MTH 9821 Numerical Methods for Finance

Fall 2023

Homework 12

Assigned: November 27; Due: December 4

This homework is to be done as a group. Each team will hand in one homework solution, using the blueprint solution provided by our teaching assistant.

The homework has the following parts:

1. Pricing Down–and–Out barrier options using trinomial trees
2. Pricing Down–and–Out barrier options using Monte Carlo
3. Pricing Down–and–Out barrier options using finite difference methods

Pricing Down–and–Out European Barrier Options using Trinomial Trees

Let $\delta t = \frac{T}{N}$, where T is the given maturity of an option and N is the specified number of time steps in the tree model.

Trinomial trees parametrization:

$$u = e^{\sigma\sqrt{3\delta t}}; \qquad d = e^{-\sigma\sqrt{3\delta t}}; \tag{1}$$

$$p_u = \frac{1}{6} + (r - q - \frac{\sigma^2}{2})\sqrt{\frac{\delta t}{12\sigma^2}}; \tag{2}$$

$$p_m = \frac{2}{3}; \tag{3}$$

$$p_d = \frac{1}{6} - (r - q - \frac{\sigma^2}{2})\sqrt{\frac{\delta t}{12\sigma^2}}. \tag{4}$$

The value of a down–and–out call with strike K, barrier $B < K$, and maturity T, on an underlying asset with spot price S following a lognormal distribution with volatility σ and paying dividends continuously at rate q is

$$V(S,K) = C(S,K) - \left(\frac{B}{S}\right)^{2a} C\left(\frac{B^2}{S}, K\right),$$

where $a = \frac{r-q}{\sigma^2} - \frac{1}{2}$, $C(S,K)$ is the value at time 0 of a call option with strike K on the same underlying asset, and $C\left(\frac{B^2}{S}, K\right)$ is the Black–Scholes value at time 0 of a call with strike K and maturity T on an asset having spot price $\frac{B^2}{S}$ (and the same volatility σ).

Consider a seven months down–and–out call with strike \$40 and barrier \$36 on a lognormally distributed underlying asset with spot price \$42 and volatility 28%, paying 1.5% dividends continuously. Assume that the continuously compounded risk–free interest rate is constant at 4%.

(i) Compute the value of the down–and–out call.

(ii) Use the trinomial trees parametrization (1–4) to price the down–and–out–call with $N = 10 : 1000$ time steps. Plot the approximation error of the method.

(iii) Identify and report the theoretical values of the optimal number of time steps that should be used to price the barrier option using trinomial trees. Confirm using the data from part (ii) that the option values computed using optimal trinomial trees indeed minimize locally the error of the trinomial tree pricer. Plot the errors corresponding to the optimal nodes only.

Pricing Down–and–Out European Barrier Options using Monte Carlo

Generating random numbers. Generate N independent samples from the standard normal distribution as follows:

Generate N_0 independent samples from the uniform distribution on $[0,\ 1]$ by using the Linear Congruential Generator

$$\begin{aligned} x_{i+1} &= a x_i \ + \ c \ (\mathrm{mod}\ k) \\ u_{i+1} &= \frac{x_{i+1}}{k}, \end{aligned}$$

with $x_0 = 1$, $a = 39373$, $c = 0$, and $k = 2^{31} - 1$ to generate $u_1,\ u_2,\ \ldots,\ u_{N_0}$.

Use the Box–Muller Method to generate independent samples from the standard normal distribution using the Marsaglia–Bray algorithm below:

while $X > 1$
    Generate $u_1, u_2 \in U([0,1])$
    $u_1 = 2u_1 - 1$; $u_2 = 2u_2 - 1$
    $X = u_1^2 + u_2^2$
end
$Y \ = \ \sqrt{-2\frac{\ln(X)}{X}}$
$Z_1 = u_1 Y$; $Z_2 = u_2 Y$
return Z_1, Z_2

Use the samples u_i, $i = 1 : N_0$, from the uniform distribution generated previously. Make sure N_0 is large enough in order to generate the required number N independent samples of the standard normal distribution.

Consider a seven months down–and–out call with strike \$40 and barrier \$36 on a lognormally distributed underlying asset with spot price \$42 and volatility 28%, paying 1.5% dividends continuously. Assume that the continuously compounded risk–free interest rate is constant at 4%.

We simulate the risk neutral random path of the asset on n different paths, each one discretized by m time steps corresponding to $\delta t = \frac{T}{m}$. To do this, $N = nm$ independent samples of the standard normal distribution must be generated.

