Fast binomial procedures for pricing Parisian/ParAsian options

Marcellino Gaudenzi, Antonino Zanette

Fast binomial procedures for pricing Parisian/ParAsian options

Marcellino Gaudenzi Antonino Zanette *

Abstract

The discrete procedures for pricing Parisian/ParAsian options depend, in general, on three dimensions: time, space, time spent over the barrier. Here we present some combinatorial and lattice procedures which reduce the computational complexity to second order. In the European case the reduction was already given by Lyuu-Wu [11] and Li-Zhao [10], in this paper we present a more efficient procedure in the Parisian case and a different approach (again of order 2) in the ParAsian case.
In the American case we present new procedures which decrease the complexity of the pricing problem for the Parisian/ParAsian knock-in options.
The reduction of complexity for Parisian/ParAsian knock-out options is still an open problem.

Keywords: Parisian options; ParAsian options; tree methods; binomial methods; combinatorial formula.
2000 MSC: 91G60, 60H35, 60C05.

Introduction

Parisian options are barrier options which can be knocked in or out depending on the time that the underlying asset has spent over a barrier. Such a time can be counted either consecutively or cumulatively. In the former case (Parisian contracts) the clock counting the time is reset as soon as the underlying asset goes down the barrier, in the latter case (ParAsian contracts) the clock is not reset but continues ticking as long as the underlying asset is beyond the barrier.
There are several approaches for pricing Parisian and ParAsian options. Haber and al. [9] and Vetzal-Forsyth [12] introduced a Partial Differential Equation (PDE). They considered a three dimensional PDE problem (time, asset price and time spent over the barrier) which can be solved using finite differences or finite elements methods. The PDE approach covers both Parisian and ParAsian options as well as the case of early exercise features (American options).
A different approach consists in the use of binomial or trinomial lattice techniques. In the Cox, Ross, Rubinstein [4] framework, a binomial tree can incorporate the Parisian/ParAsian feature by considering the paths for which the duration condition is satisfied.
In the discrete technique there are different ways to count the time spent over the barrier: we can count the nodes of the path which lies over the barrier ("counting nodes" approach) or the number of time steps which stays completely over the barrier ("counting steps"). In the Parisian case the two approaches are equivalent, but in the ParAsian case the counting methods are different.
Avellaneda-Wu [1] use a convergent trinomial lattice in order to price European Parisian options. The procedure, as in the PDE case, has computational complexity of order $O(n^3)$, where n is the number of

*Dipartimento di Scienze Economiche e Statistiche, Via Tomadini 30/A, Università di Udine, 33100 Udine, Italy
marcellino.gaudenzi@uniud.it and antonino.zanette@uniud.it

time steps of the tree. More recently Lyuu-Wu [11] proposed a combinatorial binomial approach which permits to obtain a procedure of order $O(n^2)$ for the European Parisian case, improving the algorithm of Costabile [3] of complexity $O(n^3)$. All such techniques cannot be applied to the ParAsian case. Li-Zhao [10] provided a new combinatorial approach based on generating functions and the Chung-Feller counting theorem. The Chung-Feller Theorem can be applied since they use the "counting steps" approach. They obtain a procedure of order $O(n^{\frac{5}{2}})$ in the Parisian case and of order $O(n^2)$ in the ParAsian case.
We propose here a different binomial tree combinatorial approach which allows us to obtain a procedure of order $O(n^2)$ both for European Parisian. and ParAsian options. In the ParAsian case we use a "counting node approach" which seems more related to the discrete framework, proving a procedure which is completely different from the one of Li-Zhao [10].
Furthermore we are able to treat the Parisian/ParAsian knock-in American case with a second order complexity procedure as well. The possibility of reducing the complexity of the algorithms to the second order in the Parisian/ParAsian knock-out American case remains an open problem.
It is well know that the use of the binomial method for pricing barrier options is problematic from the computational point of view. Costabile [3] and Lyuu-Wu [11] use the Boyle-Lau [2] technique in order to overcome such a problem. This method, based on the idea to choose trees with a line of nodes closest as possible to the barrier, permits to obtain sufficiently precise estimates, but it is problematic in the case of a barrier closed to the initial value of the underlying asset ("near barrier problem"). Here we use the algorithm proposed in Gaudenzi-Zanette [8] in order to further increase the efficiency of the numerical procedures presented for the Parisian/ParAsian options. Such an algorithm is based on a backward procedure where the nodes of the tree are generated from the barrier and it permits to overcome the problem related to the specification error of the barrier (see Figlewsky-Gao [6]) and to treat the near barrier problem in a natural way.
The paper is organized as follows: in Section 1 we present the model of the risk asset and the option pricing problem; in Section 2 we present the proposed algorithms for the European Parisian/ParAsian options; in Section 3 we introduce the algorithm for pricing Parisian/ParAsian American knock-in options. Finally, in Section 4, we provide a comparison of the results obtained by such techniques with the tree methods and finite difference methods proposed in the past literature.

1 Model and tree structure

1.1 The model

In this paper, we consider a market model where the evolution of a risky asset is governed by the Black-Scholes stochastic differential equation

$$\frac{dS_t}{S_t} = (r-\delta)dt + \sigma dB_t, \quad S_0 = s_0, \tag{1}$$

where $(B_t)_{0\leq t\leq T}$ is a standard Brownian motion, under the risk neutral measure Q. The nonnegative constant r is the interest rate, σ is the volatility of the risky asset and δ is the continuous dividend yield. T is the time to maturity and we will denote by K the strike of the option.
For pricing Parisian/ParAsian options in this lognormal model, we consider now a binomial approach. Let n be the number of steps of the binomial tree and $\Delta T = T/n$ the corresponding time-step.
The standard discrete binomial process is given by

$$S_{(i+1)\Delta T} = S_{i\Delta T} Y_{i+1}, \quad 0 \leq i \leq n-1,$$

where the random variables $Y_1, \dots, Y_n$ are independent and identically distributed with values in the set $\{u, d\}$. Let us denote by $\pi = \mathbb{P}(Y_n = u)$ the probability of an up jump and by $\rho = e^{-r\Delta T}$ the discount factor.
The Cox-Ross-Rubinstein tree (see [4]) corresponds to the choice $u = \frac{1}{d} = e^{\sigma\sqrt{\Delta T}}$ and

$$\pi = \frac{e^{(r-\delta)\Delta T} - e^{-\sigma\sqrt{\Delta T}}}{e^{\sigma\sqrt{\Delta T}} - e^{-\sigma\sqrt{\Delta T}}}.$$

We shall consider Parisian-style options (which means Parisian or ParAsian options) with barrier B and window period W. For sake of conciseness, in the sequel we shall consider only "up" barriers, the cases of "down" barrier can be treated in a similar way. In the Parisian knock-out case the barrier option vanishes if the price of the underlying asset remains for a period longer than the window period W over the barrier. In the knock-out ParAsian case the barrier option expires if the total time spent over the barrier is greater than W (hence the "Parisian" price is greater than the "ParAsian" price). Correspondingly in the knock-in case the previous windows period conditions activate the barrier. In the American case the option may be exercised at every time for which the Parisian-style contract is active.
Parisian-style options need pricing algorithms of barrier options. For this purpose we use a tree structure adapted to this case.

