Introduction to Tree methods for financial derivatives

Claude MARTINI

March 8, 1999

Contents

Premia 18

1 Trees for standard options

1.1 Cox-Ross-Rubinstein as an approximation to Black-Scholes

The generalized Cox-Ross-Rubinstein (CRR) model has an interest on his own as a basic model of stochastic discrete-time process for the underlying asset of a financial derivative. One of its main attractive feature is the easiness of standard option pricing by backward induction which relies on the possibility of performing a perfect hedge at every node of the tree (cf. The Generalized CRR Model). In fact the original motivation of Cox, Ross and Rubinstein was to approximate the Black-Scholes price of an option.

Let N denote the step number of the algorithm: the Black-Scholes dynamic under the risk-neutral probability is replaced by the dynamic of the generalized CRR scheme

$$\begin{array}{rcl} S_{n+1}(h) & = & u\,S_n(h) \text{ with proba. } p^* \\ & & dS_n(h) \text{ with proba. } 1-p^* \end{array}$$

with $h = \frac{T}{N}$, where T is the time to maturity (ie the current date is taken to be zero), with a suitable choice of the parameters:

$$u = e^{\sigma\sqrt{h}},\ d = e^{-\sigma\sqrt{h}},\ 1 + r = e^{\rho h}$$

Note that $p^* = \frac{1+r-d}{u-d}$ is the risk-neutral probability in the generalized CRR scheme.

1.1.1 Convergence of the marginal law at maturity

The first idea which comes to study the convergence of the algorithm is to compute the characteristic function of $S_N(h)$. In fact, it is easier with $\ln S_N(h)$.

Without loss of generality assume $S_0 = 1$. For $\lambda \in \mathbb{R}$

$$\begin{aligned} & E^{P^*}\left[\exp\left(i\lambda \ln S_N(h)\right)\right] \\ = \; & E^{P^*}\left[\exp\left(i\lambda \ln \prod_{n=0}^{N-1} \frac{S_{n+1}(h)}{S_n(h)}\right)\right] \\ = \; & E^{P^*}\left[\exp\left(i\lambda \ln \frac{S_1(h)}{S_0}\right)\right]^N \\ = \; & \left(p^* \exp\left(i\lambda\sigma\sqrt{h}\right) + (1-p^*)\exp\left(-i\lambda\sigma\sqrt{h}\right)\right)^N \end{aligned}$$

and since $p^* = \frac{1+r-d}{u-d} \sim \frac{1}{2} + \frac{\left(\rho - \frac{\sigma^2}{2}\right)}{2\sigma}\sqrt{h} + O\left(h\sqrt{h}\right)$

$$\begin{aligned} E^{P^*}\left[\exp\left(i\lambda \ln S_N(h)\right)\right] & \sim \left(1 + \left[i\lambda\left(\rho - \tfrac{\sigma^2}{2}\right) - \lambda^2\tfrac{\sigma^2}{2}\right]\tfrac{T}{N}\right)^N \\ & \to \exp\left(\left[i\lambda\left(\rho - \tfrac{\sigma^2}{2}\right) - \lambda^2\tfrac{\sigma^2}{2}\right]T\right) \\ & = E\left[\exp\left(i\lambda\left(\left(\rho - \tfrac{\sigma^2}{2}\right)T + \sigma B_T\right)\right)\right] \\ & = E^{P^*_{BS}}\left[\exp\left(i\lambda \ln S_T\right)\right] \end{aligned}$$

where P^*_{BS} is the risk-neutral Black-Scholes probability. Therefore UNDER THE RISK NEUTRAL MEASURES

$$S_N(h)^{CRR} \to S_T^{BS} \qquad \text{(Conv)}$$

in LAW as $N \to \infty$.

This grants the convergence of the price of standard european options with payoffs continuous and bounded, e.g. Put options. The convergence of the Call prices follows by Call-Put parity, since the CRR scheme satisfies the Call-Put parity relationship.

An instructive feature of this calculation is that the limiting law depends only on $p^* e^{i\lambda \ln(u)} + (1-p^*)\, e^{i\lambda \ln(d)}$ through its Taylor expansion up to $o(h)$. Thus u, d or/and p^* could be altered as long as the involved terms of the development are not modified.