1. Generate n different paths for the evolution of the underlying asset

Use the independent samples $z_1,\ z_2,\ \ldots,\ z_N$ of the standard normal distribution obtained previously using the Box–Muller Method.

Discretize the time to maturity using m time steps corresponding to $\delta t = \frac{T}{m}$. Let $t_j = j\delta t$, for $j = 0 : m$. Use the multiplicative formula for the evolution of the underlying, i.e., on every path, compute

$$S(t_{j+1}) \ = \ S(t_j) \ \exp\left(\left(r - q - \frac{\sigma^2}{2}\right)\delta t \ + \ \sigma\sqrt{\delta t} z_{im+j}\right), \ \forall\ j = 0 : (m-1).$$

2. Compute the exact value C_{dao} using the closed formula for valuing down-and-out calls.

The value of the down–and–out call with barrier B less than the strike K is

$$V(S,K) \ = \ C(S,K) \ - \ \left(\frac{B}{S}\right)^{2a} \ C\left(\frac{B^2}{S}, K\right), \tag{5}$$

where

$$a \ = \ \frac{r-q}{\sigma^2} - \frac{1}{2},$$

$C(S,K)$ is the value at time 0 of a plain vanilla call option with strike K and maturity T on the same underlying asset, and $C\left(\frac{B^2}{S}, K\right)$ is the Black–Scholes value at time 0 of a plain vanilla call with strike K and maturity T on an asset having spot price $\frac{B^2}{S}$ (and the same volatility as the underlying).

3. Use the $N = 10,000 \cdot 2^9$ independent samples from the standard normal distribution obtained previously to generate n different paths for the evolution of the underlying. Let V_i, $i = 1 : n$, the value of the down and out call provided the underlying evolves along the path i.

Compute an approximate value

$$\widehat{V}(n) \;=\; \frac{1}{n}\sum_{i=1}^{n} V_i,$$

and the corresponding approximation error $|C_{dao} - \widehat{V}(n)|$.

3.1. Consider a fixed number of time intervals corresponding to approximately one day, i.e., choose $m = 200$. Use $n = 50 \cdot 2^k$ paths, where $k = 0 : 9$, i.e., $n = \frac{N_k}{m}$, where $N_k = 10,000 \cdot 2^k$, $k = 0 : 9$.

3.2. Consider optimal values for the number of time intervals and paths for each $N_k = 10,000 \cdot 2^k$, $k = 0 : 9$, i.e., let

$$m_k \;=\; \text{ceil}\left(N_k^{1/3} T^{2/3}\right); \quad n_k \;=\; \text{floor}\left(\frac{N_k}{m_k}\right),$$

where T is the maturity of the option.

Report the results in the following format:

N_k	$m = 200$	n	$\widehat{V}(n)$	$\lvert C_{dao} - \widehat{V}(n)\rvert$	m_k	n_k	$\widehat{V}(n_k)$	$\lvert C_{dao} - \widehat{V}(n_k)\rvert$
10,000								
20,000								
40,000								
80,000								
160,000								
320,000								
640,000								
1,280,000								
2,560,000								
5,120,000								

Pricing Down–and–Out European Barrier Options using Finite Differences

The value of the down–and–out call with barrier B less than the strike K is

$$V(S,K) \;=\; C(S,K) \;-\; \left(\frac{B}{S}\right)^{2a} \; C\left(\frac{B^2}{S}, K\right), \tag{6}$$

where

$$a \;=\; \frac{r-q}{\sigma^2} - \frac{1}{2},$$

$C(S,K)$ is the value at time 0 of a plain vanilla call option with strike K and maturity T on the same underlying asset, and $C\left(\frac{B^2}{S}, K\right)$ is the Black–Scholes value at time 0 of a plain vanilla call with strike K and maturity T on an asset having spot price $\frac{B^2}{S}$ (and the same volatility as the underlying).

Value derived from the Closed Formula

Use formula (6) to find the value of a seven months down–and–out call with strike \$40 and barrier \$36 on a lognormally distributed underlying asset with spot price \$42 and volatility 28%, paying 1.5% dividends continuously. Assume that the continuously compounded risk–free interest rate is constant at 4%.

