The barriers of each instrument are constant throughout the life of the instrument. We also assume that all instruments have the same expiration date T (this restriction can later easily be removed). The present day spot price of the underlying asset is denoted by S_0 , and the following condition must hold:

$$d_i < S_0 < u_i \quad (i \in \mathcal{P}) \quad (3.1)$$

This condition is straightforward, as every instrument that violates it is effectively worthless today.

The open price interval in which the i th instrument is possibly alive is denoted by $a_i = (d_i, u_i)$, $i \in \mathcal{P}$. Let $\emptyset \neq P \subseteq \mathcal{P}$ be a portfolio. Define $A_P = \bigcap_{i \in P} a_i$, $U_P = \sup A_P$, $D_P = \inf A_P$. It is easy to see that A_P is also an open interval. Furthermore, $U_P = \min_{i \in P} u_i$, $D_P = \max_{i \in P} d_i$, and (3.1) holds for U_P and D_P .

Let $B_U(P) = \{i \in P \mid u_i \neq U_P\}$ be the *upper extension* of P . Correspondingly, let $B_D(P) = \{i \in P \mid d_i \neq D_P\}$ be the *lower extension* of P . $B_U(P)$ resp. $B_D(P)$ indicate which instruments in P remain possibly alive when the price of the underlying asset crosses U_P resp. D_P , and are themselves portfolios (possibly empty). Figure 1 provides an example for $n = 4$.

$Q_1 = B_U(P)$ and $Q_2 = B_D(P)$ lead to the boundary conditions that are required when the portfolio P is evaluated in $[0, T] \times [D_P, U_P] \in \mathbb{R}^2$. In a

² L plays the role of ∞ and is needed only for symbolic purposes. It has to be bigger than any barrier realized in any instrument in the market. d_i can be safely set to 0 if there is no down-and-out-barrier. 0 also plays merely a symbolic role and could be replaced by an arbitrary negative number.

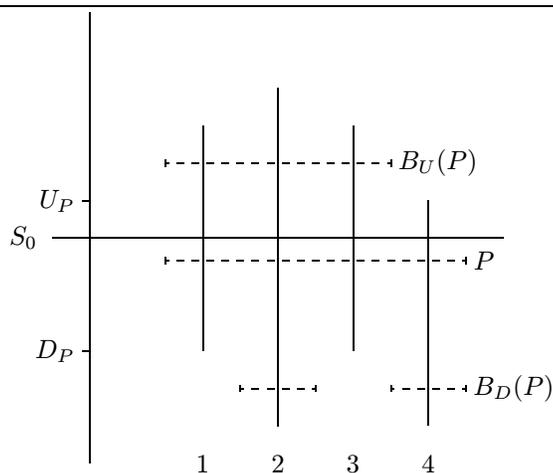


Figure 1: The relationship between a portfolio P of n options and its upper resp. lower extensions, $B_U(P)$ resp. $B_D(P)$. The vertical axis marks S , the price of the underlying. U_P is the smallest up-and-out barrier among the up-and-out barriers in P . Similarly, D_P is the smallest down-and-out barrier among the down-and-out barriers in P . Each option i is represented by a vertical bar whose endpoints mark its barriers u_i and d_i

recursive manner, the solution of the pricing equation for P depends on the solutions of separate pricing equations for both Q_1 and Q_2 , respectively.

3.2 The hierarchy of pricing equations

Let \mathcal{P} be a portfolio. It is clear that the pricing equation has to be solved for \mathcal{P} .

Definition 3.1. For a portfolio $\mathcal{P} \neq \emptyset$, the hierarchy $\mathcal{H} = \mathcal{H}(\mathcal{P})$ of \mathcal{P} is defined to be the smallest subset of $2^{\mathcal{P}}$ such that

- $\mathcal{P} \in \mathcal{H}$, and
- $P \in \mathcal{H}$ implies $Q \in \mathcal{H}$ for any nonempty $Q \in \{B_U(P), B_D(P)\}$.

Clearly, $\emptyset \notin \mathcal{H}$. We will make statements about the size of \mathcal{H} in Sect. 3.3. In Fig. 2, a high level sketch of the algorithm to find \mathcal{H} is given (a more efficient version is used in the actual implementation).

The fact that $B_D(P)$ resp. $B_U(P)$ are strict subsets of any $P \subseteq \mathcal{P}$ makes the algorithm to solve the hierarchy of pricing equations simply a matter of sorting \mathcal{H} based on the subset relation, as outlined in Fig. 3.

-
1. Set $\mathcal{H} := \{\mathcal{P}\}$
 2. Repeat the following:
 - (a) Set $\mathcal{H}' := \bigcup_{P \in \mathcal{H}} \{B_D(P), B_U(P)\}$
 - (b) Set $\mathcal{H}' := \mathcal{H}' \setminus (\mathcal{H} \cup \{\emptyset\})$
 - (c) Set $\mathcal{H} := \mathcal{H} \cup \mathcal{H}'$
- until $\mathcal{H}' = \emptyset$

Figure 2: The algorithm to find \mathcal{H} , the hierarchy of all portfolios whose pricing equations have to be solved in order to find the solution for the pricing equation of \mathcal{P}

3.3 The size of \mathcal{H}

Trivially, $1 \leq |\mathcal{H}| \leq 2^n - 1$. In the case of barrier options with constant barriers, however, a more precise (and satisfactory) upper bound can be derived.

Proposition 3.2. *Given a portfolio \mathcal{P} of $n \geq 1$ instruments such that*

- $u_1 > u_2 > \dots > u_n$,
- $1 \leq i, j \leq n$ and $i \neq j$ imply $d_i \neq d_j$, and
- condition (3.1) is fulfilled.

Then $|\mathcal{H}(\mathcal{P})| \leq n(n+1)/2$.

