ANALELE ŞTIINŢIFICE ALE UNIVERSITĂŢII "AL.I. CUZA" DIN IAŞI (S.N.)
MATEMATICĂ, Tomul LVI, 2010, f.1

NUMERICAL APPROXIMATION OF BLACK-SCHOLES EQUATION

BY

GINA DURA and ANA-MARIA MOŞNEAGU

Abstract. This study deals with well-known Black-Scholes model in a complete financial market. We obtain numerical methods for european and exotic options, for one asset and for two assets models.

Mathematics Subject Classification 2000: 65F10, 65F05, 65N06, 91B28.

Key words: Black-Scholes equation, European call/put option, payoff, finite-differences method.

Introduction. Options are used on markets and exchanges. The Black-Scholes model is a convenient way to calculate the price of an option. In this study numerical methods will be processed to solve that equation. The numerical methods are based on finite differences. We test fully implicit, semi-implicit and explicit methods. For the discretized problem, a linear algebraic system, we test direct and iterative methods. This way we intend to create a general numerical scheme for different types of options.

Section 1 deals with the Black-Scholes model. In this part we show some options used, European call and put option, and the mathematical properties of these options.

Section 2 shows our numerical setup for the equation presented in Section 1. This section deals with the space discretization of the parabolic differential equation with one underlying asset, using finite difference methods. We obtain an explicit and two implicit methods (fully implicit and semi-implicit method) to solve the discretized system. Numerical experiments are related for each method.

In Section 3 we extend the Black-Scholes model for two underlying assets. We still use the finite difference method. The explicit method is used and numerical tests are discussed.

1. The Black-Scholes model. The concept of arbitrage allows to establish precise relationships between prices and hence to determine them. We discuss option strategies in general and use arbitrage, together with the model for asset price movements, to derive the celebrated Black-Scholes differential equation for the simplest options, the so-called European vanilla options. We also discuss the boundary conditions to be satisfied by European call and put option.

One of the fundamental concepts underlying the theory of financial derivative pricing and hedging is that of arbitrage. In financial terms, there are never any opportunities to make an instantaneous risk-free profit. But almost all finance theory assume the existence of risk-free investments that give a guaranteed return with no chance of default. A good approximation to such an investment is a government bond or a deposit in bank. The greatest risk-free return that one can make on a portfolio of assets is the same as the return for the equivalent amount of cash placed in bank. For general theory of financial derivatives, e.g. [4], [8], [12]. For related results on elliptic equation see also [1], [2].

1.1. Option values, payoffs and strategies. We will introduce some simple notation :
- V = the option value; when the distinction is important we use $C(S,t)$ to denote a call option and $P(S,t)$ to denote a put option. This value is a function of current value of the underlying asset S and time, i.e. $V = V(S,t)$. The value of the option also depends on the following parameters:
- σ = the volatility of the underlying asset;
- E = the exercise price ;
- T = the expiry time;
- r = the interest rate .

Let us consider what happens just at the moment of expiry of the call option, that is, at time $t = T$. A simple arbitrage argument is enough:

- If $S > E$ at expiry, it makes financial sense to exercise the call option, handing over an amount E, to obtain an asset worth S. The profit from such a transaction is then $S - E$.

- If $S < E$ at expiry, we should not exercises the option because we would make a loss of $S - E$. In this case, the option expires worthless. Thus, the value of a call option at expiry can be written as

$$C(S,T) = max(S - E, 0) \tag{1}$$

We consider now European options, i.e. options which may only be exercised at expiry (at time $T = t$).

We assume that we can buy and sell any number (not necessarily an integer) of the underlying asset, and that we may sell assets that we do not own.

Suppose that we have an option whose value $V(S,t)$ depends only on S and t. It is not necessary at this stage to specify whether V is a call or a put. The underlying equation is

$$dV = \sigma S \frac{\partial V}{\partial S} dX + \left(\mu S \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t} \right) dt \tag{2}$$

where the random variable X, or the change dX, is a Wiener process and μ is a known constant (the drift).

It gives the random walk followed by V. Note that we require V to have at least one t derivative and two S derivatives (e.g. [12], Chapter 3.5).

The corresponding Black-Scholes partial differential equation is

$$\frac{\partial V}{\partial t}(S,t) + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}(S,t) + rS\frac{\partial V}{\partial S}(S,t) - rV(S,t) = 0 \tag{3}$$

(see [3]).

The delta, given by $\Delta = \frac{\partial V}{\partial S}$, is the rate of change of the value of our option or portio of options with respect to S. It is of fundamental importance in both theory and practice. It is a measure of the correlation between the movements of the option or other derivative products and those of the underlying asset.

