Tree methods for Pricing Exotic Options

Antonino Zanette
University of Udine

antonino.zanette@uniud.it

Path-dependent options

Black-Scholes model

$$\frac{dS_t}{S_t} = r dt + \sigma dB_t, \quad S_0 = s_0,$$

- Barrier option.
- Asian option.
- Lookback option

Barrier option

The price of an down-out option

$$P(0, s) = E\left[e^{-rT} f(S_T) \mathbf{1}_{(\mathrm{m_T > L})} | \mathrm{S_0 = s}\right].$$

where m_T is

$$m_T = \min_{0 \le t \le T} S_t$$

Binomial method

$$E\left[e^{-rT} f(S^N)\mathbf{1}_{(\mathrm{S_k > L, k=0\ldots N})}\right].$$

Backward induction

$$\begin{cases} v(N,x) = f(x) & \text{if} \quad x > L \\ v(N,x) = 0 & \text{if} \quad x \le L \\ v(n,x) = e^{-r\Delta T}\Big[qv(n+1,xu) + (1-q)v(n+1,xd)\Big] & \text{if} \quad x > L \\ v(n,x) = 0 & \text{if} \quad x \le L \end{cases}$$

Drawbacks

The classical CRR may be problematic when applied to barrier options because the convergence is very slow compared with that for standard vanilla options. (Boyle and Lau Journal of Derivatives 94)

The reason is clear: let n_L denote the index such that

$$S_0 d^{n_L} \geq L > S_0 d^{n_L+1}$$

Then the algorithm, N being fixed, yields the same result for any value of the barrier between $S_0 d^{n_L}$ and $S_0 d^{n_L+1}$.

Tree literature for continuous barrier options

All the paper in the literature share the same idea : the barrier coincides (or is very close) with the tree's nodes in order to improve the convergence behaviour.

- Boyle-Lau Choose the number of time step in order to be close to the barrier with a layer of nodes. Journal of Derivatives 94
- Ritchken He align a layer of nodes of the trinomial tree with each barrier. Journal of Derivatives 1995.
- Cheuck Vorst Trinomial method that solve the “near-barrier” problem. Journal of International Money and Finance 1997.
- Gaudenzi-Lepellere Interpolations near barrier and intial stock prices. International Journal of Applied an Theoretical Finance 2008.
- Dai-Liu The Bino-trinomial tree. Journal of Derivatives 2010.
- Gaudenzi-Zanette Tree mesh points approach. Decisions Economics an Finance 2008.

Ritchken algorithm

Ritchken noted that the trinomial method, for the extra freedom in chosing the parameters λ, can be prefered to the binomial one. The main idea here is to choose the stretch parameter λ such that the barrier is hit exactly.

$$s_0 d^N = L$$

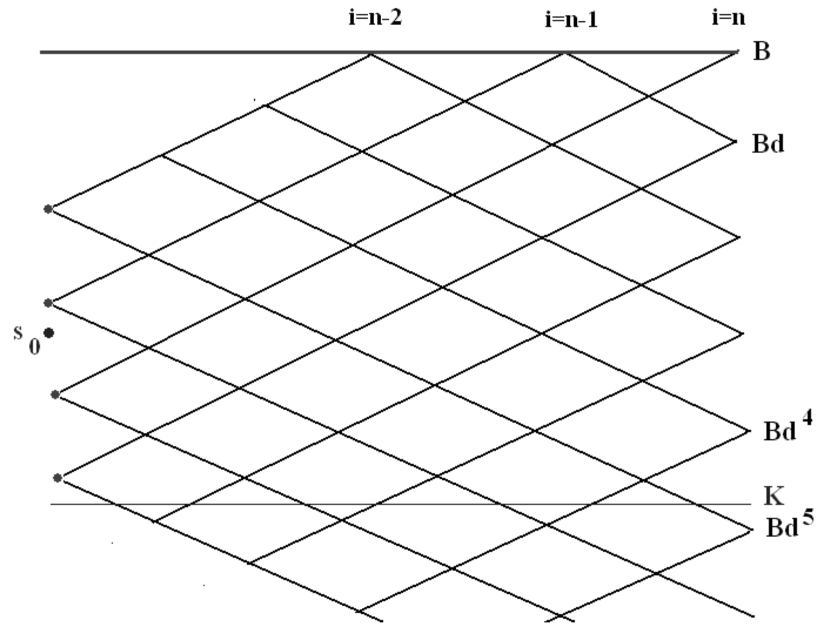
and then choose

$$\lambda = \frac{1}{N} \frac{\ln\left(\frac{S_0}{L}\right)}{\sigma\sqrt{\Delta T}}.$$

Tree mesh points method

- As remarked in several previous papers (see Boyle-Lau 94,Cheuk-Vorst 97, Gaudenzi-Lepellere 06) the price of a barrier option is a 'good approximation' of the continuous value when the barrier lies (or it is close) on a line of nodes of the tree.
- We construct a tree where all the tree mesh points are generated by the barrier itself.
- permits us to treat in a natural way and efficiently the 'near-barrier' problem, that occurs when the initial asset price is very close to the barrier.

Tree mesh points



It is worth to say that the mesh does not seem to be natural in order to describe the evolution of the asset price.

Nevertheless, this is not important. In fact we only need to set up the state-space of the Markov chain that we want to approximate the continuous time process.

Finite Difference approach for PDE

- In this way at time $t = 0$ we obtain four nodes with underlying assets: Bd^{j_S+1}, Bd^{j_S}, Bd^{j_S-1}, Bd^{j_S-2} and corresponding prices: $v_0(Bd^{j_S+l}), \quad l = 1, 0, -1, -2.$

- We interpolate (by a Lagrange 4 points interpolation) the points

$$(Bd^{j_S+l}, v_0(Bd^{j_S+l})), \quad l = 1, 0, -1, -2$$

  at the value s_0.

- In the case of down-and-out call option the nodes of the tree now are of type Bu^j, $j \geq 0$.