1.2 Tree structure for barrier options

The tree structure previously introduced requires some adjustments in the barrier case. In this section we will discuss the binomial tree that will be used for our pricing problems. We will use the approach introduced in [8] in order to treat efficiently the problem of the specification error on the barrier due to the binomial method (see [2]).
We assume that the number n of time steps of the tree is even and we construct a tree with nodes whose underlying is

$$S_{i,j} = Bu^{2j-i}, \text{ with } i = 0, ..., n.$$

In the usual CRR tree one has $j = 0, ..., i$ whereas here j has a different range since the tree is enlarged and translated (see Figure 1). In fact we need that at time 0, s_0 lies between four nodes of the tree, two over s_0 and two under s_0, in order to perform a four points interpolation at s_0 which will provide the price of the option.
To this end we denote by j_S the largest even integer j such that $Bu^j \leq s_0$. Then we take

$$j_{min} = \frac{j_S}{2} \tag{2}$$

so that j at time step i varies between $j_{min} - 1$ and $j_{min} + i + 2$.
In this way at time $t = 0$ we will obtain four nodes $S_{0,j_{min}+j}$, $j = -1, 0, 1, 2$, with underlying assets: Bu^{j_S+2j}, $j = -1, 0, 1, 2$. The price at s_0 will be computed by interpolating (using Lagrange interpolation) the option prices evaluated in these four nodes at s_0. When $j_S = -2$ [resp. $j_S = 0$] hence $Bu^{-2} \leq s_0 < B$ [$B \leq s_0 < Bu^2$], we will interpolate using only the 3 points of the underlying asset Bu^{-4}, Bu^{-2}, B [B, Bu^2, Bu^4]. This simple approach permits us to treat easily and efficiently the 'near-barrier' problem, that occurs when the initial asset price is very close to the barrier.

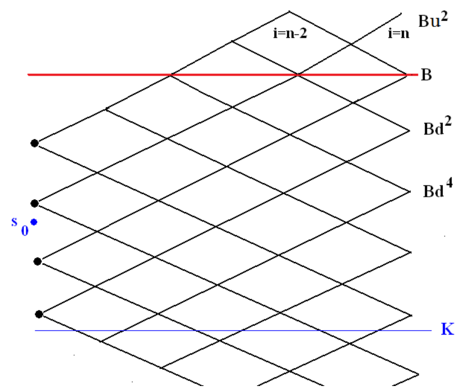

Figure 1: The tree structure

2 Second order binomial algorithms for Parisian/ParAsian options in the European case

We propose here binomial procedures of computational complexity $O(n^2)$ for European Parisian and ParAsian options. In the Parisian case we propose a modification of the Avellaneda-Wu [1] tree method which permits us to reduce the complexity of the algorithm to the order 2. A second order procedure in this case has been already obtained by a different technique by Lyuu-Wu [11]. On the contrary the Avellaneda-Wu and Lyuu-Wu procedures are not applicable to the ParAsian case. Moreover in the ParAsian case there are not lattice algorithms available in literature with complexity $O(n^2)$. The finite difference procedure of Haber et al [9], solving the PDE associated to the ParAsian case is of order $O(n^3)$. To this purpose, we introduce here a new backward programming algorithm, of complexity $O(n^2)$, which exploits appropriate combinatorial formulas available in Gaudenzi [7].
All the procedures here introduced could be applied on a standard CRR tree, but we will apply it to the tree described in the previous section, in such a way we can evaluate the price of the Parisian-style options with a more precise technique.
We will consider the cases of European Parisian and ParAsian up-and-out call options. The knock-and-in case can be easily obtained by the parity conditions for knock-and-in and knock-and-out barrier options holding in the European case, in the case of Parisian/ParAsian options the only difference is that we have to consider the sum of an up-and-out call option with time period W and an up-and-in call option with the same time period W. The treatments for down or put options are similar.
In the sequel we set

$$l = int(\frac{W}{T}n)$$

l representing the window period in the discrete setting. We also assume $l < n$. This means that a knock-out Parisian option will be active if there are not more than l nodes (counted either consecutively or cumulatively) lying over the barrier. "Over the barrier" means the weak inequality, that is either on the barrier or strictly over the barrier. Similarly a knock-in Parisian option will become active if there are at least $l + 1$ nodes lying over the barrier.

2.1 European Parisian options

We will use a backward dynamic programming procedure which considers only the nodes of the tree lying either exactly on the barrier or below the barrier, with a particular treatment in the case for

which $s_0 > B$.
For the nodes strictly below the barrier we simply use the standard backward induction procedure

$$v_i(S) = \rho[\pi v_{i+1}(Su) + (1-\pi)v_{i+1}(Sd)] \tag{3}$$

where $v_i(S)$ denotes the European Parisian option price at time i associated to the node with underlying asset S.
For the nodes lying exactly on the barrier (where the recursion formula (3) does not hold) we will compute the option price value by considering all possible paths starting from these nodes and reaching a node lying below the barrier with a number of steps less or equal to l.
To this end we first need to evaluate the number of the paths which stay over a barrier. At first we consider the number of all paths starting from the level Bu^m, arriving at level Bu^j, $m, j \geq 0$, after s time steps and staying always over the barrier lying at the level B. Such number, which can be evaluated by the "reflection principle" (see Feller [5]) from which we get

$$\binom{s}{\frac{s+m-j}{2}} - \binom{s}{\frac{s+m+j+2}{2}}. \tag{4}$$

We must have $s+m+j$ even otherwise the final node is not reachable and $|m-j| \leq s$. Moreover, if $m+j+2 > s$ all the paths lie always over the barrier and the second binomial coefficient in (4) vanishes.
In the particular case $j = 0$ we get

$$B_{s,m} = \binom{s}{\frac{s+m}{2}} - \binom{s}{\frac{s+m+2}{2}} = \frac{2m+2}{s+m+2}\binom{s}{\frac{s+m}{2}}. \tag{5}$$