1.1.2 Some remarks about the limit $N \to \infty$

Notice that the upper and lower value of the spot at maturity are $u^N = e^{\sigma\sqrt{T}\sqrt{N}}$ and $d^N = e^{-\sigma\sqrt{T}\sqrt{N}}$ whereas the ratio of 2 successive points is $\frac{u}{d} = e^{2\sigma\sqrt{h}} = e^{2\sigma\sqrt{\frac{T}{N}}}$. Thus at the same time, the scan of the law of $S_N(h)$ goes to $\mathbb{R}_+^*$ and the grid gets more and more dense. In the same way it is easy to show that the points visited by the process $S(h)$ will eventually become dense in $[0,t] \times \mathbb{R}_+^*$. Although it is elementary, this is a nice phenomenon.

1.1.3 Code implementation

It is very easy to design the corresponding code (cf Routine tr_coxrossrubinstein.c).

1.1.4 Convergence of the delta

Observe that

$$\begin{aligned}\Delta_0 S_0 &= \frac{C_{u,N-1} - C_{d,N-1}}{(u-d)} \\ &= (1+r)^{-(N-1)} \frac{E^{P_*}\left[\varphi\left(uS_0X\right)\right] - E^{P_*}\left[\varphi\left(dS_0X\right)\right]}{u-d}\end{aligned}$$

where X is a random variable independent from S_0 so that

$$\Delta_0 S_0 = (1+r)^{-(N-1)} E^{P_*}\left[\frac{\varphi\left(uS_0X\right) - \varphi\left(dS_0X\right)}{u-d}\right]$$

Assume now that φ is a C^1 function. Then

$$\varphi\left(uS_0X\right) - \varphi\left(dS_0X\right) = \int_d^u S_0 X \varphi'\left(aS_0X\right) da$$

so

$$\begin{aligned}\Delta_0 S_0 &= (1+r)^{-(N-1)} \frac{1}{(u-d)} \int_d^u E^{P_*}\left[S_0 X \varphi'\left(aS_0X\right)\right] da \\ &= (1+r)^{-(N-1)} E^{P_*}\left[S_0 X \varphi'\left(a\left(h\right) S_0X\right)\right]\end{aligned}$$

for some point $a\left(h\right)$ in $[d,u]$ by the mean value property. Assuming now that $y\varphi'\left(y\right)$ is a Lipschitz function we know that so is the CRR price with the same Lipschitz constant which doesn't depend on h. In particular the function $b \mapsto E^{P_*}\left[S_0 X \varphi'\left(bS_0X\right)\right]$ is uniformly continuous uniformly in h, therefore since $[d,u] \to [1,1]$ as $h \to 0$

$$\lim_{h\to 0} E^{P_*}\left[S_0 X \varphi'\left(a\left(h\right) S_0X\right)\right] = \lim_{h\to 0} E^{P_*}\left[S_0 X \varphi'\left(S_0X\right)\right] \tag{1}$$

Now by the standard convergence result

$$\begin{aligned}\lim_{h\to 0} E^{P_*}\left[S_0 X \varphi'\left(S_0X\right)\right] &= E^{P_* BS}\left[S_t \varphi'\left(S_t\right)\right] \\ &= e^{\rho T} S_0 \Delta_0^{BS}\end{aligned}$$

Therefore the delta does converge towards the Black-Scholes delta, at least for sufficiently smooth payoffs. For a Call (or Put) option the argument to get (1) must be slightly modified.

1.2 Variants of the CRR tree

We follow more or less closely the excellent review in [1] To achieve the convergence in law (Conv on page 3) many other choices of u and d and q (denoting the probability inside the tree) may be done, regardless of any arbitrage or financial consideration: the tree algorithm becomes a numerical approximation algorithm among other ones, the oly purpose is to get a good convergence to the limiting price and delta.

Note that all the following trees will remain recombining trees since this is true as soon as u and d remain constant within the tree. Only the choice of u, d and the probability is at hand here.