Proof. By induction over n . For $n = 1$, $|\mathcal{H}(\mathcal{P})| = 1$ by inspection and the proposition is fulfilled. Now let \mathcal{P} represent a portfolio of $n > 1$ instruments with the properties stated in the proposition. Then $\mathcal{Q} = \mathcal{P} \setminus \{n\}$ represents a portfolio of $n - 1$ instruments that also matches the properties stated in the proposition. See Fig. 4 for a diagram.

Clearly, $\mathcal{P} \in \mathcal{H}(\mathcal{P})$. By construction, $B_U(\mathcal{P}) = \mathcal{Q}$ which implies $\mathcal{Q} \in \mathcal{H}(\mathcal{P})$, and by transitivity $\mathcal{H}(\mathcal{Q}) \subset \mathcal{H}(\mathcal{P})$ (refer to the algorithm in Fig. 2). Now consider the sequence $B_0 = \mathcal{P}$, $B_1 = B_D(B_0)$, $B_2 = B_D(B_1)$, \dots , $B_{n-1} = B_D(B_{n-2})$, $B_n = \emptyset$. This sequence has $n + 1$ distinct elements, because by assumption the d_i 's are all distinct. For each B_i , $0 \leq i \leq n - 1$, we claim that

$$B_i \setminus \{n\} \in \mathcal{H}(\mathcal{Q}) \quad (3.2)$$

(note that n might or might not be in B_i). To see this choose $i_0 \in \{0, \dots, n-1\}$ such that $n \in B_i$ for $i \leq i_0$ and $n \notin B_i$ for $i > i_0$, and note that for $i < i_0$, $B_D(B_i \setminus \{n\}) = B_{i+1} \setminus \{n\}$ holds. Together with $B_0 \setminus \{n\} = \mathcal{Q}$, this implies that $B_1 \setminus \{n\} \in \mathcal{H}(\mathcal{Q})$ and recursively $B_i \setminus \{n\} \in \mathcal{H}(\mathcal{Q})$ for $0 \leq i \leq i_0$. Furthermore,

-
1. Set $k := |\mathcal{H}|$
 2. Find an ordering $P_{l_1}, P_{l_2}, \dots, P_{l_k}$ of the portfolios in \mathcal{H} such that for $i, j \in \{1, \dots, k\}$, $P_{l_i} \subset P_{l_j}$ implies $i < j$
 3. For $i = 1, \dots, k$ do the following:
 - (a) If $B_U(P_{l_i}) \neq \emptyset$ then retrieve the solution to $B_U(P_{l_i})$ (which has already been computed) and use it as the upper boundary condition
 - (b) If $B_D(P_{l_i}) \neq \emptyset$ then retrieve the solution to $B_D(P_{l_i})$ and use it as the lower boundary condition
 - (c) Solve the pricing equation for P_{l_i} , while using the data produced in the previous two steps if necessary

Figure 3: The algorithm to solve the pricing equations of all portfolios in \mathcal{H} in the correct order

$B_{i_0} \setminus \{n\} = B_{i_0+1}$. This and the fact that $B_i \setminus \{n\} = B_i$ for $i > i_0$ make (3.2) also true for $i_0 + 1 \leq i \leq n - 1$.

Since $B_U(B_i) = B_i \setminus \{n\}$ (u_n is the smallest up-and-out barrier) and therefore $B_U(B_i) \in \mathcal{H}(\mathcal{Q})$ for $0 \leq i \leq n - 1$ by (3.2), the set of portfolios that are not already covered by $\mathcal{H}(\mathcal{Q})$ contains exactly $\mathcal{P}, B_1, \dots, B_{i_0}$. Thus, with $i_0 \leq n - 1$, it follows that the size of $\mathcal{H}(\mathcal{P})$ is bounded by

$$|\mathcal{H}(\mathcal{P})| \leq |\mathcal{H}(\mathcal{Q})| + 1 + (n - 1) \quad (3.3)$$

or, by induction,

$$|\mathcal{H}(\mathcal{P})| \leq n(n - 1)/2 + 1 + (n - 1) = n(n + 1)/2 \quad (3.4)$$

which completes the proof. \square

In practical applications, the u_i and d_i need not be distinct. It is easy to see that these cases lead to smaller sizes of \mathcal{H} , making the upper bound derived in Prop. 3.2 the general worst case upper bound for every portfolio of vanilla, single and double barrier options of size n as characterized in Sect. 3.1. The following corollary shows that this upper bound is tight.

Corollary 3.3. *Let \mathcal{P} represent a portfolio of n double barrier options with barriers $u_1 > u_2 > \dots > u_n$ and $d_1 > d_2 > \dots > d_n$. Then $|\mathcal{H}(\mathcal{P})| = n(n + 1)/2$.*

Proof. By following the argument of Prop. 3.2. The sequence of lower extensions B_i containing n that have to be considered in each inductive step has always the maximum number of elements. \square

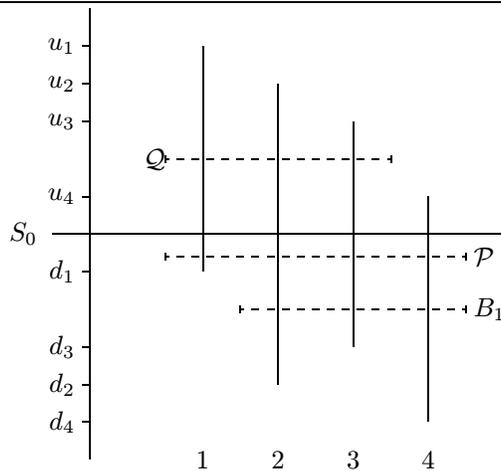


Figure 4: Portfolios \mathcal{P} and \mathcal{Q} for $n = 4$. Each instrument is represented by a vertical bar that marks the region of the price of the underlying asset in which it is possibly alive. B_1 , the lower extension of \mathcal{P} , is also marked. One can see that $B_U(B_1) = B_D(\mathcal{Q})$

Many interesting cases do not involve double barrier options. In the following, a specialization of Prop. 3.2 for single barrier options is derived.

