

Computational Finance - 11th Assignment

Deadline: June 29

Exercise 33 (Perpetual Put Option)

(4+2+4 points)

For $T \rightarrow \infty$ it is sufficient to analyze the ODE

$$\frac{\sigma^2}{2} S^2 \frac{d^2 V}{dS^2} + (r - \delta) S \frac{dV}{dS} - rV = 0.$$

Consider an American put with high contact to the payoff $V = (K - S)^+$ at $S = S_f$. Show:

a) Upon substituting the boundary condition for $S \rightarrow \infty$ one obtains

$$V(S) = c \left(\frac{S}{K} \right)^{\lambda_2},$$

where $\lambda_2 = \frac{1}{2} \left(1 - q_\delta - \sqrt{(q_\delta - 1)^2 + 4q} \right)$, $q = \frac{2r}{\sigma^2}$, $q_\delta = \frac{2(r-\delta)}{\sigma^2}$
and c is a positive constant.

Hint: Apply the transformation $S = Ke^x$. (The other root λ_1 drops out.)

b) V is convex.

For $S < S_f$ the option is exercised; then its intrinsic value is $K - S$. For $S > S_f$ the option is not exercised and has a value $V(S) > K - S$. The holder of the option decides when to exercise. This means, the holder makes a decision on the high contact S_f such that the value of the option becomes maximal.

c) Show: $V'(S_f) = -1$, if S_f maximizes the value of the option.

Hint: Determine the constant c such that $V(S)$ is continuous in the contact point.

Exercise 34 (UL decomposition)

(3 points)

Assume a system of linear equations $Ax = b$ with tridiagonal matrix A . Formulate the *UL* decomposition as an algorithm.

Exercise 35 (Brennan-Schwartz Algorithm)

(3+2 points)

Let A be a tridiagonal matrix and b and g vectors. The system of equations $Aw = b$ is to be solved such that the side condition $w \geq g$ is obeyed componentwise. Assume for a put $w_i = g_i$ for $1 \leq i \leq i_f$ and $w_i > g_i$ for $i_f < i \leq n$, where i_f is unknown.

- a) Formulate an algorithm that solves $Aw = b$ in the *backward/forward* approach. In the final forward loop, for each i the calculated candidate w_i is tested for $w_i \geq g_i$: In case $w_i < g_i$ the calculated value w_i is corrected to $w_i = g_i$.
- b) Apply the algorithm to the case of a put with relevant A, b, g .

Exercise 36 (Computation of American Options)

(20P points)

Implement an algorithm for the calculation of American-style options, following the prototype algorithm below. Use Exercises 34 and 35. For this assignment, it is sufficient to implement the case of a put.

Test your program with the following example: $K = 10$, $r = 0.25$, $\sigma = 0.6$, $T = 1$, $\delta = 0.2$. Calculate approximations to $V_P(10, 0)$.

Algorithm (prototype algorithm)

Set up the function $g(x, \tau)$ listed in the summary below.

Choose θ ($\theta = 1/2$ for Crank-Nicolson).

Fix the discretization by choosing $x_{\min}$, $x_{\max}$, m , $\nu_{\max}$ (for example: $x_{\min} = -5$, $x_{\max} = 5$, $\nu_{\max} = m = 100$).

Calculate $\Delta x := (x_{\max} - x_{\min})/m$,

$$\Delta \tau := \frac{1}{2} \sigma^2 T / \nu_{\max},$$

$$x_i := x_{\min} + i \Delta x \text{ for } i = 0, \dots, m,$$

$$\lambda := \Delta \tau / \Delta x^2 \text{ and } \alpha := \lambda \theta.$$

Initialize the iteration vector w with

$$g^{(0)} = (g(x_1, 0), \dots, g(x_{m-1}, 0))^{tr}.$$

Arrange for the matrix A .

τ -loop: for $\nu = 0, 1, \dots, \nu_{\max} - 1$:

$$\tau_\nu := \nu \Delta \tau$$

initialize the vector b with

$$b_i := w_i + \lambda(1 - \theta)(w_{i+1} - 2w_i + w_{i-1}) \text{ for } 2 \leq i \leq m - 2,$$

$$b_1 := w_1 + \lambda(1 - \theta)(w_2 - 2w_1 + g_{0\nu}) + \alpha g_{0,\nu+1},$$

$$b_{m-1} := w_{m-1} + \lambda(1 - \theta)(g_{m\nu} - 2w_{m-1} + w_{m-2}) + \alpha g_{m,\nu+1}$$

subroutine for the LCP solution w , directly as in Exercises 34 and 35

$$w^{(\nu+1)} = w$$

Summary of American options, for a put ($r > 0$) or a call ($\delta > 0$), after transformation into (x, τ, y) -variables:

$$q := \frac{2r}{\sigma^2}; \quad q_\delta := \frac{2(r - \delta)}{\sigma^2}$$

$$\text{put: } g(x, \tau) := \exp\left\{\frac{1}{4}((q_\delta - 1)^2 + 4q)\tau\right\} \max\{e^{\frac{1}{2}(q_\delta - 1)x} - e^{\frac{1}{2}(q_\delta + 1)x}, 0\}$$

$$\text{call: } g(x, \tau) := \exp\left\{\frac{1}{4}((q_\delta - 1)^2 + 4q)\tau\right\} \max\{e^{\frac{1}{2}(q_\delta + 1)x} - e^{\frac{1}{2}(q_\delta - 1)x}, 0\}$$

$$\left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2}\right)(y - g) = 0$$

$$\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0, \quad y - g \geq 0$$

$$y(x, 0) = g(x, 0), \quad 0 \leq \tau \leq \frac{1}{2}\sigma^2 T$$

$$\lim_{x \rightarrow \pm\infty} y(x, \tau) = \lim_{x \rightarrow \pm\infty} g(x, \tau)$$

Information:

- The deadline for the programming exercise is July 6. Please turn in a printed version of your code and send it to *numerik_programm@gmx.de*.
- The second session of the Cologne Computational Finance Laboratory takes place after exercises on July 4 and on July 6.