The linear differential operator $\mathcal{L}_{BS}$ given by

$$\mathcal{L}_{BS} = \frac{\partial}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2}{\partial S^2} + rS\frac{\partial}{\partial S} - r \tag{4}$$

has a financial interpretation as a measure of the difference between the return on a hedged option portfolio (the first two terms) and the return on a bank deposit (the last two terms). Althought this difference must be identically zero for an European option, in order to avoid arbitrage.

Our partial differential equation is of parabolic type and we can make general statements about the sort of boundary conditions that lead to a unique solution. Typically, we must pose two conditions in S.

For example we could specify that

$$V(S,t) = V_a(t) \text{ for } S = a \tag{5}$$

and

$$V(S,t) = V_b(t) \text{ for } S = b, \tag{6}$$

where V_a and V_b are given functions of t.

If the equation is of backward type we must also impose a final condition such as

$$V(S,t) = V_T(S) \text{ for } t = T, \tag{7}$$

where V_T is a known function.

Consider the Black-Scholes equation (3) and apply the change of variables

$$k(t) = T - t \tag{8}$$

to transform the parabolic backward equation into a forward one.

From the following relation

$$\frac{\partial V(S,t)}{\partial t} = \frac{\partial V(S,k(t))}{\partial k(t)} k'(t) = -\frac{\partial V(S,k)}{\partial k} \tag{9}$$

we see that we can replace again k with t.

By doing this, equation (3) becomes

$$\frac{\partial V}{\partial t}(S,t) - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}(S,t) - rS\frac{\partial V}{\partial S}(S,t) + rV(S,t) = 0. \tag{10}$$

For an European call option, let us denote the function value V by C:

$$\frac{\partial C}{\partial t}(S,t) - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2}(S,t) - rS\frac{\partial C}{\partial S}(S,t) + rC(S,t) = 0. \tag{11}$$

The final condition

$$C(S,T) = \max(S-E,0) \tag{12}$$

becomes initial condition

$$C(S,0) = \max(S-E,0) \tag{13}$$

and the boundary conditions are

$$\begin{cases} C(0,t) = 0 \\ C(S,t)/S \to 1, \quad \text{for } S \to \infty. \end{cases} \tag{14}$$

For an European put option, V is usualy denoted by P:

$$\frac{\partial P}{\partial t}(S,t) - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 P}{\partial S^2}(S,t) - rS\frac{\partial P}{\partial S}(S,t) + rP(S,t) = 0. \tag{15}$$

The final condition

$$P(S,T) = \max(E-S,0) \tag{16}$$

is changed to initial condition

$$P(S,0) = \max(E-S,0) \tag{17}$$

and the boundary conditions are

$$\begin{cases} P(0,t) = Ee^{-rt} \\ P(S,t) \to 0, \qquad \text{for } S \to \infty. \end{cases} \tag{18}$$

2. Finite-difference methods for European call and put options. Finite-difference methods are used to solve the Black-Scholes partial differential equation.

2.1. Space discretization. The interval $[0,T]$ is divided into M equally sized subintervals of length Δt. The price of the underlying asset will take values in the unbounded interval $[0,\infty)$. However, an artificial limit $S_{\max}$ is introduced. The size of $S_{\max}$ requires experimentation, but a simple rule is to let $S_{\max}$ be around three to four times the exercise price

E. The interval $[0, S_{\max}]$ is divided into N equally sized subintervals of length ΔS. Hence the space $[0, S_{\max}] \times [0, T]$ is approximated by a grid

$$(n\Delta S, m\Delta t) \in [0, N\Delta S] \times [0, M\Delta t]\,, \tag{19}$$

where $n = 0, \ldots, N$ and $m = 0, \ldots, M$. In which follows, we denote by v_n^m the numerical approximation of $V\,(n\Delta S, m\Delta t)$.

From (14) we get

$$C(S_{\max}, t) = S_{\max}, \text{for } t \in [0, T] \tag{20}$$

and from (18)

$$P(S_{\max}, t) = 0, \text{for } t \in [0, T]\,. \tag{21}$$

2.2. The explicit method. We use a forward difference approximation for the time derivative, a central difference approximation for the first order S derivative and a symmetric central difference approximation for the second order S derivative

$$\frac{\partial V}{\partial t}(n\Delta S, m\Delta t) = \frac{v_n^{m+1} - v_n^m}{\Delta t} + O(\Delta t), \tag{22}$$

$$\frac{\partial V}{\partial S}(n\Delta S, m\Delta t) = \frac{v_{n+1}^m - v_{n-1}^m}{2\Delta S} + O((\Delta S)^2), \tag{23}$$

$$\frac{\partial^2 V}{\partial S^2}(n\Delta S, m\Delta t) = \frac{v_{n+1}^m - 2v_n^m + v_{n-1}^m}{(\Delta S)^2} + O((\Delta S)^2). \tag{24}$$

We get the following discretized form of equation (10):

$$\begin{aligned} &\frac{v_n^{m+1} - v_n^m}{\Delta t} - \frac{1}{2}\sigma^2 n^2 (\Delta S)^2 \frac{v_{n+1}^m - 2v_n^m + v_{n-1}^m}{(\Delta S)^2} \\ &- rn\Delta S \frac{v_{n+1}^m - v_{n-1}^m}{2\Delta S} + rv_n^m = 0,\ n = \overline{1, N-1}, m = \overline{0, M-1}. \end{aligned} \tag{25}$$

In this case, the error is of order $O(\Delta t + (\Delta S)^2)$.