Proposition 3.4. *Given a portfolio \mathcal{P} of $n \geq 1$ instruments such that $L = u_1 = u_2 = \dots = u_{n_d} > u_{n_d+1} > \dots > u_n > S_0$, and $S_0 > d_1 > d_2 > \dots > d_{n_d} > d_{n_d+1} = \dots = d_n = 0$ for some $n_d \in \{0, \dots, n\}$ (i.e., there are n_d down-and-out barrier options and $n_u = n - n_d$ up-and-out barrier options in \mathcal{P}). Then $|\mathcal{H}(\mathcal{P})| = n_d + n_u + n_d n_u$.*

Proof. For given n_d, n_u , define $f(n_d, n_u)$ to be the size of $\mathcal{H}(\mathcal{P})$ for some \mathcal{P} that represents a portfolio that fulfills the assumptions of the proposition, with n_d down-and-out barriers and n_u up-and-out barriers. It is easy to verify that $f(n_d, 0) = n_d$. By induction over n_u , additional properties of f are derived as follows.

Let \mathcal{P} represent a portfolio of n_d down-and-out barrier options and $n_u \geq 1$ up-and-out barrier options as stated in the proposition, with $n = n_d + n_u$. Then $\mathcal{Q} = \mathcal{P} \setminus \{n\}$ is a portfolio of n_d down-and-out barrier options and $n_u - 1$ up-and-out barrier options, also fulfilling the criteria of the proposition. Moreover, $\mathcal{H}(\mathcal{Q}) = f(n_d, n_u - 1)$. Figure 5 shows the situation for $n_d = n_u = 2$.

Since $B_U(\mathcal{P}) = \mathcal{Q}$, the upper extension of \mathcal{P} need not be considered as source of additional pricing equations to be solved. For $B_1 = B_D(\mathcal{P})$, $n \in B_1$ by construction. In general, following the argument that was employed in Prop. 3.2, we find that for $B_i = B_D(B_{i-1})$, $1 \leq i \leq n_d$, we have $n \in B_i$ and $B_U(B_i) \in \mathcal{H}(\mathcal{Q})$ (here, $B_0 = \mathcal{P}$). By induction and definition of \mathcal{Q} , $f(n_d, n_u) =$

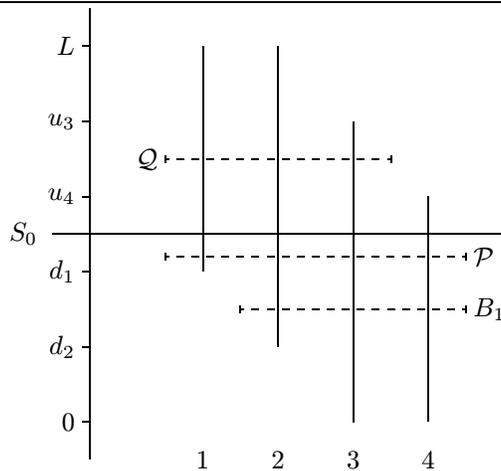


Figure 5: Portfolios \mathcal{P} and \mathcal{Q} for $n_d = n_u = 2$. Each instrument is represented by a vertical bar that marks the region of the price of the underlying asset in which it is possibly alive. Also shown is $B_1 = B_D(\mathcal{P})$

$$f(n_d, n_u - 1) + n_d + 1.$$

By symmetry, $f(n_d, n_u) = f(n_d - 1, n_u) + n_u + 1$. Together with the initial conditions $f(n_d, 0) = n_d$ and $f(0, n_u) = n_u$, we conclude that $f(n_d, n_u) = n_d + n_u + n_d n_u$. \square

Prop. 3.4 shows that the number of pricing equations that have to be solved for a portfolio of single barrier options is linear both in the number of up-and-out resp. down-and-out barrier options. Again, Prop. 3.4 describes the worst case upper bound; in general, the barriers need not be distinct. However, the formula given in Prop. 3.4 remains precise if n_d and n_u are interpreted as the overall numbers of distinct up-and-out resp. down-and-out barriers in any portfolio \mathcal{P} that contains only single-barrier options. If \mathcal{P} also contains vanilla options, one additional pricing equation needs to be solved, and the combinatorial complexity becomes $|\mathcal{H}(\mathcal{P})| = n_d + n_u + n_d n_u + 1$.

4 Numerical issues: matching the underlying lattice to all barriers

In our implementation, pricing equations are evaluated on a lattice, using an explicit finite-difference method. Lattice methods have been used before to evaluate barrier options with up to two barriers. In the UVM nonlinear framework, however, all barriers must be considered simultaneously for the reasons mentioned in the Introduction.

It is well-known to finance practitioners and to numerical analysts that the lattice should coincide with the barriers (Cheuk and Vorst (1996)). Since our pricer is designed for general portfolio inputs, we need to design an algorithm that constructs such a lattice dynamically. Our algorithm matches all barriers except for those that are very close together, within a specified limit of tolerance.³

4.1 The finite difference method

We assume that the current spot price S_0 and all relevant barriers are given inputs and that there are n up-and-out barriers $S_0 < b_1 < b_2 < \dots < b_n < \infty$. For simplicity, we assume that there are no down-and-out barriers. Set $b_0 = S_0$.

All pricing equations are evaluated on the same lattice. The lattice is constructed from $[S_D, S_U] \times [0, T]$, where S_D and S_U are suitably chosen boundaries. They do not concern us in this context and we ignore them in the following paragraphs.⁴ Let us assume initially a maximum time-step dt_{\max} , which determines the accuracy of the numerical approximation. The actual spacing in time for the lattice will be given by $t_k = k dt$, $k \in \{0, 1, \dots, \lceil T/dt \rceil\}$, for some dt , $0 < dt \leq dt_{\max}$, yet to be determined.