Here, we have used the same notations as in [7]. When $m = 0$, s must be even, so by replacing s with $2s$ the previous formula becomes

$$c_s = \frac{1}{s+1}\binom{2s}{s}. \tag{6}$$

c_s counts the number of all possible paths of $2s$ steps, starting and arriving on the barrier, which stays always over the barrier.
Now we are able to evaluate the price at the nodes on the barrier. Consider all the paths starting at node $(2i, i)$ (with underlying asset B) and arriving at node $(2(i+s)+1, i+s)$ (with underlying asset Bd), $s = 0, ..., int(\frac{l-1}{2})$, lying always over the barrier until the next to last node $(2(i+s), i+s)$. Such number of paths is c_s. The price of the option at the node $(2i, i)$ consists in the sum of the value of the option at the node $(2(i+s)+1, i+s)$ (computed by the backward induction) multiplied by the number of all the possible paths arriving at such a node and staying always over the barrier until the node $(2i, i)$ (with underlying B), multiplied by the corresponding probabilities and discounted at time $2i$. Therefore the price of the known-and-out Parisian option at the node $(2i, i)$ can be computed by the formula

$$v_{2i}(B) = \sum_{s=0}^{L} c_s\ \rho^{2s+1}\pi^s(1-\pi)^{s+1}v_{2(i+s)+1}(Bd) \quad \text{where } L = int(\frac{l-1}{2}) \tag{7}$$

We can now provide the pricing algorithm for the European Parisian options in the case $s_0 \leq B$:

1. at each node at maturity we set the option price as

$$v_n(S) = \max\{S-K, 0\} \tag{8}$$

2. for $i = n-1, ..., n-l+1$ (where the Parisian contract is surely active) we use standard backward induction (3) at all the nodes.

3. for $i = n-l, ..., |j_S| - 4$
   - at all nodes lying strictly under the barrier ($S < B$) we use the backward induction (3),
   - for all the nodes lying exactly on the barrier ($S = B$) we use the pricing formula (7).

4. for $i = -j_S - 3, ..., 0$ we use the backward induction (3) (here all the nodes are below the barrier).

5. we interpolate the four points $(Bu^{j_S+2j}, v_0(Bu^{j_S+2j}))$, $j = -1, 0, 1, 2$ at s_0, obtaining in such way the price of the option.

In the case $s_0 > B$ the step 4 (consisting in the backward induction procedure) has to be replaced by the direct computation of $v_0(Bu^{2j})$, $j = \frac{j_S}{2} - 1, \frac{j_S}{2}, \frac{j_S}{2} + 1, \frac{j_S}{2} + 2$. By virtue of (5) such prices can be obtained by

$$v_0(Bu^{2j}) = \sum_{s=j}^{L} B_{2s,2j}\ \rho^{2s+1}\pi^{s-j}(1-\pi)^{s+j+1}v_{2s+1}(Bd) \tag{9}$$

Remark 1 *In the cases $j_S = -2$, $j_S = 0$ we will use a three points interpolation in order to take into account the near barrier problem (see Section 2).*

The described algorithm has time complexity $O(n^2)$ and space complexity $O(n)$. In fact the backward induction procedure has these complexities. Moreover each computation formula (7) needs a linear number of operations. In fact, one has

$$v_{2i}(B) = \rho(1-\pi)\sum_{s=0}^{L} c_s\alpha_s v_{2(i+s)+1}(Bd) \tag{10}$$

where the coefficients c_s, α_s, $s = 0, ..., L$ can be computed recursively by:

$$c_0 = 1, \qquad c_{s+1} = \frac{4s+2}{s+2}c_s; \qquad \alpha_0 = 1, \quad \alpha_{s+1} = \rho^2\pi(1-\pi)\alpha_s.$$

The same holds true for the formula (9). In fact

$$v_0(Bu^{2j}) = \rho\frac{(1-\pi)^{j+1}}{\pi^j}\sum_{s=j}^{L} c'_s\alpha_s v_{2(i+s)+1}(Bd) \tag{11}$$

where the coefficients $c'_s = B_{2s,2j}$, $s = j, ..., L$ can be computed recursively by:

$$c'_j = 1, \qquad c'_{s+1} = \frac{(2s+2)(2s+1)}{(s+j+2)(s-j+1)}c'_s \tag{12}$$

Note that the computations of c_s, α_s do not depend on time step i, hence they can be evaluated once at the beginning of the whole procedure.

2.2 European ParAsian options

The procedure use a scheme similar to the Parisian case with an important modification of the computations of the prices on the barrier.
For the nodes strictly below the barrier, as before, we use the backward induction procedure (3). In the case of the nodes lying exactly on the barrier the direct application of the previous procedure to the ParAsian case, leads to a procedure of computational order $O(n^3)$, therefore a completely different approach is needed here.
First we remark that in the case of Parisian option we have counted the number of time steps for which the paths stay over the barrier while here we will count the number of nodes of the path lying over the barrier. If a path stays for l consecutive steps over the barrier then it has $l+1$ nodes over the barrier, therefore in the ParAsian case it will be natural to use $l+1$ as counting index instead of l. For sake of simplicity of notation we still use l in all our formulas, but in the implementation of the procedures l must be substituted by $l+1$.
Let us consider the the number $\Theta_{2m}(l)$ which counts all paths of $2m$ steps starting from the node $(2i,i)$ on the barrier, arriving at the node $(2(i+m), i+m)$ and having not more than l nodes which stay over the barrier (the first and the last node of the path are counted as well). By Corollary 2 of [7]) (see Case 1) such number is

$$\Theta_{2m}(l) = \sum_{\substack{s=0,\dots,l-2 \\ s\ even}} (l-1-s) B_{2m-2-s,0} B_{s,0} = \begin{cases} \sum_{s=0}^{L_1} (2L_1+1-2s) c_{m-s-1} c_s & \text{if } l \text{ is even} \\ \sum_{s=0}^{L_1} (2L_1+2-2s) c_{m-s-1} c_s & \text{if } l \text{ is odd} \end{cases} \quad (13)$$