1.2.1 The Random Walk scheme

This scheme is implemented in Routine tr_euler_bs.c. As long as $S_T = S_0 \exp\left(\left(\rho - \frac{\sigma^2}{2}\right) T + \sigma B_T\right)$ a very natural choice is to approximate the Brownian motion B by the standard Random Walk. This leads to

$$u = e^{\left(\rho - \frac{\sigma^2}{2}\right)h + \sigma\sqrt{h}},\ d = e^{\left(\rho - \frac{\sigma^2}{2}\right)h - \sigma\sqrt{h}}$$

and $q = \frac{1}{2}$.

The algorithm to get the option price is the straightforward discretized version of the risk-neutral expectation:

$$C_n = e^{-\rho h}\left(\frac{1}{2}C_{u,n+1} + \frac{1}{2}C_{d,n+1}\right)$$

The convergence may be proved in the same way as before.

Notice that the discretized process is not a martingale.

1.2.2 The matching-3-moments scheme

This scheme is implemented in Routine tr_thirdmoment.c. An alternative route to convergence is the Central Limit Theorem. This leads to the idea of matching the mean and variance of the conditionnal laws of the approximating chain with those of the continuous process. These are denoted the "local consistency conditions". The equations that u, d, q should satisfy are

$$\begin{aligned} qu + (1-q)\,d &= e^{\rho h} \\ qu^2 + (1-q)\,d^2 - e^{2\rho h} &= e^{2\rho h}\left(e^{\sigma^2 h} - 1\right) \end{aligned}$$

Since one degree of freedom remains, a natural idea is to match also the third moment, which gives the equation

$$qu^3 + (1-q)\,d^3 = e^{3\rho h} e^{3\sigma^2 h}$$

The solution of this system is

$$\begin{aligned} u &= \frac{e^{\rho h}Q}{2}\left[1+Q+\sqrt{Q^2+2Q-3}\right] \\ d &= \frac{e^{\rho h}Q}{2}\left[1+Q-\sqrt{Q^2+2Q-3}\right] \\ q &= \frac{e^{\rho h}-d}{u-d} \end{aligned}$$

with $Q = e^{\sigma^2 h}$.

Notice that $ud = e^{2\rho h}Q^2 > 1$: this tree is not symmetric.

1.3 Trinomial trees

Along this line there is no need any longer to remain stucked with the discrete-time no-arbitrage constraint one node-two sons. We may well choose a 3-points scheme or p-points scheme or even a number of points depending on N (this is useful for other kinds of limiting continuous-time dynamics, like Lévy processes for instance). From the previous calculation it's easy to see that the points and probabilities of the chosen scheme should be constrained by:

$$\sum p_j \exp\left(i\lambda \ln u_j\right) = 1 + \left[i\lambda\left(\rho - \tfrac{\sigma^2}{2}\right) - \lambda^2\tfrac{\sigma^2}{2}\right]h + o\left(h\right)$$

in the sense that these conditions ensure the convergence (Conv on page 3). We'll see later that these conditions are equivalent to the local consistency conditions, and also that they ensure a convergence of a much more general type.

Note that from a computational point of view in order to get a recombining tree a condition like u_{j+1}/u_j independant of j should be imposed.

A feature common to all the trinomial trees is that they allow a more precise computation of the delta, gamma, theta of the option in a natural finite-difference-like manner.

1.3.1 Trinomial schemes with matching two first moments

Let $u > m > d$ be the possible values of $\frac{S_{n+1}(h)}{S_n(h)}$, with probabilities q_1, q_2, q_3 respectively.

In order to get a recombining tree we need only

$$ud = m^2$$

The two first moment matching conditions give

$$\begin{aligned} q_1 u + q_2 m + q_3 d &= e^{\rho h} \\ q_1 u^2 + q_2 m^2 + q_3 d^2 &= e^{2\rho h} Q \end{aligned}$$

Q as before.

Since

$$q_1 + q_2 + q_3 = 1$$

it is seen that two unknowns remain.

The solution corresponding to the additional constraint

$$q_1 = q_2 = q_3 = \frac{1}{3}$$

is

$$\begin{aligned} u &= V + \sqrt{V^2 - m^2} \\ d &= V - \sqrt{V^2 - m^2} \\ m &= \frac{e^{\rho h}(3 - Q)}{2} \end{aligned}$$

with $V = \frac{e^{\rho h}(3+Q)}{4}$.