The discretization

$$\dots < S_{-2} < S_{-1} < S_0 < S_1 < S_2 < \dots$$

of space need not be uniform (in log space); the priority is rather to match each barrier b_i with some spatial level S_j . However, the discretization *can* be chosen uniform (in log space) between adjacent barriers. The factors $U_j = S_{j+1}/S_j$ resp. $D_j = S_{j-1}/S_j$ represent the size of the up resp. down moves for each spatial level. $U_i D_{i+1} = D_j U_{j-1} = 1$ always holds by definition.

Given U_j , D_j and dt , we define quantities $\bar{\sigma}_U^j$ and $\bar{\sigma}_D^j$ via

$$\begin{aligned} U_j &= 1/D_{j+1} = e^{\bar{\sigma}_U^j \sqrt{dt}} \\ D_j &= 1/U_{j-1} = e^{-\bar{\sigma}_D^j \sqrt{dt}} \end{aligned} \tag{4.1}$$

To simplify notation, we write V_j^k instead of $V(S_j, t_k)$ for all relevant $j \in \mathbb{N}$ and $k \in \{0, 1, \dots, \lceil T/dt \rceil\}$.

Proposition 4.1. *Given a lattice \mathcal{L} and—implicitly— dt , $\bar{\sigma}_U^j$, $\bar{\sigma}_D^j$ as described*

³ In practice, the tolerance is so small that most barriers are matched exactly.

⁴In practice, $S_U/S_0 = S_0/S_D = O(\sqrt{T/dt})$ leads to good results and limits the time complexity in the number of time steps to $O((T/dt)^{3/2})$. This strategy is implemented in our pricer.

above, set

$$\begin{aligned}
P_U(\sigma) &= \frac{\sigma^2}{\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_U^j)^2} \left(1 - \frac{\bar{\sigma}_D^j \sqrt{dt}}{2} \right) + \frac{\mu \bar{\sigma}_D^j \sqrt{dt}}{\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_U^j)^2} \\
P_D(\sigma) &= \frac{\sigma^2}{\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_D^j)^2} \left(1 + \frac{\bar{\sigma}_U^j \sqrt{dt}}{2} \right) - \frac{\mu \bar{\sigma}_U^j \sqrt{dt}}{\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_D^j)^2} \quad (4.2)
\end{aligned}$$

$$P_M(\sigma) = 1 - P_U(\sigma) - P_D(\sigma) .$$

Let (S_j, t_k) be a node of the lattice for some $j \in \mathbb{N}$ and $k \in \{0, 1, \dots, \lceil T/dt \rceil\}$. Then, the forward Euler approximation V_j^k of the solution V of (2.5) is

$$V_j^k = e^{-r t_k dt} \max_{\sigma} \{ P_U(\sigma) V_{j+1}^{k+1} + P_M(\sigma) V_j^{k+1} + P_D(\sigma) V_{j-1}^{k+1} \} \quad (4.3)$$

where the maximum is taken over $\{\sigma_{\min}, \sigma_{\max}\}$.

Proof. The proof mimics the one for the classical Euler scheme, adapting it to the non-uniform lattice. Define the process $X = \log(S)$. By assumption, V solves (2.5) and therefore

$$\frac{\partial V}{\partial t} + \frac{1}{2} \Sigma^2 \left\{ e^{-2X} \left(\frac{\partial^2 V}{\partial X^2} - \frac{\partial V}{\partial X} \right) \right\} \left(\frac{\partial^2 V}{\partial X^2} - \frac{\partial V}{\partial X} \right) + \mu \frac{\partial V}{\partial X} = rV \quad (4.4)$$

where

$$\Sigma^2 \{C\} = \begin{cases} \sigma_{\max}^2 & \text{if } C \geq 0 \\ \sigma_{\min}^2 & \text{if } C < 0 \end{cases} \quad (4.5)$$

Each node (S_j, t_k) of the lattice \mathcal{L} corresponds to a node $(X_j = \log S_j, t_k)$ in a dual lattice \mathcal{L}' . The increments at X_j are $\log U_j = \bar{\sigma}_U^j \sqrt{dt}$ and $\log D_j = \bar{\sigma}_D^j \sqrt{dt}$, respectively. Evaluating V on \mathcal{L} with respect to (2.5) is equivalent to evaluating V on \mathcal{L}' with respect to (4.4), after appropriate adjustment of the boundary conditions. Thus, the weights P_U , P_D and P_M derived from a forward Euler approximation for (4.4) enter unaltered into a corresponding approximation for (2.5).

We use the following finite difference approximations for the partial derivatives in (4.4):

$$\begin{aligned}
\frac{\partial V}{\partial t} &= \frac{V_j^{t+1} - V_j^t}{dt} \\
\frac{\partial V}{\partial X} &= \frac{1}{\sqrt{dt}} \frac{(\bar{\sigma}_D^j)^2 V_{j+1}^{t+1} - (\bar{\sigma}_U^j)^2 V_{j-1}^{t+1} - \left((\bar{\sigma}_D^j)^2 - (\bar{\sigma}_U^j)^2 \right) V_j^{t+1}}{\bar{\sigma}_U^j (\bar{\sigma}_D^j)^2 + \bar{\sigma}_D^j (\bar{\sigma}_U^j)^2} \quad (4.6) \\
\frac{\partial^2 V}{\partial X^2} &= \frac{2}{dt} \frac{\bar{\sigma}_D^j V_{j+1}^{t+1} + \bar{\sigma}_U^j V_{j-1}^{t+1} - (\bar{\sigma}_D^j + \bar{\sigma}_U^j) V_j^{t+1}}{\bar{\sigma}_U^j (\bar{\sigma}_D^j)^2 + \bar{\sigma}_D^j (\bar{\sigma}_U^j)^2}
\end{aligned}$$

The truncation error for $\frac{\partial V}{\partial X}$ is $O(\bar{\sigma}_U^j \bar{\sigma}_D^j)$; its approximation is therefore second order accurate in space. The approximation for $\frac{\partial^2 V}{\partial X^2}$ is second order accurate in space if $\bar{\sigma}_U^j = \bar{\sigma}_D^j$, in which case it reduces to the standard approximation; otherwise, it is first order accurate, and the truncation error is $O(|\bar{\sigma}_U^j - \bar{\sigma}_D^j|)$.