From equation (25) we get

$$\begin{aligned} v_n^{m+1} = &\frac{1}{2}(\sigma^2 n^2 \Delta t - rn\Delta t)v_{n-1}^m + (1 - \sigma^2 n^2 \Delta t - r\Delta t)v_n^m \\ &+ \frac{1}{2}(\sigma^2 n^2 \Delta t + rn\Delta t)v_{n+1}^m,\ n = \overline{1, N-1}, m = \overline{0, M-1}. \end{aligned} \tag{26}$$

We observe that the term v_n^{m+1} from time step $m+1$ is evaluated only using the terms v_{n-1}^m, v_n^m, v_{n+1}^m from one time step back m.

Denoting

$$\begin{aligned} \alpha &= \sigma^2 \Delta t, \\ \beta &= r\Delta t \end{aligned} \tag{27}$$

we obtain:

$$v_n^{m+1} = \frac{1}{2}(\alpha n^2 - \beta n)v_{n-1}^m + (1 - \alpha n^2 - \beta)v_n^m + \frac{1}{2}(\alpha n^2 + \beta n)v_{n+1}^m, \tag{28}$$

for $n = \overline{1, N-1}$ and $m = \overline{0, M-1}$.

We introduce the matrix

$$A = \begin{pmatrix} d_1 & u_2 & 0 & \cdots & 0 \\ l_1 & d_2 & u_3 & & \vdots \\ 0 & \ddots & & & \\ & & & \ddots & 0 \\ \vdots & & \ddots & \ddots & u_{N-1} \\ 0 & & \cdots & l_{N-2} & d_{N-1} \end{pmatrix}, \tag{29}$$

where $A \in \mathbb{R}^{(N-1)\times(N-1)}$,

$$v^{m+1} = \begin{pmatrix} v_1^{m+1} \\ v_2^{m+1} \\ \vdots \\ v_{N-1}^{m+1} \end{pmatrix}, \ v^m = \begin{pmatrix} v_1^m \\ v_2^m \\ \vdots \\ v_{N-1}^m \end{pmatrix}, \ z^m = \begin{pmatrix} l_0 v_0^m \\ 0 \\ \vdots \\ 0 \\ u_N v_N^m \end{pmatrix}, \tag{30}$$

where v^{m+1}, v^m, $z^m \in \mathbb{R}^{N-1}$ and

$$\begin{aligned} d_n &= 1 - \alpha n^2 - \beta, && n = 1, \ldots, N-1, \\ u_n &= \tfrac{1}{2}(\alpha(n-1)^2 + \beta(n-1)), && n = 2, \ldots, N, \\ l_n &= \tfrac{1}{2}(\alpha(n+1)^2 - \beta(n+1)), && n = 0, \ldots, N-2. \end{aligned} \tag{31}$$

At each time step $m+1$ the approximate solution can be obtained from the matrix equation:

$$v^{m+1} = Av^m + z^m, \ \ m = \overline{0, M-1}. \tag{32}$$

The values v_n^0, v_0^m, v_N^m with $n = 0, \dots, N$ and $m = 0, \dots, M$ are known from initial and boundary conditions.

The following condition of stability

$$0 < \Delta t < \frac{1}{\sigma^2(N-1) + \frac{1}{2}r} \tag{33}$$

can be proved (see [6]).

2.2.1. Implementation. The explicit method is easy to implement with any programming language capable of storing arrays of data. Using formula (32), we can calculate all of the values for the next time step one by one.

For an European call option with $T = 0.25, E = 10.0, r = 0.1, \sigma = 0.4$ we get the following numerical results:

	$N = 200, M = 2000$	$N = 1000, M = 41000$	
S	C_{approx}	C_{approx}	C_{exact}
4.0	$1.599567e-006$	$1.086949e-006$	$1.067322e-006$
8.0	0.148976	0.149320	0.149335
10.0	0.915363	0.916255	0.916291
16.0	6.252299	6.252287	6.252287
20.0	10.247000	10.246997	10.247014

The exact solution of the European call option problem, in particular case when the interest rate and volatility are constant, can be computed. C_{exact} is calculated using the relations:

$$C(S,t) = SN(d_1) - Ee^{-r(T-t)}N(d_2), \tag{34}$$

where $N(\cdot)$ is the cumulative distribution function for a standardised normal random variable, given by

$$N(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{1}{2}y^2} dy \tag{35}$$

and

$$d_{1,2} = \frac{\log(S/E) + (r \pm \frac{1}{2}\sigma^2)(T-t)}{\sigma\sqrt{T-t}}. \tag{36}$$

We observe from Figure 1 that the graphic of the approximate solution coincides with the graphic of the exact solution.