Many other choices can be done.

1.3.2 The Kamrad-Ritchken tree

This tree is implemented in Routine tr_kamradritchken_bs.c. Kamrad and Ritchken choose to take a symmetric 3-points approximation to $\ln\left(\frac{S_h}{S_0}\right)$ and to match the 2 first moments of this quantity. More precisely, if v denote the upper state this leads to:

$$\begin{aligned} v(q_1 - q_3) &= \left(\rho - \tfrac{\sigma^2}{2}\right) h \\ v^2(q_1 + q_3) - v^2(q_1 - q_3) &= \sigma^2 h \end{aligned}$$

They further simplify the last equality-still maintaining an $o(h)$ matching of the variance in

$$v^2(q_1 + q_3) = \sigma^2 h$$

Note that it can be checked by the calculation of the characteristic function that this $o(h)$ matching property is enough.

By replacing v by $\lambda\sigma\sqrt{h}$ this leads to

$$\begin{aligned} q_1 &= \frac{1}{2\lambda^2} + \frac{\left(\rho - \frac{\sigma^2}{2}\right)\sqrt{h}}{2\lambda\sigma} \\ q_2 &= 1 - \frac{1}{\lambda^2} \\ q_3 &= \frac{1}{2\lambda^2} - \frac{\left(\rho - \frac{\sigma^2}{2}\right)\sqrt{h}}{2\lambda\sigma} \end{aligned}$$

The parameter λ appears as a free parameter of the geometry of the tree, which may be useful for some purposes. It is called the stretch parameter. The value $\lambda = 1.22474$ which corresponds to $q_2 = \frac{1}{3}$ is reported to be a good choice for an ATM Call (or Put).

1.4 Miscellaneous remarks

1.4.1 Local consistency and convergence in law

Let's come back to the equality:

$$\sum p_j \exp\left(i\lambda \ln u_j\right) = 1 + \left[i\lambda\left(\rho - \frac{\sigma^2}{2}\right) - \lambda^2\frac{\sigma^2}{2}\right]h + o\left(h\right) \tag{2}$$

and assume:

$$\begin{aligned} p_j &= p_{j,0} + p_{j,1}\sqrt{h} + p_{j,2}h + o\left(h\right) \\ u_j &= 1 + u_{j,1}\sqrt{h} + u_{j,2}h + o\left(h\right) \end{aligned}$$

Then obviously

$$\sum p_{j,0} = 1,\ \sum p_{j,1} = \sum p_{j,2} = 0$$

We have

$$\begin{aligned} \exp\left(i\lambda \ln u_j\right) &= \exp\left(i\lambda\left(u_{j,1}\sqrt{h} + u_{j,2}h - \frac{u_{j,1}^2}{2}h\right) + o\left(h\right)\right) \\ &= 1 + i\lambda u_{j,1}\sqrt{h} + \left(i\lambda\left(u_{j,2} - \frac{u_{j,1}^2}{2}\right) - \frac{\lambda^2}{2}u_{j,1}^2\right)h + o\left(h\right) \end{aligned}$$

and (2) is equivalent to:

$$\begin{aligned} \sum p_{j,0} i\lambda u_{j,1} &= 0 \\ \sum p_{j,0}\left(i\lambda\left(u_{j,2} - \frac{u_{j,1}^2}{2}\right) - \frac{\lambda^2}{2}u_{j,1}^2\right) + \sum p_{j,1} i\lambda u_{j,1} &= \left[i\lambda\left(\rho - \frac{\sigma^2}{2}\right) - \lambda^2\frac{\sigma^2}{2}\right] \end{aligned}$$

and as long as p and u are real-valued:

$$\begin{aligned} \sum p_{j,0}u_{j,1} &= 0 \\ \sum p_{j,0}\left(u_{j,2}-\frac{u_{j,1}^2}{2}\right)+\sum p_{j,1}u_{j,1} &= \left(\rho-\frac{\sigma^2}{2}\right) \\ \sum p_{j,0}u_{j,1}^2 &= \sigma^2 \end{aligned} \tag{3}$$

or

$$\sum p_{j,0}u_{j,1} = 0 \tag{4}$$

$$\begin{aligned} \sum p_{j,0}u_{j,2}+\sum p_{j,1}u_{j,1} &= \rho \\ \sum p_{j,0}u_{j,1}^2 &= \sigma^2 \end{aligned} \tag{5}$$