Straightforward algebraic transformations show that with P_U , P_D and P_M as defined in the proposition,

$$W_j^k(\sigma) = e^{-r_{t_k} dt} (P_U(\sigma)V_{j+1}^{k+1} + P_M(\sigma)V_j^{k+1} + P_D(\sigma)V_{j-1}^{k+1}) \quad (4.7)$$

is the finite difference approximation for fixed σ . Observing that

$$\Sigma^2 \left\{ e^{-2X} \left(\frac{\partial^2 V}{\partial X^2} - \frac{\partial V}{\partial X} \right) \right\} \left(\frac{\partial^2 V}{\partial X^2} - \frac{\partial V}{\partial X} \right) = \max_{\sigma} \left\{ \sigma \left(\frac{\partial^2 V}{\partial X^2} - \frac{\partial V}{\partial X} \right) \right\} \quad (4.8)$$

where the maximum is taken over $\sigma \in \{\sigma_{\min}, \sigma_{\max}\}$, we find that V_j^k is the solution of a one-period optimization problem:

$$V_j^k = \max\{W_j^k(\sigma_{\min}), W_j^k(\sigma_{\max})\} \quad (4.9)$$

This is just (4.3). \square

4.2 Stability

Stability of the finite difference scheme is guaranteed if the P_U , P_M and P_D are all positive. This is the case, for instance, if P_U and P_D satisfy

$$\begin{aligned} P_U &> 0 \\ P_D &> 0 \\ P_U + P_D &< \frac{1}{2} \end{aligned} \quad (4.10)$$

everywhere. (In particular, P_U , P_D and P_M can then be interpreted as probabilities.) $P_U + P_D < \frac{1}{2}$ guarantees that the middle weight is always at least $\frac{1}{2}$; this has been found empirically to lead to a significant improvement in accuracy in the case of barrier options. In the previous section, formulas for P_U , P_M and P_D based on a fixed discretization were derived. We will now provide an algorithm that finds a discretization for given dt_{\max} , σ_{\min} , σ_{\max} , μ_{\min} and μ_{\max} that

- satisfies (4.10) and
- matches the barriers b_0, b_1, \dots, b_n in most cases.

This algorithm is presented in Fig. 6 and computes dt and the local parameters $\bar{\sigma}_U^j$ and $\bar{\sigma}_D^j$ for each S_j -level of the lattice.

Proposition 4.2. *Given dt_{\max} , σ_{\min} , σ_{\max} , μ_{\min} , μ_{\max} and $n + 1$ barriers $S_0 = b_0 < b_1 < \dots < b_n$, the algorithm depicted in Fig. 6 outputs $0 < dt \leq dt_{\max}$ and $\bar{\sigma}_U^j, \bar{\sigma}_D^j$ for all relevant $j \in \mathbb{N}$ such that (4.10) is fulfilled for all P_U and P_D that are computed according to (4.2).*

-
1. Set $\bar{\mu} := \max\{|\mu_{\min}|, |\mu_{\max}|\}$
 2. Set $dt := dt_{\max}$. This is the initial guess and might be adjusted later
 3. For each $i = 0, \dots, n$ do the following
 - (a) Set $\bar{\sigma} := 2\sigma_{\max}$ (see remark in text)
 - (b) If $i = n$ then skip the next step (there are no more barriers above b_n)
 - (c) Increase $\bar{\sigma}$ such that $b_i e^{k\bar{\sigma}\sqrt{dt}} = b_{i+1}$ for some $k \in \mathbb{N}$. If no such k exists (i.e., $\ln(b_{i+1}/b_i) < \bar{\sigma}\sqrt{dt}$), abort and report an error (see remark in text)
 - (d) Set

$$dt' := \left[\frac{2\sigma_{\min}^2}{\bar{\sigma}(\sigma_{\max}^2 + 2\bar{\mu})} \right]^2$$

Check if $dt < dt'$. If yes, skip the next step (dt has passed the test)

- (e) dt is too big: choose a new $dt > 0$ such that $dt < dt'$ and start over with step 3
- (f) For all S_j such that $b_i < S_j < b_{i+1}$ (or simply $b_i < S_j$ if $i = n$), set $\bar{\sigma}_U^j := \bar{\sigma}_D^j := \bar{\sigma}$. In addition, set $\bar{\sigma}_U^i := \bar{\sigma}$, and if $i < n$, set $\bar{\sigma}_D^{i+1} := \bar{\sigma}$

Figure 6: The algorithm to compute dt and $\bar{\sigma}_U^j$ and $\bar{\sigma}_D^j$ for all relevant j . Input is dt_{\max} , σ_{\min} , σ_{\max} , μ_{\min} and μ_{\max}

Proof. It is easy to check that $dt \leq dt_{\max}$, for dt is set to dt_{\max} initially and, if necessary, only reduced in step 3e. $dt > 0$ holds because all quantities for dt' in step 3d are positive, thus $dt' > 0$.