where $L_1 = int(\frac{l}{2}) - 1$.
In [7] the computation of $\Theta_{2m}(l)$ has been deduced from the computation of the number $T_{2m}(l)$ of all the paths of $2m$ steps starting, as before, from the node $(2i,i)$ on the barrier, arriving at the node $(2(i+m), i+m)$ and having exactly l nodes which stay over the barrier. Such number (see Case 1 of Theorem 1 in [7]) is

$$T_{2m}(l) = \sum_{s=0}^{L_1} c_{m-s-1} c_s \quad (14)$$

In fact, by the previous equation one has

$$T_{2m}(0) = T_{2m}(1) = 0, \qquad T_{2m}(2j) = T_{2m}(2j+1) = T_{2m}(2j-2) + c_{m-j} c_{j-1} \ j = 1, ..., m. \quad (15)$$

$$\Theta_{2m}(0) = T_{2m}(0) = 0, \qquad \Theta_{2m}(j) = \Theta_{2m}(j-1) + T_{2m}(j) \ j = 1, ..., 2m+1. \quad (16)$$

Consider again the node $(2i,i)$ lying on the barrier. Given a path γ starting from this node and arriving at maturity, we consider the largest index m, $i \le m \le \frac{n}{2}$, such that the node $(2m,m)$ belongs to the path γ. We call this node the "exit node" of the path γ.
The set of all paths starting from the node $(2i,i)$ and arriving at maturity will be partitioned in disjoint subsets Γ_s, $s = i, ..., \frac{n}{2}$, whose elements are the paths having $(2s,s)$ as exit node.
We now consider, for $s < \frac{n}{2}$ the two subsets of Γ_s: Γ_s^{down} which consist of all paths of Γ_s which stay always strictly under the barrier after their passage at the exit node, Γ_s^{up} which consist of all the paths of Γ_s staying always strictly over the barrier after their passage at the exit node. Then we denote by $\Gamma_{s,l}^{down}$ [$\Gamma_{s,l}^{up}$] the set of all the paths of Γ_s^{down} [Γ_s^{up}] whose trajectory between $(2i,i)$ and $(2s,s)$ has not more than l nodes which stay over the barrier. The number of paths of $\Gamma_{s,l}^{down}$ is equal to $\Theta_{2s-2i}(l)$.
Every path of $\Gamma_{s,l}^{down}$ [resp. $\Gamma_{s,l}^{up}$] after the exit node $(2s,s)$ always remains strictly under [over] the barrier. Hence his contribution to the ParAsian option price at the node $(2i,i)$ is related to the price

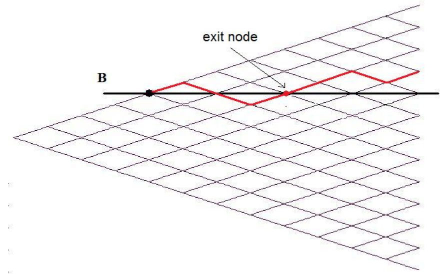

Figure 2: Exit node of a path starting from the barrier

of an European knock-out up [down] standard barrier option which has $(2s+1, s)$ $[(2s+1, s+1)]$ as initial node, $n-2s-1$ time steps and barrier B.
Therefore we consider a tree of n time steps (of the same form as before), the barrier B, and the backward induction binomial procedure needed for pricing a standard European knock-out up [down] barrier option. If we denote by $v_i^{up}(S)$ $[v_i^{down}(S)]$ the price value function of this barrier option at time steps i and asset price S, the values $v_{2i+1}^{up}(Bd)$ $[v_{2i+1}^{down}(Bu)]$ are the prices needed in the computation of the ParAsian option.
With the previous notations, one has that the price at the node $(2i, i)$ of our ParAsian option is then given by $v_{2i}(B) = P_{2i}(l)$, where

$$P_{2i}(l) = \rho(1-\pi) \sum_{s=0}^{n/2-i-1} \Theta_{2s}(l)\rho^{2s}[\pi(1-\pi)]^s v_{2(i+s)+1}^{up}(Bd) +$$

$$\rho\pi \sum_{s=n/2-i+1-int(l/2)}^{n/2-i-1} \Theta_{2s}(l-n+2i+2s)\rho^{2s}[\pi(1-\pi)]^s v_{2(i+s)+1}^{down}(Bu) + \tag{17}$$

$$\Theta_{n-2i}(l)\rho^{n-2i}[\pi(1-\pi)]^{n/2-i} \max(B-K, 0)$$

The first term of this formula is the sum of all the contributions of the paths of Γ_s^{down} with $s < \frac{n}{2}$. The second term is the sum of all the contributions of the paths of Γ_s^{up} with $s < \frac{n}{2}$ (in this case the path has $n-2i-2s$ nodes lying over the barrier after the exit node). The third term is the contribution of the paths of $\Gamma_{n/2}$.
When $s_0 \leq B$ the pricing algorithm for the European ParAsian options is similar to the one described in the Parisian case, the pricing formula (7) is replaced by the pricing formula (17).
When $s_0 > B$ the algorithm in the ParAsian case has to be changed with respect to the Parisian case since it is not possible to use the value of the option below the barrier. In fact, a path which starts over the barrier, when arriving at a node under the barrier has already spent a certain time over the barrier, so the price at the node does not coincide with the price of the option starting from that node which is the price evaluated by the backward procedure. So we apply a direct forward procedure in order to evaluate the four prices $v_0(Bu^{2j})$, $j = \frac{js}{2}-1, \frac{js}{2}, \frac{js}{2}+1, \frac{js}{2}+2$ needed for the interpolation. More precisely we have to consider the contribution to the price $v_0(Bu^{2j})$, $j > 0$, of all the paths starting to this node, arriving at the node $(2s, s)$ of the barrier and lying always strictly over the barrier before such node. For the price at this node we can consider the previous formula (17) where the discrete time l has to be decreased by the time already spent over the barrier, so that, by virtue

of (5) we obtain

$$v_0(Bu^{2j}) = \sum_{s=j}^{L} B_{2s-2,2j-1}\ \rho^{2s}\pi^{s-j}(1-\pi)^{s+j}P_{2s}(l-2s) \tag{18}$$