- When there are no nodes between s_0 and B (near-barrier problem) we modify the choice of the interpolation points taking $(Bd^{j_S+l}, v_0(Bd^{j_S+l})), \quad l = 2, 1, 0, -1.$

RMSRE for 5000 Options

	n	SP	$Cheuk-Vorst$	$Gaud.-Lepellere$
Up put	100	$2.67\ 10^{-3}$ (1.73)	$4.49\ 10^{-3}$ (2.58)	$3.13\ 10^{-3}$ (1.09)
	200	$1.23\ 10^{-3}$ (5.11)	$1.62\ 10^{-3}$ (10.05)	$1.49\ 10^{-3}$ (3.46)
	400	$6.18\ 10^{-4}$ (18.15)	$6.68\ 10^{-4}$ (39.93)	$7.79\ 10^{-4}$ (12.96)
	600	$3.82\ 10^{-4}$ (42.20)	$4.29\ 10^{-4}$ (90.57)	$5.02\ 10^{-4}$ (29.27)
Down put	100	$1.48\ 10^{-3}$ (2.12)	$1.66\ 10^{-3}$ (2.47)	$1.86\ 10^{-3}$ (1.54)
	200	$7.51\ 10^{-4}$ (7.14)	$8.77\ 10^{-4}$ (9.50)	$9.48\ 10^{-4}$ (5.00)
	400	$3.84\ 10^{-4}$ (26.31)	$4.19\ 10^{-4}$ (35.65)	$4.50\ 10^{-4}$ (18.24)
	600	$2.43\ 10^{-4}$ (59.16)	$2.76\ 10^{-4}$ (78.93)	$3.00\ 10^{-4}$ (39.00)

Asian options

The price of an European Asian option is given by

$$P(0,s,s) = E\left[e^{-rT} f(S_T, A_T) | S_0 = s, A_0 = s\right].$$

where A_T is the integral mean

$$A_T = \frac{1}{T}\int_0^T S_t$$

Payoff examples

- Fixed Asian Call: the payoff is $(A_T - K)_+$.
- Fixed Asian Put: the payoff is $(K - A_T)_+$.
- Floating Asian Call: the payoff is $(S_T - A_T)_+$.
- Floating Asian Put: the payoff is $(A_T - S_T)_+$.

American Asian options

The price of an American Asian option of initial time 0 and maturity T is:

$$P(0, S_0, A_0) = \sup_{\tau \in \mathcal{T}_{0,T}} E\left[e^{-r\tau}\psi(S_\tau, A_\tau)|S_0 = s_0, A_0 = s_0\right],$$

$$A_\tau = \frac{1}{\tau}\int_0^\tau S_t dt$$

Discrete approximation

Idea: approximate the integral mean with the arithmetic average.

$$E\left[e^{-rT} f(S^N, A^N)\right].$$

where

$$A^N = \frac{1}{N+1}\sum_{n=0}^{N} S^n$$

Pure Binomial method

The average process $(A_i)_{0\le i\le n}$ is recursively computed by

$$A_{i+1} = \frac{(i+1)A_i + S_{i+1}}{i+2}, A_0 = s_0.$$

The bidimensional transition matrix is given by

$$\begin{array}{lllll} \text{up} & (x,y) & \to & (xu, \frac{(n+1)y+xu}{n+2}) & \text{with probability } q \\ \text{down} & (x,y) & \to & (xd, \frac{(n+1)y+xd}{n+2}) & \text{with probability } 1-q \end{array}$$

Backward induction

$$\begin{cases} v(N,x,y) = f(x,y) \\ v(n,x,y) = e^{-r\Delta T}\Big[qv(n+1,xu,\dfrac{(n+1)y+xu}{n+2}) + (1-q)v(n+1,xd,\dfrac{(n+1)y+xd}{n+2})\Big], \end{cases}$$

Rem In the American case we have to take in account the early exercise $(y-k)_+$

Complexity

The obtained tree is not recombining so that the algorithm is of exponential complexity. The evaluation of $v(0, s_0, s_0)$ requires time computations and memory requirement of the order $O(2^n)$ and this fact shows that the algorithm is completely unfeasible from a practical point of view. Oss Se $n = 50$, $2^{50} = 1.12 \times 10^{15}$.

Implementation of the algorithm

- Computation of 2^N averages at maturity $v(N, x, y) = f(x, y)$. Binary representation.

$$vp[i] = (vm[i] - K)_+, \quad i = 0 \ldots\ldots (2^N - 1)$$

- For all $n = (N-1) \ldots\ldots\ldots 0$

$$vp[i] = e^{-r\Delta T}\Big(q \quad vp[2i+1] + (1-q) \quad vp[2i]\Big), \quad i = 0 \ldots\ldots\ldots (2^n - 1)$$

Hull-White algoritm

Idea: The main idea of this procedure is to restrict the range of the possible arithmetic averages to a set of some representative values. These values are selected in order to span all the possible values of the averages reachable at each node of the tree. The price is then computed by a backward induction procedure where the prices associated to the averages not included in the set of representative values, are obtained by some suitable interpolation methods.

$$A_{min}^{N} = s_0 \frac{1}{N+1} \sum_{k=0}^{N} d^k = s_0 \frac{1}{N+1} \frac{1 - d^{N+1}}{1-d}$$

$$A_{max}^{N} = s_0 \frac{1}{N+1} \sum_{k=0}^{N} u^k = s_0 \frac{1}{N+1} \frac{u^{N+1} - 1}{u-1}$$

In particular for every node (n, j)

$$A_{min}^{n,j} = \frac{1}{n+1} s_0 (1 + d + \ldots\ldots + d^{j-1} + d^j + d^j u + d^j u^2 + \ldots\ldots + d^j u^{n-j}) =$$

$$\frac{1}{n+1} s_0 \left[\frac{1 - d^{j+1}}{1-d} \right] + \frac{1}{n+1} s_0 d^j \left[\frac{u^{n-j+1} - 1}{u-1} - 1 \right]$$

$$A_{max}^{n,j} = \frac{1}{n+1} s_0 \left[\frac{u^{n-j+1} - 1}{u-1} \right] + \frac{1}{n+1} s_0 u^{n-j} \left[\frac{1 - d^{j+1}}{1-d} - 1 \right]$$

Hull-White algorithm

Discretization mesh of type

$$A^{k,n} = s_0 e^{mh}$$

where for a given h, the range of m values is selected to span the possible average at timestep n. Hull and White suggest that, to ensure accuracy for the algorithm, the value $h = 0.005$ is sufficient. Linear interpolation should be performed Complexity of order N^3.

FS Method Forward Shooting Grid Method of Barraquand-Pudet for both Fixed or Floating Strike cases.

$S_j^n = s_0 e^{j\sigma\sqrt{h}}, A_k^n = s_0 e^{k\sigma\sqrt{h}} \qquad j,k \ = -n, ..., n$ where $n = N, .., 0$.
If at time n the bidimensional process is at (S_j^n, A_k^n), at time n+1 the process can reach in the upward and downward transition cases

$$\begin{array}{lllll} \text{up} & (S_j^n, A_k^n) & \rightarrow & (S_{j+1}^{n+1}, A_{k+}^{n+1}) & \text{with probability } p_u \\ \text{down} & (S_j^n, A_k^n) & \rightarrow & (S_{j-1}^{n+1}, A_{k-}^{n+1}) & \text{with probability } p_d \end{array}$$

$$(1) \qquad \begin{cases} C_{j,k}^N = \psi(S_j^N, A_k^N) = (A_k^N - K)_+ \\ C_{j,k}^n = \max\left(\psi(S_j^n, A_k^n), e^{-r\Delta T}\left[p_u C_{j+1,k+}^{n+1} + p_d C_{j-1,k-}^{n+1}\right)\right] \end{cases}$$

Remark 1 Time complexity of FSG algorithm is $O(N^3)$ and the convergence is slow
Remark 2 However, these techniques have some drawbacks related both to the precision of the approximations and to the convergence to the continuous value, as observed by Forsyth et al in Review of Derivatives Research 2002. Forsyth et al proved that a procedure of order $O(n^{\frac{7}{2}})$ is necessary in order to assure the convergence of these algorithms.