To see that $P_U, P_D > 0$ consider (4.2) in the form

$$\begin{aligned} 2 \left(\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_U^j)^2 \right) P_U &= 2\sigma^2 - \sigma^2 \bar{\sigma}_D^j \sqrt{dt} + 2\mu \bar{\sigma}_D^j \sqrt{dt} \\ 2 \left(\bar{\sigma}_U^j \bar{\sigma}_D^j + (\bar{\sigma}_D^j)^2 \right) P_D &= 2\sigma^2 + \sigma^2 \bar{\sigma}_U^j \sqrt{dt} - 2\mu \bar{\sigma}_U^j \sqrt{dt} \end{aligned} \quad (4.11)$$

where $\sigma \in \{\sigma_{\min}, \sigma_{\max}\}$ and $\mu_{\min} \leq \mu \leq \mu_{\max}$. $\bar{\sigma}_U^j, \bar{\sigma}_D^j > 0$ implies that $P_U, P_D > 0$ is equivalent to

$$\begin{aligned} 2\sigma^2 - \sigma^2 \bar{\sigma}_D^j \sqrt{dt} + 2\mu \bar{\sigma}_D^j \sqrt{dt} &> 0 \\ 2\sigma^2 + \sigma^2 \bar{\sigma}_U^j \sqrt{dt} - 2\mu \bar{\sigma}_U^j \sqrt{dt} &> 0 \end{aligned} \quad (4.12)$$

for $\sigma \in \{\sigma_{\min}, \sigma_{\max}\}$ and $\mu_{\min} \leq \mu \leq \mu_{\max}$. With $\bar{\mu} = \max\{|\mu_{\min}|, |\mu_{\max}|\}$ and

$$\bar{\sigma}(\sigma_{\max}^2 + 2\bar{\mu}) < \frac{2\sigma_{\min}^2}{\sqrt{dt}} \quad (4.13)$$

for $\bar{\sigma} \in \{\bar{\sigma}_D^j, \bar{\sigma}_U^j\}$ (implied by step 3d), this can be shown as follows:

$$\begin{aligned}
& 2\sigma^2 - \sigma^2 \bar{\sigma}_D^j \sqrt{dt} + 2\mu \bar{\sigma}_D^j \sqrt{dt} \\
& \geq 2\sigma_{\min}^2 - \bar{\sigma}_D^j (\sigma_{\max}^2 + 2\bar{\mu}) \sqrt{dt} \\
& > 2\sigma_{\min}^2 - \frac{2\sigma_{\min}^2}{\sqrt{dt}} \sqrt{dt} \\
& = 0
\end{aligned} \tag{4.14}$$

for P_U and, by the same token,

$$\begin{aligned}
& 2\sigma^2 + \sigma^2 \bar{\sigma}_U^j \sqrt{dt} - 2\mu \bar{\sigma}_U^j \sqrt{dt} \\
& \geq 2\sigma_{\min}^2 - \bar{\sigma}_U^j (\sigma_{\max}^2 + 2\bar{\mu}) \sqrt{dt} \\
& > 0
\end{aligned} \tag{4.15}$$

for P_D . Finally, by using $\bar{\sigma}_U^j, \bar{\sigma}_D^j \geq 2\sigma_{\max}$ (step 3a) and (4.13), we get

$$\begin{aligned}
P_U + P_D &= \frac{\sigma^2}{\bar{\sigma}_U^j (\bar{\sigma}_D^j + \bar{\sigma}_U^j)} - \frac{\frac{1}{2}\sigma^2 \bar{\sigma}_D^j \sqrt{dt}}{\bar{\sigma}_U^j (\bar{\sigma}_D^j + \bar{\sigma}_U^j)} + \frac{\mu \bar{\sigma}_D^j \sqrt{dt}}{\bar{\sigma}_U^j (\bar{\sigma}_D^j + \bar{\sigma}_U^j)} + \\
& \quad \frac{\sigma^2}{\bar{\sigma}_D^j (\bar{\sigma}_U^j + \bar{\sigma}_D^j)} + \frac{\frac{1}{2}\sigma^2 \bar{\sigma}_U^j \sqrt{dt}}{\bar{\sigma}_D^j (\bar{\sigma}_U^j + \bar{\sigma}_D^j)} - \frac{\mu \bar{\sigma}_U^j \sqrt{dt}}{\bar{\sigma}_D^j (\bar{\sigma}_U^j + \bar{\sigma}_D^j)} \\
&= \frac{\sigma^2}{\bar{\sigma}_D^j \bar{\sigma}_U^j} + \frac{\frac{1}{2}\sigma^2 \left((\bar{\sigma}_U^j)^2 - (\bar{\sigma}_D^j)^2 \right) \sqrt{dt}}{\bar{\sigma}_D^j \bar{\sigma}_U^j (\bar{\sigma}_D^j + \bar{\sigma}_U^j)} + \frac{\mu \left((\bar{\sigma}_D^j)^2 - (\bar{\sigma}_U^j)^2 \right) \sqrt{dt}}{\bar{\sigma}_D^j \bar{\sigma}_U^j (\bar{\sigma}_D^j + \bar{\sigma}_U^j)} \\
&= \frac{1}{\bar{\sigma}_D^j \bar{\sigma}_U^j} \left[\sigma^2 + \left(\frac{1}{2}\sigma^2 (\bar{\sigma}_U^j - \bar{\sigma}_D^j) + \mu (\bar{\sigma}_D^j - \bar{\sigma}_U^j) \right) \sqrt{dt} \right] \\
&\leq \frac{1}{4\sigma_{\max}^2} \left[\sigma_{\max}^2 + \left(\frac{1}{2}\sigma_{\max}^2 + \bar{\mu} \right) |\bar{\sigma}_U^j - \bar{\sigma}_D^j| \sqrt{dt} \right] \\
&\leq \frac{1}{4\sigma_{\max}^2} \left[\sigma_{\max}^2 + \frac{1}{2} (\sigma_{\max}^2 + 2\bar{\mu}) \max\{\bar{\sigma}_U^j, \bar{\sigma}_D^j\} \sqrt{dt} \right] \\
&< \frac{1}{4\sigma_{\max}^2} \left[\sigma_{\max}^2 + \frac{1}{2} \frac{2\sigma_{\min}^2}{\sqrt{dt}} \sqrt{dt} \right] \\
&\leq \frac{1}{4\sigma_{\max}^2} (\sigma_{\max}^2 + \sigma_{\min}^2) \\
&\leq \frac{1}{2}
\end{aligned} \tag{4.16}$$

which completes the proof. \square

Two remarks are in order. Firstly, step 3a in the algorithm can safely be replaced by

3a' Set $\bar{\sigma} := \sqrt{2}\sigma_{\max}$

In this case $P_U + P_D < \frac{1}{2}$ only if $\bar{\sigma}_U^j = \bar{\sigma}_D^j$. For $\bar{\sigma}_U^j \neq \bar{\sigma}_D^j$, the upper bound becomes $P_U + P_D < 1$ instead. This still guarantees $P_M > 0$ and therefore does not break the probability framework of the derivation. Moreover, $\bar{\sigma}_U^j \neq \bar{\sigma}_D^j$ for at most n S -levels of the lattice (n is the number of barriers). The ratio of the number of “good” j ’s ($\bar{\sigma}_U^j = \bar{\sigma}_D^j$) over the number of “bad” j ’s ($\bar{\sigma}_U^j \neq \bar{\sigma}_D^j$) is therefore negligible as the granularity of the lattice gets finer. Indeed, 3a’ is used in our pricer.