The time complexity of the procedure is $O(n^2)$. In fact the backward induction procedure has this time complexity. Moreover each computation formula (17) needs a linear number of operations if a preprocessing procedure consisting in the computation of the matrix $\Theta_{2s}(j)$, $s = 1, ..., \frac{n}{2}$, $j = 1, ..., 2s+1$ is performed. By virtue of the recursive relations (15) and (16), this matrix can be computed with a second order complexity algorithm before the pricing procedure. The computation of formula (18) has complexity $O(n^2)$ using the computation formula (17) which is linear. In fact the formula (18) can be written

$$v_0(Bu^{2j}) = (\frac{1-\pi}{\pi})^j \sum_{s=j}^{L} c''_s \alpha_s P_{2s}(l-2s) \tag{19}$$

where the coefficients c''_s can be computed recursively in a similar way as in (12).
The space complexity of the procedure, as described previously, is $O(n^2)$. However it can be reduced to $O(n)$ with some suitable choices, like to store only $\Theta_s(j)$, $j = 0, ..., s+1$, and to evaluate the sums until this term, then we evaluate $\Theta_{s+1}(j)$ and so on.

3 Binomial method of second order in the American knock-in case

In the discrete framework an American Parisian [ParAsian] knock-in option is an option which becomes active when the underlying asset presents l consecutive [cumulative] nodes of the path over the barrier. If the option becomes active at time step $i \leq n$ and has underlying asset S, the holder got a plain-vanilla American option with $n-i$ time steps to maturity and initial underlying asset S. We denote by $v_i^{amer}(S)$ the price of this option. Let us remark that by a unique standard backward binomial procedure, of order $O(n^2)$, we can obtain the values of v_i^{amer} at every node of the tree.

3.1 American Parisian knock-in

In order to evaluate American Parisian knock-in options we will use a scheme similar to those described in the case of European Parisian options (see Section 2.1).
For our pricing formulas we need to evaluate the number of paths of the tree starting from the level B, arriving at the level Bu^s after $l-1$ time steps ($l+s$ odd) and never lying under the level B. Such number is equal to $B_{l-1,s}$ and it has been already computed in (5).
The price of a knock-in Parisian option on the nodes of the barrier is computed by the formula

$$v_{2i}(B) = \sum_{\substack{s=0,\ldots,l-1\\ s+l\ odd}} B_{l-1,s}\ \rho^{l-1}\pi^{\frac{l+s-1}{2}}(1-\pi)^{\frac{l-s-1}{2}}v^{amer}_{2i+l-1}(Bu^s) + \sum_{s=0}^{L_1} c_s\ \rho^{2s+1}\pi^s(1-\pi)^{s+1}v_{2(i+s)+1}(Bd), \tag{20}$$

where $L_1 = int(\frac{l}{2}) - 1$. The first sum considers all contributions given by the paths starting from the node $(2i, i)$ and having l consecutive nodes (i.e. $l-1$ time steps) which lie over the barrier. These contributions take into account the plain-vanilla American option price at the node of the paths where

the option becomes active. The second sum considers all the contributions given by the paths starting from the node $(2i, i)$ and arriving below the barrier with less than $l-1$ time steps (see Figure 3).
As in the case of European Parisian options, when $s_0 > B$ we need to compute $v_0(Bu^{2j})$, $j = \frac{js}{2}-1, \frac{js}{2}, \frac{js}{2}+1, \frac{js}{2}+2$. Now we have:

$$v_0(Bu^{2j}) = \sum_{\substack{s=j_0,\dots,2j+l-1 \\ s+l \ odd}} \left(\binom{l-1}{\frac{l+2j-s-1}{2}} - \binom{l-1}{\frac{l+2j+s+1}{2}} \right) \rho^{l-1} \pi^{\frac{l+s-2j-1}{2}} (1-\pi)^{\frac{l-s+2j-1}{2}} v_{l-1}^{amer}(Bu^s)$$
$$+ \sum_{s=j}^{L_1} B_{2s,2j}\ \rho^{2s+1} \pi^{s-j} (1-\pi)^{s+j+1} v_{2s+1}(Bd), \quad (21)$$

where $j_0 = \max(2j - l + 1, 0)$. The first sum considers all the contributions given by the paths lying always over the barrier for the first $l-1$ time steps and the number of such paths is evaluated by virtue of (4). The second sum is the equivalent of (9).
The numerical algorithm proposed in the European Parisian case can be modified in the following way. The steps 1 and 2 are replaced by the terminal condition $v_{n-l+1}(S) = 0$ at all the nodes. In the Step 3 the pricing formula (7) is replaced by (20). In the case $s_0 > B$, in Step 4 we use formula (21) instead of (9).
Using similar arguments as in the European Parisian case we can state that the time complexity is again $O(n^2)$ and the space complexity in $O(n)$.

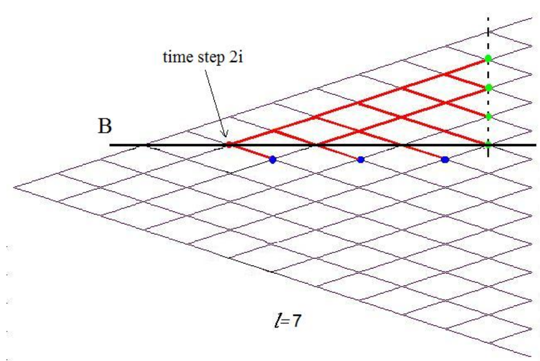

Figure 3: Price at the node of the barrier in the case of American Parisian up-and-in options

3.2 American ParAsian knock-in

In this case we use a different procedure: we evaluate by combinatorial formulas directly the price without considering the backward induction.
We can obtain procedures of order 2 by using the binomial formulas introduced in [7] which allows to count the number of paths having exactly l nodes, counted cumulatively, lying over the barrier. To this end we consider the coefficients $B_{s,k}$ defined in (5). Such coefficients can be evaluated, for all s, k, $s, k = 0, ..., n$, by a procedure of order 2 in a easy way. In fact we have

$$B_{2s,0} = c_s, \qquad B_{s,s} = 1,$$

$$B_{s,k} = 0, \qquad \text{if } s+k \text{ is odd},$$

$$B_{s,k} = B_{s-1,k-1} + B_{s-1,k+1}, \qquad \forall k \in \{1, ..., s-1,\ s+k \text{ even}\}. \tag{22}$$

We need to evaluate the number $T_{n,j,m}(l)$, i.e. the number of all paths of n steps starting from the initial position, whose underlying is Bu^j, arriving at level Bu^m and having exactly l nodes over the barrier. We remark that the number $T_n(l)$, previously introduced (see (14), is equal to $T_{n,0,0}(l)$. We consider only paths whose position of arrival is over the barrier, hence $m \geq 0$, while j can be both negative and positive.