Singular points methods

- American Asian arithmetic average option
- Binomial algorithm with 200 steps
- Relative error of order 10^{-4}
- Very few requirement of computational time (less than 2 sec) and space memory.

Singular points method

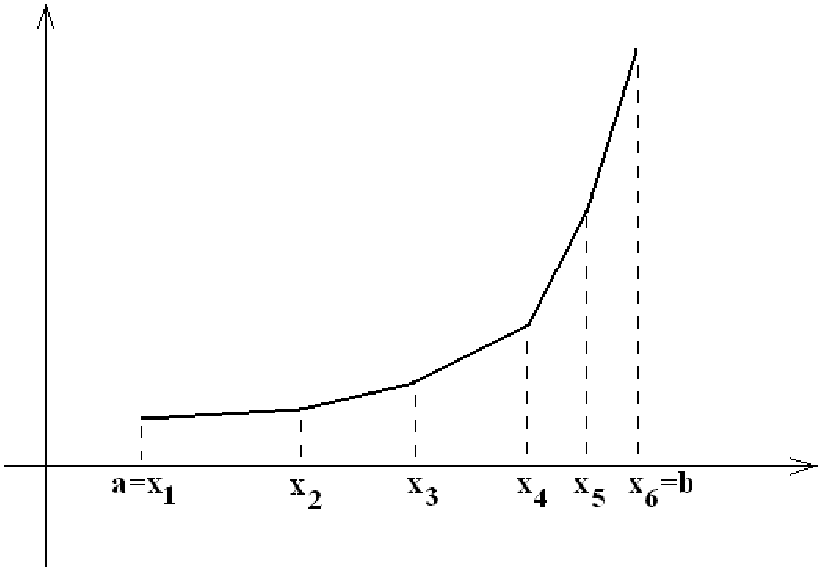
- The main idea of our method is to give a continuous representation of the option price function at every node of the tree as a piecewise linear convex function of the path-dependent variable (average or maximum/minimum)
- These functions are characterized only by a set of points that we name singular points.
- The property of convexity allows to obtain in a simple way upper and lower bounds of the price.

Singular points

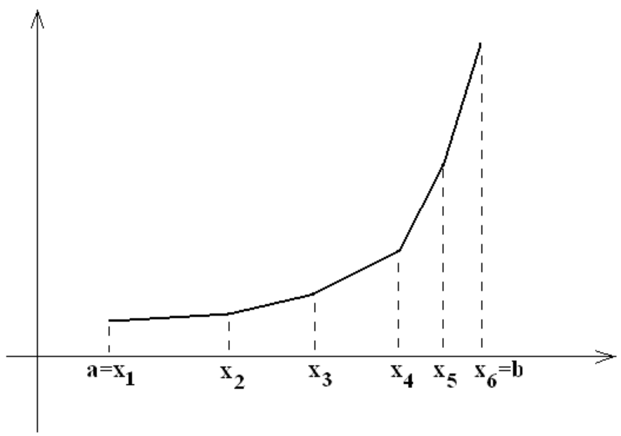
Given a set of points: $(x_1, y_1), ..., (x_n, y_n)$, such that $a = x_1 < x_2 < ... < x_n = b$ and

$$\frac{y_i - y_{i-1}}{x_i - x_{i-1}} < \frac{y_{i+1} - y_i}{x_{i+1} - x_i}, \qquad i = 2, ..., n-1, \tag{2}$$

let us consider the function $f(x)$, $x \in [a, b]$, obtained by interpolating linearly the given points.



We consider only piecewise linear functions with strictly increasing slopes, so that the function f is convex



The points $(x_1, y_1), \ldots, (x_n, y_n)$ (which characterize f), will be called the singular points of f.

UPPER BOUND

Lemma 1 *Let f be a piecewise linear and convex function defined on $[a, b]$, and let $C = \{(x_1, y_1), ..., (x_n, y_n)\}$ be the set of its singular points.*
Removing a point (x_i, y_i) from the set C, the resulting piecewise linear function $\widetilde{f}$, whose set

of singular points is $C \setminus \{(x_i, y_i)\}$, is again convex in $[a, b]$ and we have:

$$f(x) \leq \widetilde{f}(x), \quad \forall x \in [a, b].$$

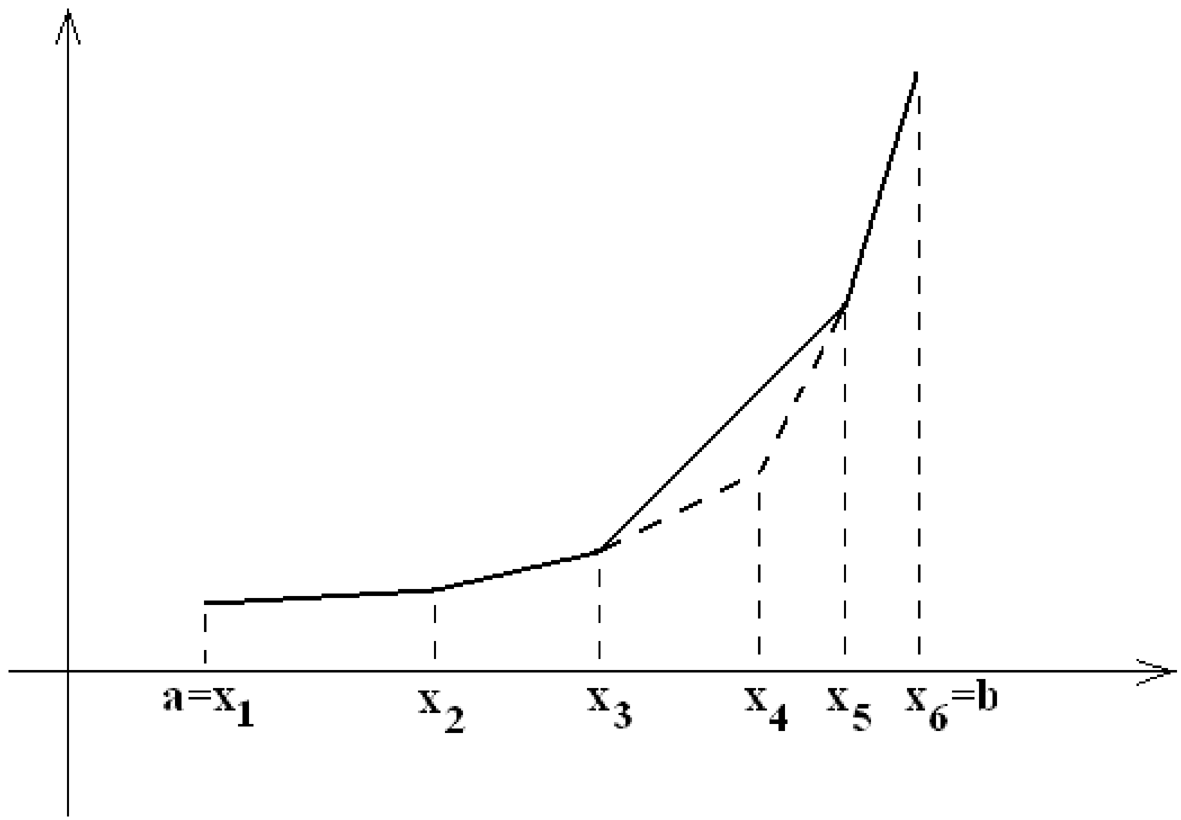


Figure 1: Upper estimate: x_4 has been removed.