Secondly, the error condition in step 3c has not been mentioned in the above discussion. If two barriers are too close to each other (which is discovered in step 3c), one of them can simply be ignored, and the algorithm can be rerun with the number of barriers reduced by one. This is the heuristic that is used in our pricer.

5 Numerical results

5.1 Implementation

Our pricer uses an explicit finite difference solver as described in Sect. 4. It allows for time-varying volatility and drift parameters and trims the lattice to achieve better performance. It performs the closure operation described in Sect. 3 to find the hierarchy of pricing equations that need to be solved. An implementation on a Pentium/166 machine running Windows NT 4.0 was used to run the following experiments. (A Linux implementation can also be accessed through a browser interface at “<http://home.cs.nyu.edu:8080/cgi-bin/vop>”.)

In the following, the pricer is shortly called “VOP” (for Virtual Option Pricer). Experiments 1 and 2 will be primarily concerned with numerical accuracy.

5.2 Experiment 1: pricing a double barrier call option

In order to test the accuracy of the finite difference solver of VOP in the case of double barrier options, we compare its results to those given by Geman and Yor (1996) for their method respectively for the method described in Kunitomo and Ikeda (1992). Geman and Yor use a probabilistic approach to price double barrier options, while Kunitomo and Ikeda suggest to use a pricing formula expressed as the sum of an infinite series whose convergence they claim is rapid.

In Fig. 7, three test cases are shown, as well as the answers produced by VOP, Geman and Yor’s method (G-Y) and Kunitomo and Ikeda’s method (K-I). Throughout, $S_0 = 2$ and $T = 365$ days. D and U in the table denote the up-and-out resp. the down-and-out barrier. The lattice granularity for VOP is set to $N = 1, 5, 20$ and 50 time steps per day, respectively. For $N = 1$, the result appears almost instantaneously. For $N = 5$, roughly one second is needed to find the answer. VOP’s time complexity in N is $O(N^{3/2})$, due to the trimming applied to the outer parts of the lattice.

	N	$\sigma = 0.2$	$\sigma = 0.5$	$\sigma = 0.5$
	(periods	$r = 0.02$	$r = 0.05$	$r = 0.05$
	per	$K = 2$	$K = 2$	$K = 1.75$
	day)	$D = 1.5, U = 2.5$	$D = 1.5, U = 3$	$D = 1, U = 3$
VOP	1	0.040899	0.017666	0.075914
VOP	5	0.041050	0.017819	0.076102
VOP	20	0.041079	0.017848	0.076158
VOP	50	0.041085	0.017853	0.076168
G-Y	–	0.0411	0.0178	0.07615
K-I	–	0.041089	0.017856	0.076172

Figure 7: Prices obtained for a double barrier call option with each of the three methods VOP, G-Y and K-I. For VOP, the time step varies between $dt = 1/365$ and $dt = 1/(50 \times 365)$

The convergence of the prices computed by VOP is very satisfactory. For $N = 5$, VOP and the other two methods yield identical results over up to four digits after the decimal point.

5.3 Experiment 2: pricing a portfolio of single barrier put options

In a second experiment, the following portfolio of down-and-out put options was priced:

type	strike	barrier	expiration	position
put	100	98	30 days	long 200 contracts
put	100	95	30 days	long 10 contracts
put	100	90	30 days	long 2 contracts
put	100	85	30 days	long 1 contract

The other parameters were $S_0 = 100$, $r = 0.025$ and $\sigma = 0.2$. The position in each individual put option is roughly inverse proportional to the relative value that it contributes to the portfolio as a whole.

The results obtained in this experiment are summarized in Fig. 8. Also shown in Fig. 8 are the values of $\bar{\sigma}_D$ and $\bar{\sigma}_U$ as defined in Sect. 4. There are four regions in the lattice for which the values of $\bar{\sigma}_D$ and $\bar{\sigma}_U$ differ; the three interior barriers at 98, 95 and 90 mark the boundaries between these regions. Note that the values of $\bar{\sigma}_U$ and $\bar{\sigma}_D$ converge rapidly as N increases.

For $N = 50, 100, 200$ and 400 the answer was computed in 2, 7, 18 and 51 seconds, respectively. A closed form formula for down-and-out barrier puts yields 10.287 as the Black-Scholes value. The numerical result is sufficiently close for $N \geq 100$.