The following change of variables transforms x and τ into S and t, respectively, and maps $V(S,t)$, the value of the down–and–out call option, into $u(x,\tau)$, a solution to the heat equation:

$$V(S,t) \;=\; \exp(-ax - b\tau)u(x,\tau), \tag{7}$$

where

$$x \;=\; \ln\left(\frac{S}{K}\right); \quad \tau \;=\; \frac{(T-t)\sigma^2}{2},$$

and the constants a and b are given by

$$\begin{aligned} a &= \frac{r-q}{\sigma^2} - \frac{1}{2}; \\ b &= \left(\frac{r-q}{\sigma^2} + \frac{1}{2}\right)^2 + \frac{2q}{\sigma^2}. \end{aligned}$$

The function $u(x,\tau)$ satisfies the heat equation on the following bounded domain:

$$u_\tau(x,\tau) \;=\; u_{xx}(x,\tau) \;\forall\; (x,\tau) \in [x_{left}, x_{right}] \times [0, \tau_{final}],$$

where $x_{left} = \ln\left(\frac{B}{K}\right)$. The boundary conditions are

$$\begin{aligned} u(x,0) &= K\exp(ax)\;\max(\exp(x)-1,0), \;\forall\; x_{left} \;\le\; x \;\le\; x_{right}; \\ u(x_{left},\tau) &= 0, \;\forall\; 0 \;\le\; \tau \;\le\; \tau_{final}; \\ u(x_{right},\tau) &= K\exp(ax_{right}+b\tau)\left(\exp\left(x_{right} - \frac{2q\tau}{\sigma^2}\right) - \exp\left(-\frac{2r\tau}{\sigma^2}\right)\right), \;\forall\; 0 \;\le\; \tau \;\le\; \tau_{final}. \end{aligned}$$

We will use a computational domain where $x_{compute} = \ln\left(\frac{S_0}{K}\right)$ is required to be a grid point in the finite difference discretization.

Computational Domain: $x_{compute}$ on the grid

In the (x,τ) space one of the nodal values will be

$$x_{compute} \;=\; \ln\left(\frac{S_0}{K}\right).$$

Also,

$$x_{left} \;=\; \ln\left(\frac{B}{K}\right).$$

The upper bound τ_{final} for τ is

$$\tau_{final} \;=\; \frac{T\sigma^2}{2}.$$

To choose x_{right}, start with M and α_{temp} given (α will be slightly smaller than α_{temp} in the end). Then,

$$\delta\tau = \frac{\tau_{final}}{M}$$

and

$$\delta x_{temp} = \sqrt{\frac{\delta\tau}{\alpha_{temp}}}.$$

Then,

$$N_{left} = \text{floor}\left(\frac{x_{compute} - x_{left}}{\delta x_{temp}}\right),$$

where floor(y) is the largest integer smaller than or equal to y.

Then,

$$\delta x = \frac{x_{compute} - x_{left}}{N_{left}}$$

and $\alpha < \alpha_{temp}$ is defined as

$$\alpha = \frac{\delta\tau}{(\delta x)^2}.$$

Choose the following temporary right end point:

$$\widetilde{x}_{right} = \ln\left(\frac{S_0}{K}\right) + \left(r - q - \frac{\sigma^2}{2}\right)T + 3\sigma\sqrt{T}.$$

Let

$$N_{right} = \text{ceil}\left(\frac{\widetilde{x}_{right} - x_{compute}}{\delta x}\right),$$

where ceil(y) is the smallest integer larger than or equal to y. Then,

$$N = N_{left} + N_{right}$$

and

$$x_{right} = x_{compute} + N_{right}\delta x.$$

Identify the computational domain, i.e., compute and record, for each $M \in \{4, 16, 64, 256\}$ and for $\alpha \in \{0.4, 4\}$, the following parameters describing the computational domain: α, x_{left}, x_{right}, N, δx, $\delta\tau$.

Finite difference solvers

Use Forward Euler with $\alpha_{temp} = 0.4$, Backward Euler with $\alpha_{temp} \in \{0.4, 4\}$, and Crank-Nicolson with $\alpha_{temp} \in \{0.4, 4\}$ to solve the diffusion equation for $u(x, \tau)$. For Backward Euler, use tridiagonal LU without pivoting to solve the linear system at each time step. For Crank-Nicolson, use SOR with relaxation parameter $\omega = 1.2$. Use the previous time step calues as the initial guess. The stopping criterion for SOR is that the norm of the difference between two consecutive approximations is less than $tol = 10^{-6}$.

Note: there is a total of 5 finite difference approximations, corresponding to:

(1) Forward Euler, for $\alpha_{temp} = 0.4$ (one);
(2) Backward Euler, using tridiagonal LU without pivoting, for $\alpha_{temp} \in \{0.4, 4\}$ (two);
(3) Crank-Nicolson using SOR with $\omega = 1.2$, for $\alpha_{temp} \in \{0.4, 4\}$ (two).