By Theorem 1 of [7] we get

$$T_{n,j,m}(l) = \begin{cases} \displaystyle\sum_{\substack{s=0,\dots,n-1-l \\ s-j \text{ even}}} B_{n-2-s,m} B_{s,-j-2} - \sum_{\substack{s=0,\dots,n-1-l \\ s-j \text{ odd}}} B_{n-2-s,m-1} B_{s,-j-1}, & \text{if } j < -1 \text{ and } m > 0 \\ \displaystyle\sum_{\substack{s=0,\dots,n-1-l \\ n+s \text{ even}}} B_{n-2-s,0} B_{s,-j-2} & \text{if } j < -1 \text{ and } m = 0 \\ \displaystyle\sum_{\substack{s=0,\dots,l-2 \\ n+s \text{ even}}} B_{n-2-s,0} B_{s,m+j} & \text{if } j \geq 0 \text{ and } m \geq 0 \end{cases} \tag{23}$$

We will need to evaluate the difference between $T_{n,j,m}(l)$ and $T_{n,j,m}(l-1)$. By the previous equations we get

$$\begin{aligned} T_{n,j,m}(l) &= T_{n,j,m}(l-1) + \begin{cases} -B_{l-2,m} B_{n-l,-j-2} & \text{if } n+l-j \text{ is even} \\ B_{l-2,m-1} B_{n-l,-j-1} & \text{if } n+l-j \text{ is odd} \end{cases} && \text{if } j < -1 \text{ and } m > 0 \\ T_{n,j,m}(l) &= T_{n,j,m}(l-1) + \begin{cases} -B_{l-2,0} B_{n-l,-j-2} & \text{if } l \text{ is even} \\ 0 & \text{if } l \text{ is odd} \end{cases} && \text{if } j < -1 \text{ and } m = 0 \\ T_{n,j,m}(l) &= T_{n,j,m}(l-1) + \begin{cases} B_{n-l,0} B_{l-2,m+j} & \text{if } n+l \text{ is even} \\ 0 & \text{if } n+l \text{ is odd} \end{cases} && \text{if } j \geq 0 \text{ and } m \geq 0 \end{aligned} \tag{24}$$

In order to establish the price of an American ParAsian knock-in option we need to compute the coefficients $T_{n,j,m}(l)$ at all the nodes over the barrier. Formula (23) requires a number of computations of order $O(n)$, but the nodes involved are of order $O(n^2)$. Therefore, we have to modify the procedure in order to reduce the computational complexity to second order.

Assume first that the underlying asset of the initial position lies strictly under the barrier, i.e. $j < -1$. In order to develop the procedure we will use some properties of the numbers $T_{n,j,m}(l)$:

1. $T_{s,j,m}(l) = 0 \qquad$ if $s - j + m$ is odd.
2. $T_{s,j,l-1}(l) = B_{s-l,-j-1}$ if $s - j + l$ is odd.
3. $T_{s-1,j,l-2}(l-1) = B_{s-l,-j-1}$ if $s - j + l$ is odd.
4. $T_{s-1,j,m+1}(l-1) + T_{s-1,j,m-1}(l-1) = T_{s,j,m}(l)$ if $m > 0$.
5. $T_{s-1,j,m-1}(l-1) = T_{s,j,m}(l) - T_{s-1,j,m+1}(l-1)$ if $m > 0$.

The justifications of the properties are the following:

Property 1. If we start from the level Bu^j and we reach the level Bu^m in s time steps, then necessarily $j + m$ has the same parity of s.

Property 2. The highest level Bu^m reachable by a path which has exactly l nodes over the barrier is Bu^{l-1}. Such a level can be reached only if the path stays always strictly under the barrier for the first $s-l$ time steps and arrives, in $s-l$ time steps, at level Bu^{-1}. The number of such paths is $B_{s-l,-j-1}$.
Property 3. It is achieved from Property 2 substituting s, l by $s-1$, $l-1$ respectively.
Property 4. It is achieved from the definition of $T_{n,j,m}(l)$.
Property 5. It is achieved from Property 4.

By virtue of the previous properties we can evaluate all coefficients $T_{s,j,m}(l)$ for $0 \leq m \leq l-1$, $0 \leq s \leq n$, by a procedure of order $O(n^2)$ as follows:

Initialization:

- calculate first the coefficients $B_{s,k}$, $s = 0, ..., n$, $k = 0, ..., s$ by the scheme (22);
- calculate the numbers $T_{n,j,m}(l)$, $m = 0, ..., l-1$, with $m-j$ even, by (23) (here we are considering only the nodes at maturity);
- calculate the numbers $T_{s,j,l-1}(l)$, for $s = l-1-j, ..., n$ with $l-j+s$ odd, by Property 2 (here we are considering the highest reachable nodes);
- calculate the numbers $T_{s,j,l-2}(l-1)$, for $s = l-1-j, ..., n-1$ with $l-j+s$ even, by Property 3;

Backward calculation. For $s = n-1, n-2, ..., l-1-j$ do the following steps:

- calculate $T_{s-1,j,m-1}(l-1)$ from $T_{s-1,j,m+1}(l-1)$ and $T_{s,j,m}(l)$ for $m = l-2, l-3, ..., 1$, with $s-j+m$ even, by Property 5,
- calculate $T_{s-1,j,m}(l)$ from $T_{s-1,j,m}(l-1)$ for $m = 0, 1, ..., l-2$, with $s-j+m$ odd, by (24).