N (periods per day)	price	$\bar{\sigma}_D, \bar{\sigma}_U$ between 98 and above	$\bar{\sigma}_D, \bar{\sigma}_U$ between 98 and 95	$\bar{\sigma}_D, \bar{\sigma}_U$ between 95 and 90	$\bar{\sigma}_D, \bar{\sigma}_U$ between 90 and below
1	7.724288	0.385972	0.296992	0.344318	0.364003
2	7.792857	0.334262	0.342937	0.298188	0.315236
5	9.966664	0.287687	0.332048	0.288719	0.305226
10	10.114629	0.305138	0.313057	0.296953	0.287770
20	10.209345	0.287687	0.295153	0.288719	0.287272
50	10.253239	0.303248	0.300008	0.292163	0.285988
100	10.270897	0.296902	0.282850	0.286931	0.287371
200	10.279317	0.287288	0.289663	0.286434	0.285988
400	10.283227	0.285905	0.282850	0.283001	0.283639

Figure 8: Prices obtained for a portfolio of four down-and-out put options. The time step varies between $dt = 1/365$ and $dt = 1/(400 \times 365)$. Also indicated are the values $\bar{\sigma}_D$ and $\bar{\sigma}_U$, summarized for the four significant regions of the lattice, as determined by the interior barriers at 98, 95 and 90

5.4 Experiment 3: hedging two barrier options

We no longer assume that the volatility is constant; rather, $\sigma_{\min} = 0.1$ and $\sigma_{\max} = 0.2$ are set as upper and lower bounds. Furthermore, we choose $S_0 = 100$ and $r = 0.02$.

Consider an agent who wants to evaluate a liability structure A stemming from short positions in a double barrier call and a single barrier put:

type	strike	U&O-barrier	D&O-barrier	expiration	position
call	110	120	90	30 days	short
put	100	--	95	30 days	short

In order to hedge this position, the agent enters offsetting positions B in liquidly traded vanilla options as follows:

type	strike	expiration	quoted price	position
call	110	30 days	0.17 imp. vol	short 3.3 contracts
call	100	30 days	0.13 imp. vol	long 1.1 contracts
call	90	30 days	0.15 imp. vol	short 4 contracts

It can be calculated that the agent receives 39.075 from these last transactions (the call with strike 90 being the major contributor). The worst-case price range of the combined position $A \cup B$ is shown in Fig .9, obtained from running the pricer with various values of dt .

The result can be interpreted as follows: if the agent charges at least 1.14732 (see the last column in Fig. 9) for the liability structure A of barrier options

N (periods per day)	Lower bound on combined position	Upper bound on combined position	Lower bound minus premium for vanillas
20	-40.219785	-38.373091	-1.144785
50	-40.221489	-38.373258	-1.146489
100	-40.221955	-38.373244	-1.146955
200	-40.222174	-38.373222	-1.147174
400	-40.222320	-38.373255	-1.147320

Figure 9: Price range obtained for a portfolio of a double barrier option, a single barrier option and three traded vanillas. The volatility band is $0.1 \leq \sigma \leq 0.2$. Prices are negative because the agent is a seller. The agent receives a net premium of 39.075 for her transactions in the vanilla options

and at the same time enters the offsetting position B , she will break even or make a profit provided the volatility stays within the band $0.1 \leq \sigma \leq 0.2$ over the next 30 days.⁵

For this scenario, four portfolios had to be evaluated ($|\mathcal{H}| = 4$): (1) the two barrier options plus the vanillas; (2) the double barrier option plus the vanillas; (3) the single barrier option plus the vanillas; and (4) the vanillas. The computing times for $N = 20, 50, 100, 200$ and 400 were 1, 6, 19, 51 and 148 seconds, respectively. ($N = 400$ is equivalent to $400 \times 30 = 12000$ time steps.) It can be seen that $N = 20$ is sufficient for accuracy to three leading digits. For four leading digits, N must be at least 50.

6 Conclusion

Nonlinear models are useful for managing volatility risk, but they require advanced numerical *and* algorithmic techniques. We presented the details of an algorithm that allows to implement the Uncertain Volatility Model to situations where combinations of vanilla, single and double barrier options are to be evaluated simultaneously. Our algorithm can be applied to different types of exotic options, such as “range accruals,” not directly covered in VOP.

We used a forward Euler finite difference method on a non-uniform lattice tailored to the set of barriers presented in the input. We showed that this method can compete with closed-form solutions in terms of accuracy and speed. In addition, the method is well suited to tackle problems which cannot be solved in closed form. We presented an example in which an optimal hedge

⁵It can be shown that position B is optimal in the sense that every other position makes it necessary to charge at least 1.14732 in order to provide cover for the worst case volatility scenario—see Avellaneda and Parás (1996) for details. Note, however, that the situation is not symmetric for the buy and sell side: $-38.373255 - (-39.075) = 0.701745$ is not necessarily the price of A under the hedge B which covers the worst case from a buyer’s point of view.

for a portfolio of two barrier options was evaluated, under uncertain volatility assumptions.

References

- Avellaneda, M., A. Levy, and A. Parás (1995): “Pricing and Hedging Derivative Securities in Markets with Uncertain Volatilities,” *Applied Mathematical Finance*, 2, 73–88.
- Avellaneda, M., and A. Parás (1996): “Managing the Volatility Risk of Portfolios of Derivative Securities: the Lagrangian Uncertain Volatility Model,” *Applied Mathematical Finance*, 3, 21–52.
- Carr, P., and A. Chou (1997): “Breaking barriers,” *RISK*, 10, No. 9, 139–145.
- Carr, P., K. Ellis, and V. Gupta (1997): “Static Hedging of Exotic Options,” *Journal of Finance*, forthcoming.
- Cheuk, T. H. F. and T. C. F. Vorst (1996): “Complex Barrier Options,” *The Journal of Derivatives*, 4, No. 1, 8–22.
- Conze, A., and Viswanathan (1991): “Path Dependent Options: The Case of Lookback Options,” *Journal of Finance*, 46, No. 5, 1893–1907.
- Geman, H., and M. Yor (1996): “Pricing and Hedging Double-barrier Options: a Probabilistic Approach,” *Mathematical Finance*, 6, No. 4, 365–378.
- Kunitomo, N., and M. Ikeda (1992): “Pricing Options with Curved Boundaries”, *Mathematical Finance*, 2, No. 4, 275–272.
- Roberts, C. O., and C. F. Shortland (1997): “Pricing Barrier Options with Time-dependent Coefficients”, *Mathematical Finance*, 7, No. 1, 93–93.
- Rubinstein, M., and E. Reiner (1991): “Breaking Down the Barriers,” *RISK*, 4, No. 8.