LOWER BOUND

Lemma 2 *Let f be a piecewise linear and convex function defined on $[a, b]$, and let $C = \{(x_1, y_1), ..., (x_n, y_n)\}$ be the set of its singular points.*

Let $(\overline{x}, \overline{y})$ be the intersection between the straight line joining (x_{i-1}, y_{i-1}), (x_i, y_i) and the one joining (x_{i+1}, y_{i+1}), (x_{i+2}, y_{i+2}).

If we consider the new set of $n - 1$ singular points

$$\{(x_1, y_1), ..., (x_{i-1}, y_{i-1}), (\overline{x}, \overline{y}), (x_{i+2}, y_{i+2}), ..., (x_n, y_n)\},$$

the associated piecewise linear function $\widetilde{f}$ is again convex on $[a, b]$ and we have:

$$f(x) \geq \widetilde{f}(x), \quad \forall x \in [a, b].$$

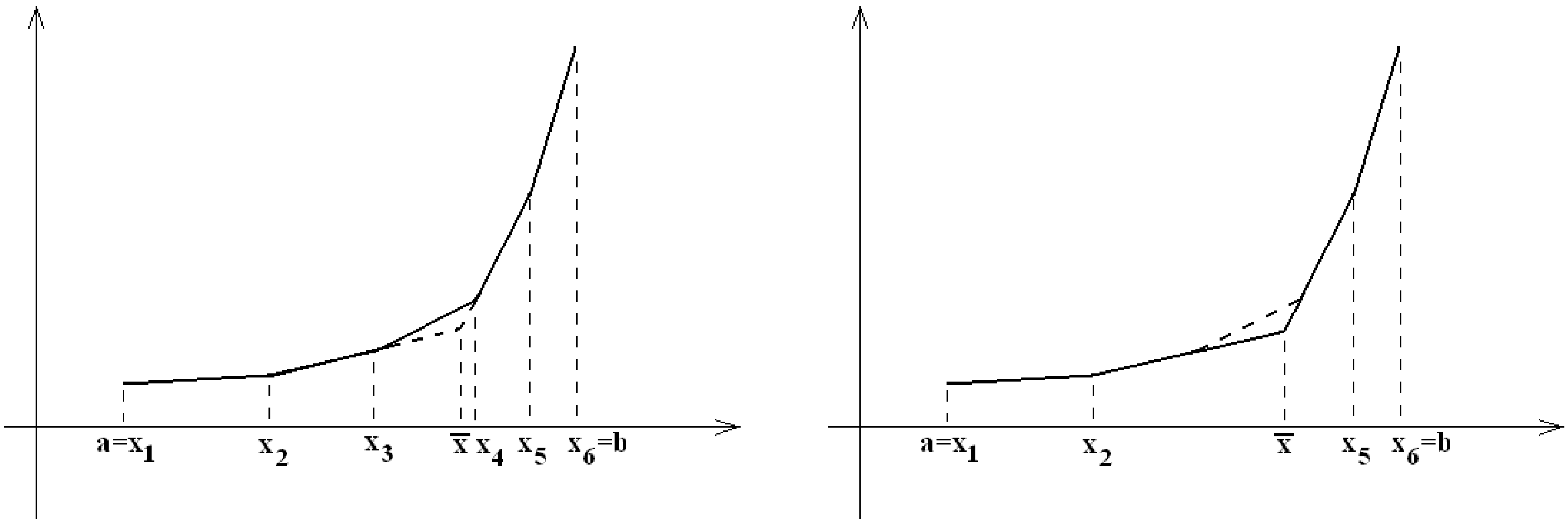


Figure 2: Lower estimate: x_3 and x_4 have been removed, $\overline{x}$ has been inserted.

Fixed strike European Call Asian options

- We will give a continuous representation of the option price function at every node of the tree as a piecewise linear convex function of the average.
- The price function at every node of the tree is characterized only by its singular points.
- Backward induction algorithm.

Notations

- Let us denote by $N_{i,j}$ the node of the tree whose underlying is $S_{i,j} = s_0 u^{2j-i}$, $i = 0, ..., n$, $j = 0, ..., i$.
- We will associate to each node $N_{i,j}$ a set of singular points, whose number is $L_{i,j}$. The singular points will be denoted by

$$(A_{i,j}^l, P_{i,j}^l), \quad l = 1, ..., L_{i,j}.$$

Backward algorithm: at maturity n

- At every node the average values vary between a minimum average $A_{n,j}^{min}$ and a maximum average $A_{n,j}^{max}$.
- For every $A \in [A_{n,j}^{min}, A_{n,j}^{max}]$ the price of the option can be continuously defined by $v_{n,j}(A) = (A - K)_+$.
- The function $v_{n,j}(A)$ is a piecewise linear and convex function whose singular points are easily valuable.

Critical points at maturity n

- if $K \in (A_{n,j}^{min}, A_{n,j}^{max})$ then the price value function $v_{n,j}(A)$ is characterized by the 3 singular points $(A_{n,j}^{l}, P_{n,j}^{l})$, $l = 1, 2, 3$ $(L_{n,j} = 3)$, where

$$\begin{array}{lll} & A_{n,j}^{1} = A_{n,j}^{min}, & P_{n,j}^{1} = 0; \\ (3) & A_{n,j}^{2} = K, & P_{n,j}^{2} = 0; \\ & A_{n,j}^{3} = A_{n,j}^{max}, & P_{n,j}^{3} = A_{n,j}^{max} - K. \end{array}$$

.

- if $K \notin (A_{n,j}^{min}, A_{n,j}^{max})$ then the price value function $v_{n,j}(A)$ is characterized by the 2 singular points $(A_{n,j}^{l}, P_{n,j}^{l})$, $l = 1, 2$, $(L_{n,j} = 2)$, where

$$(4) \qquad \begin{array}{ll} A_{n,j}^{1} = A_{n,j}^{min}, & P_{n,j}^{1} = (A_{n,j}^{min} - K)_{+} \ ; \\ A_{n,j}^{2} = A_{n,j}^{max}, & P_{n,j}^{2} = (A_{n,j}^{max} - K)_{+} \ . \end{array}$$

- In the case $j = 0$ and $j = n$ the minimum and maximum of the averages coincide and $L_{n,j} = 1$.