Let's look now at the local consistency equations for S :

$$\begin{aligned} \sum p_j u_j &= \exp(\rho h)+o(h) \\ \sum p_j u_j^2 &= \exp\left(\left(2\rho+\sigma^2\right)h\right)+o(h) \end{aligned}$$

which may be written down the following way:

$$\begin{aligned} \sum p_{j,0}u_{j,1} &= 0 \\ \sum p_{j,0}u_{j,2}+\sum p_{j,1}u_{j,1} &= \rho \\ 2\sum p_{j,0}u_{j,1} &= 0 \\ \sum p_{j,0}u_{j,1}^2+2\sum p_{j,0}u_{j,2}+2\sum p_{j,1}u_{j,1} &= \left(2\rho+\sigma^2\right) \end{aligned}$$

or alternatively

$$\begin{aligned} \sum p_{j,0}u_{j,1} &= 0 \\ \sum p_{j,0}u_{j,2}+\sum p_{j,1}u_{j,1} &= \rho \\ \sum p_{j,0}u_{j,1}^2 &= \sigma^2 \end{aligned}$$

so that local consistency is equivalent to (4).

1.4.2 Exact martingalian schemes

If the scheme at hand is designed to compute an expectation under some risk-neutral probability, it might be a clever choice to choose an approximating probability which is also risk-neutral for the discrete-time Makov chain for any N and not only asymptotically. The building of the tree may thus be easily checked by using the (discrete-time) optional sampling theorem: for a martingale X and every uniformly bounded stopping time τ, $E[X_\tau] = X_0$. From a practical point of view, this also means that the output prices of the scheme will be arbitrage-free, there will be no numerically-generated arbitrage.

1.4.3 Flat trees and American options

The algorithm for pricing American options is the natural backward scheme:

$$C_n = \max\left(\varphi\left(S_n\right),\, e^{-\rho h}\sum q_i C_{n+1,u_i}\right) \tag{6}$$

It gives the exact price in the CRR model for the case of the CRR tree, this may be proved directly very easily.

The algorithm requires the computation of the intinsic value at each node. A computationnal interest of a flat tree (like Kamrad-Ritchken or CRR) is therefore that it allows the computation of the intrinsic values across all the possible values of the underlying (either $2N+1$ or $N+1$ for the above trees) before performing the backward scheme. This may reduce dramatically the computationnal cost of the algorithm.

2 Trees for barrier options

2.1 Inaccuracy of the direct method

Let's consider only the case of a Down-and-Out Call with a constant rebate R attached to the limit L. The first idea to price this option within the CRR scheme is to apply directly the backward recurrence scheme. In fact it is possible to show by calculus (although it is a bit tedious) that the obtained price shall converge to the right Black-Scholes limit. Nevertheless, it is observed that the convergence is very bad compared with that for standard options. 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 $S_0 d^{n_L+1}$. Therefore the convergence can not be faster than, roughly speaking, $\frac{\partial C_{BS}}{\partial L}\left(d^{n_L} - d^{n_L+1}\right) \backsim \frac{c}{\sqrt{N}}$, (where C_{BS} denotes the Black-Scholes price of the option) whereas the convergence for standard European options is known to be of order $\frac{1}{N}$.

An alternative method is to feed the algorithm with the right value of the barrier. We shall discuss 2 different ways to do this.

2.2 The Bardhan-Derman-Kani-Ergener algorithm

This algorithm is implemented in Routine tr_dermankani.c. The method is the following: consider a node just before breaching from above the barrier, that is at level $S_0 d^{n_L}$. Let us rename mod= $S_0 d^{n_L}$ as the "modified barrier"

and eff=$S_0 d^{n_L+1}$ as the "effective barrier". We shall use the CRR backward scheme with the modified barrier, but with modified values of the rebate at the (modified) barrier. The modification of the boudary value is made as follows: should L be equal to the effective barrier, then the price at the modified barrier would be given by the CRR algorithm at this node:

$$C\left(L=\text{eff}\right)=(1+r)^{-1}\left[\left(\frac{(1+r)-d}{u-d}\right)C_u+R\left(\frac{u-(1+r)}{u-d}\right)\right]$$

Should L match the "modified barrier", then the price at this node would be $C\left(L=\text{mod}\right)=R$.