Choose $M = 4$ to begin with. Run each finite difference method for the initial value of M chosen above, and then quadruple the number of points on the τ-axis, i.e., choose $M \in \{4, 16, 64, 256\}$.

Pointwise Convergence and the Greeks:

Let U^M be the vector of length $N - 1$ which gives the finite difference solution after M time steps. Recall that $x_{compute} = x_{N_{left}}$. Then, $U^M(N_{left})$ is the finite difference approximation to $u(x_{compute}, \tau_{final})$.

Use the following change of variables to compute the approximate value of the option, $V_{approx}(S_0, 0)$ from $u(x_{compute}, \tau_{final})$:

$$V_{approx}(S_0, 0) = \exp(-ax_{compute} - b\tau_{final})\, u(x_{compute}, \tau_{final}). \tag{8}$$

Let $V_{exact}(S_0, 0)$ be the value of the down–and–out call computed above using formula (6). The pointwise error is

$$error\_pointwise \;=\; |V_{approx}(S_0, 0) - V_{exact}(S_0, 0)|. \tag{9}$$

Finite difference approximations for the Δ, Γ, and Θ of the option can be obtained as follows:

Let

$$\begin{aligned} S_{-1} &= K\exp(x_{N_{left}-1}) \;=\; K\exp(x_{compute} - \delta x); \\ S_0 &= K\exp(x_{N_{left}}) \;=\; K\exp(x_{compute}); \\ S_1 &= K\exp(x_{N_{left}+1}) \;=\; K\exp(x_{compute} + \delta x) \end{aligned}$$

be the values of S corresponding to the nodes

$$\begin{aligned} x_{N_{left}-1} &= x_{compute} - \delta x; \\ x_{N_{left}} &= x_{compute}; \\ x_{N_{left}+1} &= x_{compute} + \delta x, \end{aligned}$$

respectively, and let

$$\begin{aligned} V_{-1} &= \exp(-a x_{N_{left}-1} - b\tau_{final})\; u(x_{N_{left}-1}, \tau_{final}); \\ V_0 &= \exp(-a x_{N_{left}} - b\tau_{final})\; u(x_{N_{left}}, \tau_{final}); \\ V_1 &= \exp(-a x_{N_{left}=1} - b\tau_{final})\; u(x_{N_{left}+1}, \tau_{final}) \end{aligned}$$

be the corresponding finite difference approximate values of the option.

The central difference approximations for the Δ and Γ of the option are

$$\Delta_{central} \;=\; \frac{V_1 - V_{-1}}{S_1 - S_{-1}}; \tag{10}$$

$$\Gamma_{central} \;=\; \frac{(S_0 - S_{-1})V_1 \;-\; (S_1 - S_{-1})V_0 \;+\; (S_1 - S_0)V_{-1}}{(S_0 - S_{-1})(S_1 - S_0)((S_1 - S_{-1})/2))}. \tag{11}$$

To compute an approximation for Θ, note that the next to last time step on the τ–axis, $\tau_{final} - \delta\tau$ corresponds to time

$$\delta t \;=\; \frac{2\delta\tau}{\sigma^2}.$$

Let

$$V_{approx}(S_0, \delta t) \;=\; \exp(-a x_{N_{left}} - b(\tau_{final} - \delta\tau))\; u(x_{N_{left}}, \tau_{final} - \delta\tau).$$

The forward finite difference approximation of $\Theta = \frac{\partial V}{\partial t}$ is

$$\Theta_{forward} \;=\; \frac{V_{approx}(S_0, 0) - V_{approx}(S_0, \delta t)}{\delta t}. \tag{12}$$

Finite Difference Solution

For each finite difference method, compute and record:

(1) $U^M(N_{left})$ as "u value"
(2) $V_{approx}(S_0, 0)$ given by (8) as "Option Value";
(3) *error_pointwise* given by (9) as "Pointwise Error";
(4) $\Delta_{central}$ given by (10);
(5) $\Gamma_{central}$ given by (11);
(6) $\Theta_{forward}$ given by (12).

To understand the numbers you provide, please include the following: for Forward Euler with $\alpha_{temp} = 0.4$ and for Backward Euler with $\alpha_{temp} = 0.4$, let $M = 4$. Run your codes and record the values of the finite difference approximations at each nodes, including at the boundary nodes. For $M = 4$ and $\alpha_{temp} = 0.4$ the corresponding value of N is $N = 5$.

Thus, for Forward Euler and Backward Euler for the domain including $x_{compute}$ you will have to fill out two tables with five rows (corresponding to time steps from 0 - boundary conditions, to 4) and 6 columns (including the boundary conditions at x_{left} and x_{right}).