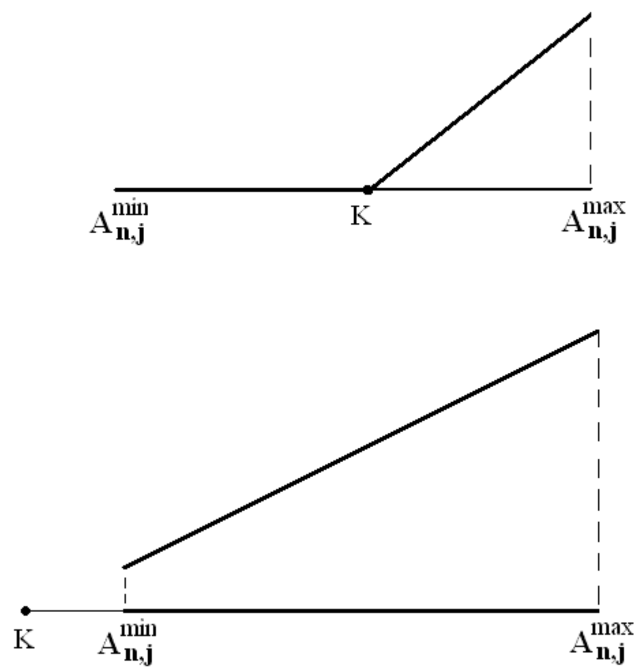


Figure 3: Singular points at maturity

Backward algorithm

Consider now the step i, $0 \le i \le n-1$.

Lemma 3 *At every node $N_{i,j}$, $i = 0, ..., n$, $j = 0, ..., i$, the function $v_{i,j}(A)$ which provides the price of the option as function of the average A, is piecewise linear and convex in the interval $[A_{i,j}^{min}, A_{i,j}^{max}]$.*

The evaluation of the singular points can be done recursively by a backward algorithm.

The claim is true at step $i = n$ (at maturity).
At step $i = n-1$, the price function $v_{i,j}(A)$, with $A \in [A_{i,j}^{min}, A_{i,j}^{max}]$, is obtained by considering the discounted expectation value:

$$v_{i,j}(A) = e^{-r\frac{T}{n}}[\pi v_{i+1,j+1}(A') + (1-\pi)v_{i+1,j}(A'')], \tag{5}$$

where

$$A' = \frac{(i+1)A + S_0 u^{2j-i+1}}{i+2}, \qquad A'' = \frac{(i+1)A + S_0 u^{2j-i-1}}{i+2}. \tag{6}$$

As $v_{n,j}(A)$ is piecewise linear and convex in his domain and
$h_1(A) = v_{i+1,j+1}(\frac{(i+1)A+S_0 u^{2j-i+1}}{i+2})$ is the composite function of a linear function of A and a piecewise linear convex one, $h_1(A)$ is piecewise linear and convex as function of A.
The same holds true for $h_2(A) = v_{i+1,j}(\frac{(i+1)A+S_0 u^{2j-i-1}}{i+2})$. We can conclude that $v_{i,j}(A)$ is piecewise linear and convex in his domain.

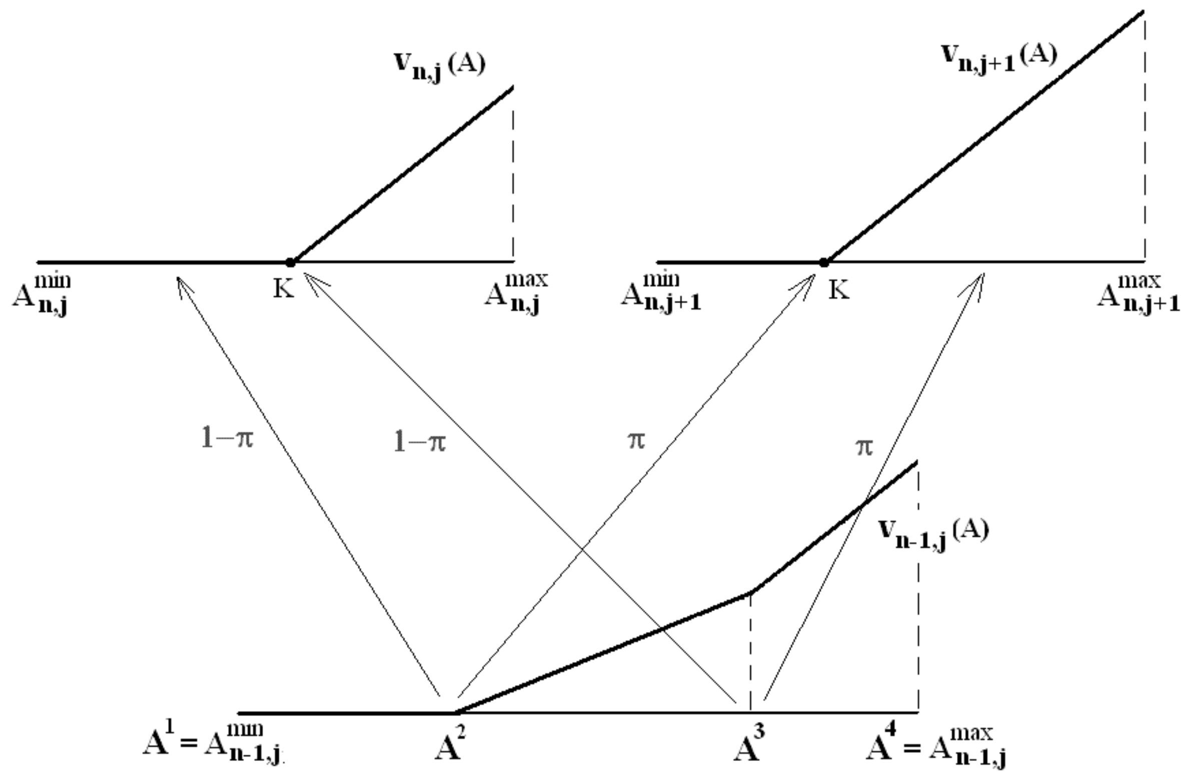


Figure 4: Singular points at i=n-1

Singular points at n-1

- Each singular average $A^l_{i+1,j}$, $l = 1, \ldots, L_{i+1,j}$ of the node $N_{i+1,j}$ is projected in a new average value B^l at the node $N_{i,j}$ by

$$B^l = \frac{(i+2)A^l_{i+1,j} - s_0 u^{2j-i-1}}{i+1}. \tag{7}$$

- Let $B^l \in [A^{min}_{i,j}, A^{max}_{i,j}]$. After a down movement of the underlying, B^l transforms into $A^l_{i+1,j}$, which price is $P^l_{i+1,j}$.
- Consider now an up movement of the underlying. In this case B^l transforms into the average: $B^l_{up} = \frac{(i+1)B_l + s_0 u^{2j-i+1}}{i+2}$. Using linear interpolation (the function is linear!) we obtain$P^l_{i+1,j+1}$.
- We can evaluate the price associated to the singular average B^l evaluating the discounted expectation value:

$$v_{i,j}(B^l) = e^{-r\Delta T}[\pi v_{i+1,j+1}(B^l_{up}) + (1-\pi)v_{i+1,j}(A^l_{i+1,j})]. \tag{8}$$