A natural idea now is to interpolate between these two values:

$$C=\frac{(L-\text{eff})}{(\text{mod}-\text{eff})}C\left(L=\text{mod}\right)+\frac{(\text{mod}-L)}{(\text{mod}-\text{eff})}C\left(L=\text{eff}\right)$$

The modified algorithm is reported to behave like a standard CRR scheme.

2.3 The Ritchken algorithm

This algorithm is implemented in Routine tr_ritchken_downout.c for Down and Out Barrier options. The idea here is to choose the stretch parameter λ such that the barrier is hit exactly. We know that λ should be greater than one, intuitively among many possibilities for λ, the closer λ is to one, the better.

The natural way to choose λ is the following: compute the value n_L above, ie

$$n_L=\left[\frac{\ln\left(\frac{S_0}{L}\right)}{\sigma\sqrt{h}}\right]$$

where $[t]$ denotes the greater integer less or equal than t. Then take $\lambda=\frac{1}{n_L}\frac{\ln\left(\frac{S_0}{L}\right)}{\sigma\sqrt{h}}$.

Here again convergence is reported to be like that for standard options.

2.4 The Rogers & Stapelton method

The principle of the Tree method proposed by Rogers and Stapelton is different from the one of the classical binomial method. Indeed, these authors propose to approximate the logarithm of the stock price $(X_t=X_0+\sigma W_t+(r-\delta-\sigma^2/2)t)_t$ by the random walk $(\xi_n)_n$ where for some fixed $\Delta x>0$,

$$\begin{cases}\xi_n=X_{\tau_n}\\ \tau_0=0,\ \tau_{n+1}=\inf\{t>\tau_n,\ |X_t-X_{\tau_n}|>\Delta x\}\end{cases}$$

The main interest is that the random walk approximates the underlying diffusion uniformly closely. In compensation, if we are interested in an option with maturity T, the number of time-steps $\nu = \sup\{n : \tau_n < T\}$ before T is random.
The price of the European option with payoff function φ, maturity T, up and out barrier b^* (resp. down and out barrier b_*, resp double out barriers $b_* < b^*$) and rebate R is approximated by

$$\sum_{n\geq 0} \mathbb{P}(\nu = n)\mathbb{E}\left(e^{-r\eta T/n}R1_{\{\eta<n\}} + e^{-rT}\varphi(\xi_n)1_{\{\eta\geq n\}}\middle|\nu = n\right)$$

where $\eta = \inf\{n : \xi_n \geq \log(b^*)\}$ (resp $\inf\{n : \xi_n \leq \log(b_*)\}$, resp. $\inf\{n : \xi_n \notin (\log(b_*), \log(b^*))\}$.
One can prove that the path followed by the random walk $(\xi_n)_n$ is independent of ν : more precisely, the transition probabilities can be computed thanks to the scale function $s(x) = -\exp(-2(r - \delta - \sigma^2/2)x/\sigma^2)$ of the underlying diffusion (X_t) by

$$\mathbb{P}(\xi_{n+1} = x+\Delta x|\xi_n = x) = 1-\mathbb{P}(\xi_{n+1} = x-\Delta x|\xi_n = x) = \frac{s(0) - s(-\Delta x)}{s(\Delta x) - s(-\Delta x)}.$$

The conditional expectation $\mathbb{E}\left(e^{-r\eta T/n}R1_{\{\eta<n\}} + e^{-rT}\varphi(\xi_n)1_{\{\eta\geq n\}}\middle|\nu = n\right)$ is computed by backward recursion on a tree with n time-steps. To handle the barrier conditions, it is not necessary to place the barriers b^* and b_* at grid points. Indeed, it is enough to modify the transition probabilities for x^* and x_* the grid points just below b^* and just above b_* :

$$\mathbb{P}(\xi_{n+1} = x^* - \Delta x|\xi_n = x^*) = (s(b^*) - s(x^*))/(s(b^*) - s(x^* - \Delta x)),$$
$$\mathbb{P}(\xi_{n+1} = x_* + \Delta x|\xi_n = x_*) = (s(x_*) - s(b_*))/(s(x_* + \Delta x) - s(b_*)).$$