Finally we can obtain the price of the option with initial underlying Bu^j by

$$v_0(Bu^j) = \sum_{s=l-j-1,...,n} \rho^s \sum_{\substack{m=0,...,l-1 \\ s-j+m \ even}} T_{s,j,m}(l) \pi^{\frac{s+m-j}{2}} (1-\pi)^{\frac{s-m+j}{2}} v_s^{amer}(Bu^m) \tag{25}$$

Consider now the case $j \geq 0$.
If $j < l-1$ the procedure is similar to the previous, it just changes Property 2 and, consequently, Property 3. Now the option becomes active at the time step $s = l-1$ when the path remains always over the barrier (see formula (4)) or, eventually, at time steps $s \geq l$ when the path has at least one node strictly under the barrier. In this second case the highest reachable level for the underlying asset at time step s by a path having exactly l nodes over the barrier, is Bu^{l-j-2}. Such level is achievable just by paths which stay over the barrier for the first j steps and the last $l-j-2$ steps. Therefore, Property 2 becomes

$$T_{s,j,l-j-2}(l) = B_{s-l+2,0} = c_{(s-l+2)/2} \quad \text{if } s-l \text{ is even}, \quad s > l.$$

We can conclude that (25) becomes

$$v_0(Bu^j) = \sum_{s=l,...,n} \rho^s \sum_{\substack{m=0,...,l-j-2 \\ s+j+m \ even}} T_{s,j,m}(l) \pi^{\frac{s+m-j}{2}} (1-\pi)^{\frac{s-m+j}{2}} v_s^{amer}(Bu^m) +$$
$$\rho^{l-1} \sum_{\substack{m=0,...,j+l-1 \\ s-j+m \ even}} \left[\binom{l-1}{\frac{l-1+m-j}{2}} - \binom{l-1}{\frac{l+m+j+1}{2}} \right] \pi^{\frac{l-1+m-j}{2}} (1-\pi)^{\frac{l-1-m+j}{2}} v_{l-1}^{amer}(Bu^m)$$

If $j \geq l-1$ the procedure is simpler, in fact the option is surely activated just at time step $l-1$ and the calculus of the $v_0(Bu^j)$ depends only on the values of $v_{l-1}^{amer}(Bu^m)$. By virtue of (4) one has

$$v_0(Bu^j) = \rho^{l-1} \sum_{\substack{m=j-l+1,\dots,j+l-1 \\ s-j+m \ even}} \left[\binom{l-1}{\frac{l-1+m-j}{2}} - \binom{l-1}{\frac{l+m+j+1}{2}} \right] \pi^{\frac{l-1+m-j}{2}} (1-\pi)^{\frac{l-1-m+j}{2}} v_{l-1}^{amer}(Bu^m)$$

4 Numerical results

In this section we provide some numerical comparisons of the algorithms presented in the previous sections with the binomial method of Lyuu-Wu [11], the trinomial method of Avellaneda-Wu [1] and the finite difference algorithms of Haber et al. [9]. In order to test the efficiency we will consider the numerical experiments proposed in [1]. We will price Parisian style options with: volatility $\sigma = 0.13$, interest rate $r = 0.056$, continuous dividend yield $q = 0.007$, current stock price $s_0 = 1/120.5$, strike price $K = 1/125$, time to maturity $T = 0.5$ and barrier $1/110$. We illustrate the numerical results for different windows periods.
In the ParAsian case we provide a numerical comparison also with the combinatorial method of Li-Zhao [10] using the parameter and the data provided by the authors.
In order to obtain more precise approximations a time interpolation with respect to the windows period is necessary. In fact, for a small number of steps, the prices are very sensitive to the integer approximation $l = int(\frac{W}{T}n)$. Hence we can use a linear interpolation of the prices corresponding to the choice of the integers l and $l+1$. This adjustment will be used for our method only in the ParAsian case (both in the European and American case).
All the computations presented in the tables have been performed in double precision on a PC with a processor Centrino at 2.4 Ghz with 4 Mb of RAM.

4.1 European Parisian options

We compute the price of up-and-out Parisian call options with the following methods:

- the PDE finite difference method of Haber et al [9] of order 3 (HSW);
- the trinomial method of Avellaneda-Wu [1] of order 3 (AW);
- the forward binomial method of Lyuu-Wu [11] of order 2 (LW);
- the backward binomial method, introduced in Section 2.1, of order 2 (GZ).

We choose for the tree methods different time steps $n = 100, 200, 400, 800, 1600$. The Lyuu-Wu method requires a different choice of such numbers related to the Boyle-Lau technique. In the finite difference case we choose a mesh close to the corresponding time steps of the tree methods.
In Table 1 we report the price estimates for European Parisian options with time of computation in parentheses.
We provide a second table, Table 2, with the barrier near the initial spot value: $B = 1/120$. In this case we choose also a longer windows period $W = 10/360, 30/360$.

n	HSW		AW		LW		GZ	
	5 days	15 days	5 days	15 days	5 days	15 days	5 days	15 days
100	199 [.00035]	266 [.00067]	206 [.00025]	277 [.00027]	181 [.00008]	251 [.00009]	211 [.00017]	281 [.00019]
200	200 [.00198]	271 [.00281]	208 [.00061]	278 [.00082]	224 [.00027]	286 [.00041]	218 [.00051]	279 [.00063]
400	200 [.0054]	262 [.0139]	215 [.0021]	279 [.0034]	210 [.0010]	274 [.0019]	218 [.0021]	280 [.0021]
800	208 [.0273]	271 [.0772]	215 [.0085]	279 [.0194]	221 [.0041]	287 [.0112]	216 [.0083]	279 [.0087]
1600	210 [.1460]	273 [.4317]	215 [.0392]	279 [.1140]	215 [.0200]	282 [.0839]	215 [.0305]	280 [.0340]

Table 1: *European Parisian up-and-out call options with* $s_0 = 1/120.5$ *and barrier* $B = 1/110$ *(all the prices has been multiplied by* 10^6*, time in seconds)*

n	HSW		AW		LW		GZ	
	10 days	30 days	10 days	30 days	10 days	30 days	10 days	30 days
100	64 [.00055]	346 [.00123]	1067 [.00020]	1491 [.00020]	105 [.00008]	397 [.00013]	138 [.00015]	465 [.00016]
200	85 [.00233]	365 [.00609]	989 [.00053]	1407 [.00054]	109 [.00034]	409 [.00072]	139 [.00052]	474 [.00059]
400	114 [.0114]	424 [.0334]	921 [.0018]	1335 [.0019]	101 [.0013]	403 [.0051]	133 [.0020]	470 [.0022]
800	114 [.0622]	433 [.1852]	131 [.0122]	473 [.0479]	126 [.0071]	472 [.0369]	131 [.0082]	473 [.0085]
1600	124 [.3373]	455 [1.032]	131 [.0701]	473 [.3637]	118 [.0399]	448 [.2878]	131 [.0330]	473 [.0353]