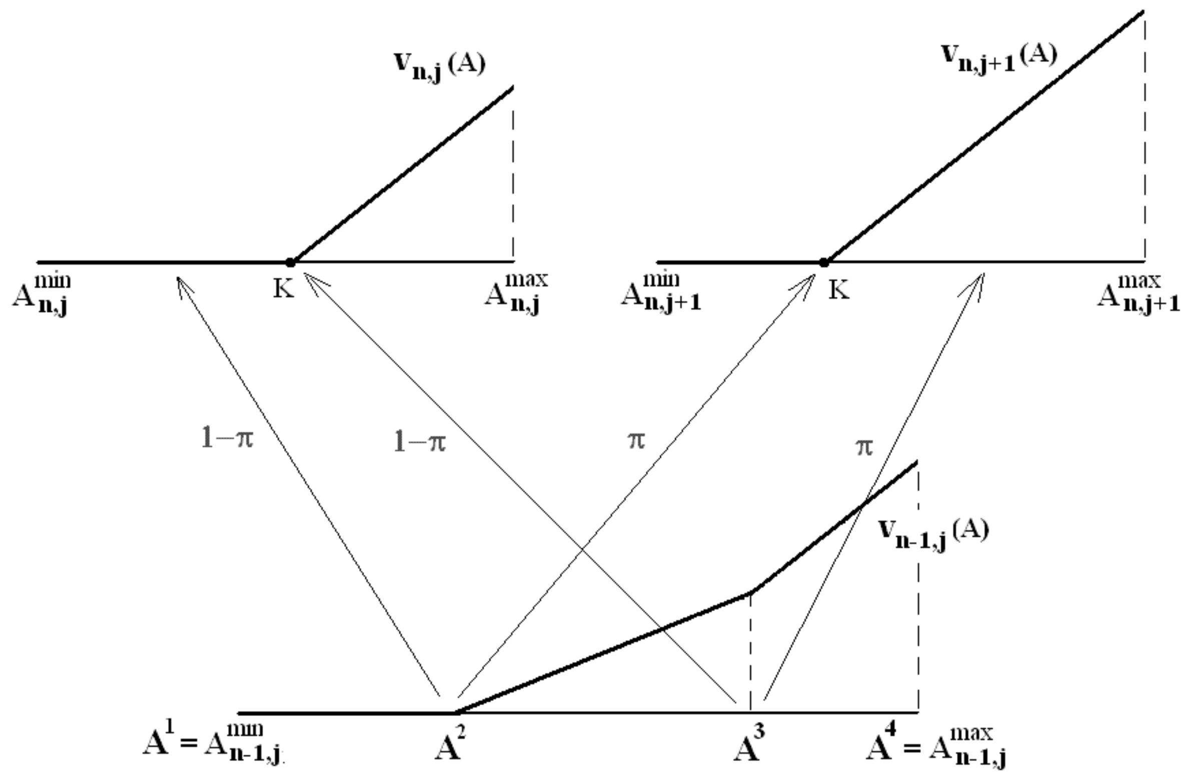


Figure 5: Singular points at i=n-1

- In a similar way each singular average $A^l_{i+1,j+1}$, $l = 1, ..., L_{i+1,j+1}$ associated to the node $N_{i+1,j+1}$ is projected in a new average C^l at the node $N_{i,j}$
- We can evaluate the corresponding price $v_{i,j}(C^l)$ in a similar way as before.
- Finally we proceed by a sorting of the averages B^l and C^l belonging to $[A^{min}_{i,j}, A^{max}_{i,j}]$, obtaining an ordered set $\{(A^l_{i,j}, P^l_{i,j}), ..., (A^{L_{i,j}}_{i,j}, P^{L_{i,j}}_{i,j})\}$ of singular points at the node $N_{i,j}$.
  These are exactly all the singular points associated to this node.

Extreme nodes

At the nodes $N_{i,i}$, $N_{i,0}$, there is only a singular point whose price is given by

$$P^1_{i,0} = e^{-r\Delta T}[\pi P^1_{i+1,0} + (1-\pi)P^1_{i+1,1}], \tag{9}$$

$$P^1_{i,i} = e^{-r\Delta T}[\pi P^1_{i+1,i+1} + (1-\pi)P^{L_{i+1,i}}_{i+1,i}]; \tag{10}$$

The value $P^1_{0,0}$ is exactly the binomial price relative to the tree with n steps of the fixed strike European Asian call option.

Fixed strike American call Asian options

- To taking into account the American feature

$$v_{i,j}(A) = \max\{v^c_{i,j}(A), A - K\}.$$

- $v_{i,j}(A)$, $A \in [A^{min}_{i,j}, A^{max}_{i,j}]$, is still a piecewise linear convex function.
- For this reason we can characterize it again by its singular points

Suppose that $A_{i,j}^{max} - K > v_{i,j}^c(A_{i,j}^{max})$ and $A_{i,j}^{min} - K < v_{i,j}^c(A_{i,j}^{min})$.
Then there exist an unique average $\overline{A}$ where the continuation value is equal to the early exercise.

Let j_0 be the largest index such that $A_{i,j}^{j0} < \overline{A}$. The new set of singular points becomes:

$$\{(A_{i,j}^1, P_{i,j}^1), ..., (A_{i,j}^{j0}, P(A_{i,j}^{j0})), (\overline{A}, \overline{A} - K), (A_{i,j}^{max}, A_{i,j}^{max} - K)\}.$$