Now $\mathbb{P}(\nu = n) = \mathbb{P}(\tau_n \leq T) - \mathbb{P}(\tau_{n+1} \leq T)$ is estimated by remarking that the variables $(\tau_{n+1}-\tau_n)_{n\geq 0}$ are Independent and Identically Distributed with common Laplace transform :

$$\mathbb{E}(e^{-\lambda\tau_1}) = \frac{\cosh((r - \delta - \sigma^2/2)\Delta x/\sigma^2)}{\cosh(\gamma\Delta x)}$$
$$\text{where } \gamma = \frac{\sqrt{(r - \delta - \sigma^2/2)^2 + 2\lambda\sigma^2}}{\sigma^2}.$$

More precisely, $\mathbb{P}(\tau_n \leq T)$ is computed thanks to the following refinement of the Central Limit Theorem

$$\mathbb{P}\left(\frac{\tau_n - n\mathbb{E}(\tau_1)}{\sqrt{n\text{Var}(\tau_1)}} \leq x\right) \sim N(x) + \frac{\mathbb{E}\left((\tau_1 - \mathbb{E}(\tau_1))^3\right)(1 - x^2)e^{-x^2/2}}{\sqrt{72\pi n\text{Var}(\tau_1)^3}}$$

where as usual $N(.)$ denotes the cumulative distribution function of the Normal law and the three first moments of the random variable τ_1 are obtained by differenciation of its Laplace transform which is given above.

3 Customization of trees

One of the main attractive features of tree algorithms for option pricing is their easy customization which is due to the visualization of the paths of the underlying in the Markov chain approximation. Thus it is often straightforward to design an algorithm for the pricing of a somewhat involved contingent claim-even if the numerical behavior of the algorithm may be poor, this gives a first idea of the price at stake. This first version of the algorithm may be improved later. Often in practice it is fruitful for this second stage to consider the algorithm like a finite-difference scheme to figure out the weakness of the approximation of the delta or of the price in a tricky region.

Let us look at an example of customization.

3.1 Example: pricing time-dependant american options

The natural way to modify the basic algorithm is to replace the backward formula (6 on page 10) by

$$C_n = \max\left(\varphi\left(nh, S_n\right),\, e^{-\rho h}\sum q_i C_{n+1,u_i}\right)$$

where $\varphi(t,x)$ is the time-dependant payoff of the option.

A particular case is that of the so-called bermuda options where the american right is in force only at a set of prescribed periods: for instance between a fixed date T_1 and maturity. In case the current date is prior to T_1 a natural idea is to apply the backard formula (6 on page 10) between step $N-1$ and n_1 where

$$(n_1 - 1)\, h \leq T_1 \leq n_1 h$$

and then the standard CRR scheme.

This first algorithm is very crude since it gives the same price, N being fixed, for any value of T_1 between $(n_1-1)\,h$ and $n_1 h$. A way to feed the algorithm with the right value is to use a Kamrad-Ritchken trinomial tree with a stretch parameter λ_1 and a number of steps n_1 between times 0 and T_1 and λ_2 and n_2 between T_1 and T. In order to get a recombining tree we get the following pasting condition at time T_1 :

$$\lambda_1\sqrt{\tfrac{T_1}{n_1}} = \lambda_2\sqrt{\tfrac{T-T_1}{n_2}}$$

Do not forget that λ_1 and λ_2 should be grater than one. A possible way to choose the parameters is: fix first $\lambda_1 \geq 1$ (for instance $\lambda_1 = 1.2274$), choose n_1, then take

$$n_2 = \left[\frac{n_1(T-T_1)}{T_1}\right] + 1$$

Then $n_2 T_1 \geq n_1 (T - T_1)$ thus $\lambda_2^2 = \lambda_1^2 \frac{T_1}{n_1(T-T_1)} n_2 \geq \lambda_1^2 \geq 1$.

References

[1] Y.W.KWOK. *Mathematical models of financial derivatives.* Springer Finance, 1998. 5