Table 2: *European Parisian up-and-out call options with* $s_0 = 1/120.5$ *and barrier* $B = 1/120$ *(all the prices has been multiplied by* 10^7*, time in seconds)*

4.2 European ParAsian options

In this case both Avellaneda-Wu and Lyuu-Wu techniques are not available. Therefore we will compare, in Table 3, our technique of order 2 with the the PDE finite difference method of Haber et al [9] of order 3.

n	HSW			GZ		
	5 days	15 days	30 days	5 days	15 days	30 days
100	185 [.00040]	238 [.00094]	290 [.00160]	192 [.00136]	235 [.00136]	289 [.00137]
200	192 [.00162]	238 [.00437]	294 [.00876]	190 [.00594]	235 [.00598]	289 [.00596]
400	185 [.0084]	229 [.0241]	282 [.0482]	190 [.0180]	235 [.0182]	289 [.0182]
800	188 [.0467]	233 [.1356]	287 [.2691]	189 [.0724]	234 [.0736]	289 [.0740]
1600	188 [.2541]	234 [.7579]	287 [1.506]	189 [.2894]	234 [.2932]	289 [.3016]

Table 3: *European ParAsian up-and-out call options with* $s_0 = 1/120.5$ *and barrier* $B = 1/110$ *(all the prices has been multiplied by* 10^6*, time in seconds)*

Moreover we propose a table where we compare the generating functions method of Li-Zhao (LZ) for ParAsian options using the "counting steps" approach with our technique based on the "counting nodes" approach. Now the parameters are $\sigma = 0.2$, $r = 0.08$, $q = 0$, $s_0 = 100$, $K = 95$, $T = 1$, $B = 110$ and windows period 15 days. In this table we report as Benchmark the price, with the corresponding confidence interval, obtained with Monte Carlo method using 10^7 simulations and 720 time discretisation steps.

n	HSW	LZ	GZ	Monte Carlo
100	1.0667	1.0157	0.9103	
500	0.9365	0.9154	0.9113	
1000	0.9213	0.9039	0.9076	0.9149 (0.9132-0.9167)
1500	0.9165	0.9112	0.9078	
2000	0.9139	0.9065	0.9073	

4.3 American Parisian/ParAsian knock-in

Also in this case neither Avellaneda-Wu nor Lyuu-Wu techniques are available. Therefore we will price our technique only with the the PDE finite difference method.
In Table 4 we report American Parisian knock-and-in options prices, whereas in Table 5 we will consider the American ParAsian knock-and-in case.

n		HSW			GZ	
	5 days	15 days	30 days	5 days	15 days	30 days
100	395 [.00182]	325 [.00248]	259 [.00331]	391 [.00050]	328 [.00050]	256 [.00054]
200	393 [.00597]	326 [.00918]	256 [.01440]	384 [.00180]	323 [.00182]	257 [.00193]
400	389 [.0221]	324 [.0398]	256 [.0675]	390 [.0068]	325 [.0068]	255 [.0071]
800	387 [.0877]	324 [.2002]	255 [.3565]	388 [.0257]	323 [.0276]	255 [.0289]
1600	387 [.3962]	323 [.9976]	255 [1.861]	387 [.1051]	324 [.1098]	255 [.1120]

Table 4: *American Parisian up-and-in call options with* $s_0 = 1/120.5$ *and barrier* $B = 1/110$ *(all the prices has been multiplied by* 10^6*, time in seconds)*

n		HSW			GZ	
	5 days	15 days	30 days	5 days	15 days	30 days
100	444 [.00191]	388 [.00279]	332 [.00395]	422 [.00072]	374 [.00078]	319 [.00090]
200	437 [.00635]	388 [.01076]	330 [.01821]	417 [.00276]	371 [.00300]	316 [.00348]
400	394 [.0247]	349 [.0499]	295 [.0870]	415 [.0096]	369 [.0108]	315 [.0130]
800	405 [.1059]	360 [.2575]	305 [.4748]	414 [.0286]	369 [.0444]	314 [.0508]
1600	402 [.4992]	356 [1.339]	302 [2.537]	414 [.1598]	368 [.1876]	314 [.2138]

Table 5: *American ParAsian up-and-in call options with* $s_0 = 1/120.5$ *and barrier* $B = 1/110$ *(all the prices has been multiplied by* 10^6*, time in seconds)*

References

[1] M. Avellaneda, L.Wu, *Pricing Parisian-Style Options with a Lattice Method*, International Journal of Theoretical and Applied Finance, 2/1 (1999) 1-16.

[2] P.P.Boyle, S.H. Lau, *Bumping up against the barrier with the binomial method*, The Journal of Derivatives, 1 (1994) 6-14.

[3] M.Costabile, *Pricing Parisian options*, Decisions in Economics and Finance 25(2) (2002) 111-125.

[4] J.Cox , S.A.Ross, M.Rubinstein, *Option Pricing: A simplified appoach*, Journal of Financial Economics, 7 (1979) 229-264.

[5] W. Feller,*Introduction to Probability Theory and Its Applications*, 1 (3rd edition), Wiley 1968.

[6] S.Figlewski, B.Gao, *The Adaptive Mesh Model: a New Approach to Efficient Option Pricing*, Journal of Financial Economics 53 (1999) 313-351.

[7] M.Gaudenzi, *Counting paths on a chessboard with a barrier*, Nonlinear Anal. Hybrid Syst, 4(3) (2010) 432-440.

[8] M.Gaudenzi, A.Zanette, *Pricing American barrier options with discrete dividends by binomial trees*, Decisions in Economics and Finance 32(2) (2009) 129-148.

[9] R.J.Haber, P.J.Schonbucker, P.Wilmott, *Pricing Parisian Options*, The Journal of Derivatives 6 (1999) 71-79.

[10] B.Q.Li, H.J.Zhao, *Pricing Parisian Options by Generating Functions*, The Journal of Derivatives, Summer, (2009) 72-81.

[11] Y.D.Lyuu and C.W.Wu *An improved combinatorial approach for pricing Parisian options*, Decisions in Economics and Finance 33(1) (2010) 49-61.

[12] K.R.Vetzal, P.A.Forsyth, *Discrete Parisian and Delayed Barrier Options: A General Numerical Approach*, Advances in Futures and Options Research 10 (1999) 1-15.