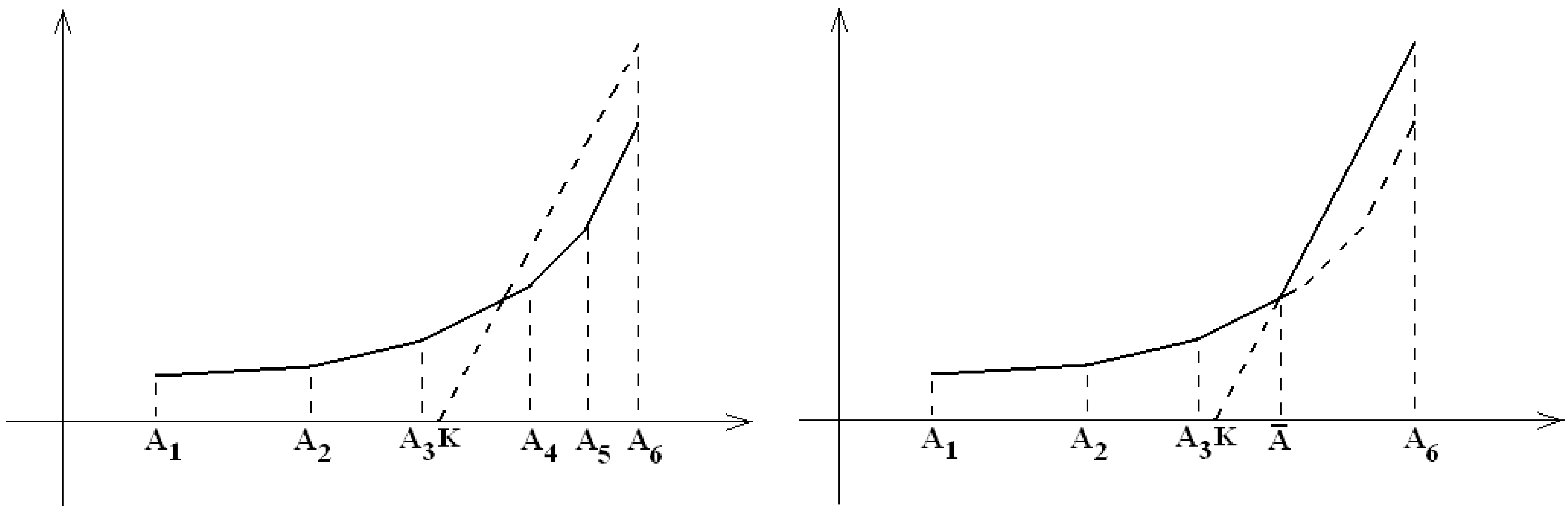
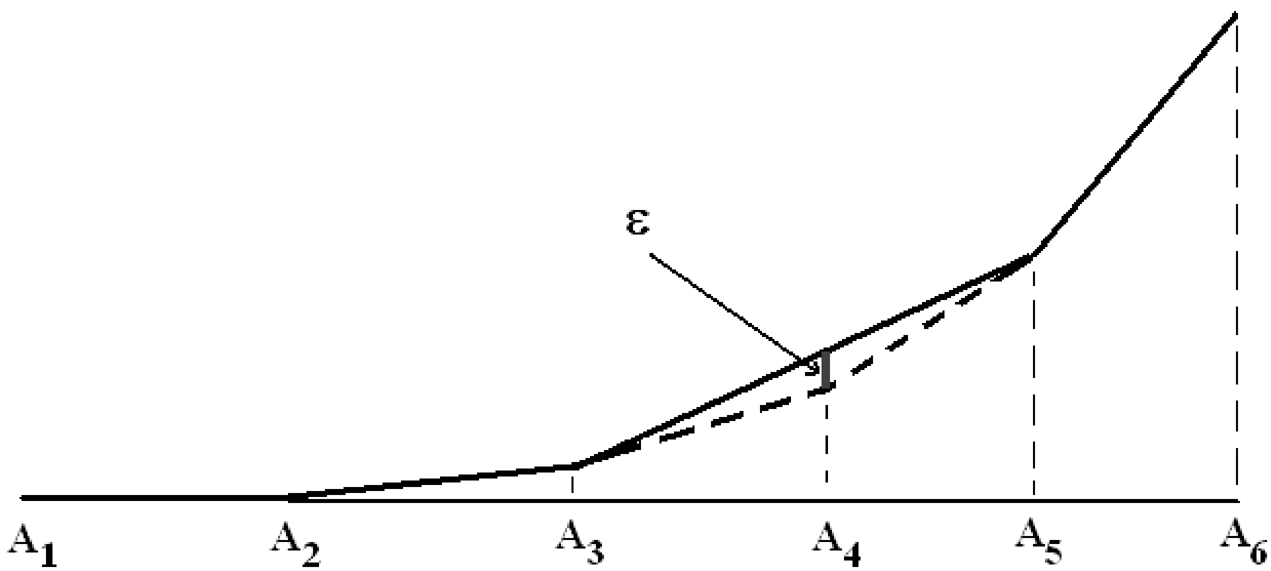


Figure 6: The point $\overline{A}$ has been inserted, A_4 and A_5 have been removed.

Upper and lower bounds

- The resulting algorithm can be of exponential complexity as the standard binomial technique.
- We are able to compute an upper and a lower bound of the binomial price reducing drastically the amount of time computation and the memory requirement.
- An a-priori control of the distance of the estimates from the pure binomial price.

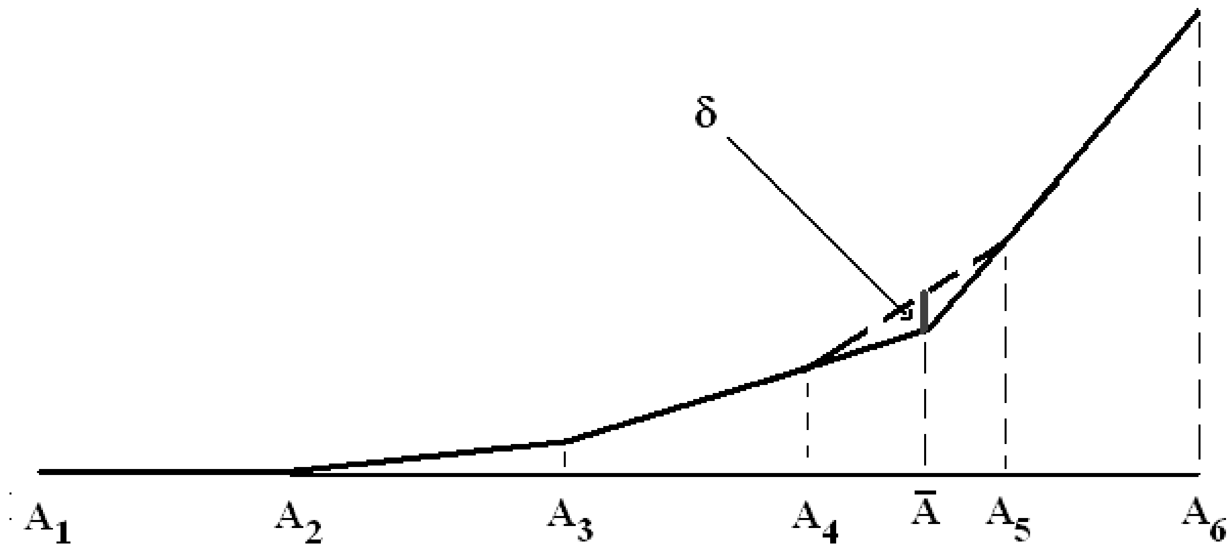
UPPER BOUND



Remove A_4 if $\epsilon \leq h$

Inductively we get that the obtained upper estimate differs from the binomial value at most for nh.

LOWER BOUND



Remove A_4 and A_5 and insert $\overline{A}$ if $\delta \leq h$

Inductively we get that the obtained lower estimate differs again from the binomial value at most for nh

Convergence results

Remark 1 *Jiang and Dai (SIAM Journal on numerical analysis 2005) proved the convergence of the exact binomial algorithm for European/ American path-dependent options. In particular they proved that the rate of convergence of the exact binomial algorithm to the continuous value is* $O(\Delta T)$.

The possibility of obtaining estimates of the exact binomial price with an error control allows us to prove easily the convergence of our method to the continuous value. Choosing h *depending on* n *and so that* $nh(n) \to 0$ *we have that the corresponding sequences of upper and lower estimates converge to the continuous price value. Moreover, choosing* $h(n) = O(\frac{1}{n^2})$, *we are able to guarantee that the order of convergence is* $O(\Delta T)$.

Numerical Results Fixed strike American Call Asian options

- We illustrate numerically the efficiency of singular points method.
- We compare the singular points algorithm with Hull-White,Barraquand-Pudet,Chalasani et al.
- We assume that the initial value of the stock prices are $s_0 = 100$, the maturity $T = 1$, the continuous dividend rates $q = 0.03$, while the values of the volatility $\sigma = 0.2, 0.4$, the interest rate $r = 0.1$, and the exercise price $K = 90, 100$ vary.
- We consider different time steps $n = 25, 50, 100, 200, 400, 800$

1. the pure binomial(PB) model (available only for $n = 25$),
2. the Hull-White method (HW) with $h = 0.005$,
3. the forward shooting grid method (FSG) of Barraquand-Pudet with $\rho = 0.1$,
4. the Chalasani et al. method (CJEV) that provides an upper and a lower bound, (available only for $n = 25, 50, 100$),
5. the singular points method providing an upper and a lower bound with error less than nh, for two different choices of h:
   - $h = 10^{-4}$ (SP_1);
   - $h = 10^{-5}$ (SP_2).

Analysis of convergence

1. the PDE-based method of d'Halluin et al. (DFL) available for both the European and the American Asian options;

2. the PDE-based method of Vecer available in the European Asian option case ;

3. the modified linear interpolation forward shooting grid method (M-FSG) of Barraquand-Pudet. We chose $\rho = 0.1$ and $n\sqrt{n}$ grid points in the Asian direction in order to guarantee the convergence (see the Premia implementation www.premia.fr);

4. the modified FSG algorithm with the Richardson extrapolation (M-FSG-Rich);

5. the singular points method (SP) providing an upper bound with a level of error smaller than nh with $h = \frac{0.1}{n^2}$ (see Remark 1);

In the European case we used the two-points extrapolation $2P_n - P_{\frac{n}{2}}$, whereas in the American case the three points extrapolation $\frac{8}{3}P_n - 2P_{\frac{n}{2}} + \frac{1}{3}P_{\frac{n}{4}}$ was adopted.
In order to compare the convergence behavior we consider the convergence ratio R

$$R = \frac{P_{\frac{n}{2}} - P_{\frac{n}{4}}}{P_n - P_{\frac{n}{2}}}$$

Lookback options

The price of an European lookback option is given by

$$P(0, s, s) = E\left[e^{-rT} f(S_T, M_T) | S_0 = s, M_0 = s\right].$$

where M_T

$$M_T = \max_{0 \le t \le T} S_t$$

$$m_T = \min_{0 \le t \le T} S_t$$

Payoff example:

- Fixed Lookback Call: the payoff is $(M_T - K)_+$.
- Fixed Lookback Put: the payoff is $(K - m_T)_+$.
- Floating Lookback Call: the payoff is $(S_T - m_T)_+$.
- Floating Lookback Put: the payoff is $(M_T - S_T)_+$.

Binomial method

$$E\left[e^{-rT}f(S^N, M^N)\right].$$

where

$$M^N = \max_{0\le n\le N} S^n$$

Pure Binomial method

The maximum process $(M_i)_{0 \leq i \leq n}$ can be computed recursively by

$$M^{n+1} = \max(M^n, S^{n+1}), M^0 = s_0$$

The bidimensional transition matrix is given by

$$\begin{array}{llcll} \text{up} & (x, y) & \to & (xu, \max(xu, y)) & \text{with probability } q \\ \text{down} & (x, y) & \to & (xd, y) & \text{with probability } 1-q \end{array}$$

Backward induction

$$\begin{cases} v(N, x, y) = f(x, y) \\ v(n, x, y) = e^{-r\Delta T}\Big[qv(n+1, xu, \max(xu, y)) + (1-q)v(n+1, xd, y)\Big], \end{cases}$$

Rem In the American case we have to take in account the early exercise $(y-k)_+$

Complexity The evaluation of $v(0, s_0, s_0)$ requires a number of computations of order n^3.

Implementation of the algorithm

Number of different maximum at every node (n, j)

$$\begin{cases} j+1 & j \leq \frac{n}{2} \\ n-j+1 & j > \frac{n}{2}, \end{cases}$$

FSG Method Forward Shooting Grid Method of Barraquand-Pudet for both Fixed or Floating Strike cases.

$S_j^n = s_0 e^{j\sigma\sqrt{h}}, M_k^n = s_0 e^{k\sigma\sqrt{h}} \qquad j,k \ = -n,...,n$ where $n = N,..,0.$

If at time n the bidimensional process is at (S_j^n, M_k^n), at time n+1 the process can reach in the upward and downward transition cases

$$\begin{array}{lllll} \text{up} & (S_j^n, M_k^n) & \to & (S_{j+1}^{n+1}, M_{k+}^{n+1}) & \text{with probability } p_u \\ \text{down} & (S_j^n, M_k^n) & \to & (S_{j-1}^{n+1}, M_{k-}^{n+1}) & \text{with probability } p_d \end{array}$$

$$\begin{cases} C_{j,k}^N = \psi(S_j^N, M_k^N) = (M_k^N - K)_+ \\ C_{j,k}^n = \max\left(\psi(S_j^n, M_k^n), e^{-r\Delta T}\left[p_u C_{j+1,k+}^{n+1} + p_d C_{j-1,k-}^{n+1}\right)\right] \end{cases} \tag{11}$$

Remark 1 Time complexity of FSG algorithm is $O(N^3)$ and the convergence is slow

Babbs method Babbs gives a very efficient and accurate solution to the problem with an one-dimensional tree method in the case of American floating strike Lookback options. The main idea is to use a change of "numeraire" approach using a reflected barrier.

$$Y_t = \frac{M_t}{S_t}$$

$$Y_{n+1} = \begin{cases} uY_n \quad with \quad p_u \\ \max(dY_n, 1) \quad with \quad p_d \end{cases} \tag{12}$$

Remark Time complexity of Babbs algorithm is $O(N^2)$ and the convergence with reflected barrier is very fast for the price