The difference of the upper boundary condition $C(S_{\max}, t)$ and the values $C(0,t)$ causes the rising error when $C(S,t) \to S_{\max}$. We observe this aspect in Figure 2.

2.3. The fully implicit method. We now use a backward difference approximation for the time derivative. We have

$$\frac{\partial V}{\partial t}(n\Delta S, (m+1)\Delta t) = \frac{v_n^{m+1} - v_n^m}{\Delta t} + O(\Delta t), \tag{37}$$

$$\frac{\partial V}{\partial S}(n\Delta S, (m+1)\Delta t) = \frac{v_{n+1}^{m+1} - v_{n-1}^{m+1}}{2\Delta S} + O((\Delta S)^2), \tag{38}$$

$$\frac{\partial^2 V}{\partial S^2}(n\Delta S, (m+1)\Delta t) = \frac{v_{n+1}^{m+1} - 2v_n^{m+1} + v_{n-1}^{m+1}}{(\Delta S)^2} + O((\Delta S)^2). \tag{39}$$

The discretization of equation (10) becomes

$$\begin{aligned} &\frac{v_n^{m+1} - v_n^m}{\Delta t} - \frac{1}{2}\sigma^2 n^2 (\Delta S)^2 \frac{v_{n+1}^{m+1} - 2v_n^{m+1} + v_{n-1}^{m+1}}{(\Delta S)^2} \\ &-rn\Delta S \frac{v_{n+1}^{m+1} - v_{n-1}^{m+1}}{2\Delta S} + rv_n^{m+1} = 0,\ n = \overline{1, N-1}, m = \overline{0, M-1}. \end{aligned} \tag{40}$$

The order of the method is $O(\Delta t + (\Delta S)^2)$.

The term v_n^m from time step m is evaluated using the terms v_{n-1}^{m+1}, v_n^{m+1}, v_{n+1}^{m+1} from the time step $m+1$:

$$\begin{aligned} v_n^m = &\frac{1}{2}(rn\Delta t - \sigma^2 n^2 \Delta t) v_{n-1}^{m+1} + (1 + r\Delta t + \sigma^2 n^2 \Delta t) v_n^{m+1} \\ &- \frac{1}{2}(rn\Delta t + \sigma^2 n^2 \Delta t) v_{n+1}^{m+1}, n = \overline{1, N-1}, m = \overline{0, M-1}. \end{aligned} \tag{41}$$

Using the notations (27) the equation (41) becomes

$$v_n^m = \frac{1}{2}(\beta n - \alpha n^2) v_{n-1}^{m+1} + (1 + \beta + \alpha n^2) v_n^{m+1} - \frac{1}{2}(\beta n + \alpha n^2) v_{n+1}^{m+1}, \tag{42}$$

for $n = \overline{1, N-1}, m = \overline{0, M-1}$.

We denote

$$A = \begin{pmatrix} d_1 & u_2 & 0 & \cdots & 0 \\ l_1 & d_2 & u_3 & & \vdots \\ 0 & \ddots & & & \\ & & & \ddots & 0 \\ \vdots & & \ddots & \ddots & u_{N-1} \\ 0 & & \cdots & l_{N-2} & d_{N-1} \end{pmatrix}, \tag{43}$$

where $A \in \mathbb{R}^{(N-1)\times(N-1)}$,

$$v^{m+1} = \begin{pmatrix} v_1^{m+1} \\ v_2^{m+1} \\ \vdots \\ v_{N-1}^{m+1} \end{pmatrix}, \; b^m = \begin{pmatrix} v_1^m \\ v_2^m \\ \vdots \\ v_{N-2}^m \\ v_{N-1}^m \end{pmatrix} - \begin{pmatrix} l_0 v_0^{m+1} \\ 0 \\ \vdots \\ 0 \\ u_N v_N^{m+1} \end{pmatrix}, \tag{44}$$

where v^{m+1}, v^m, $b^m \in \mathbb{R}^{N-1}$ and

$$\begin{aligned} d_n &= 1 + \beta + \alpha n^2, && n = 1, \ldots, N-1, \\ u_n &= -\tfrac{1}{2}(\beta(n-1) + \alpha(n-1)^2), && n = 2, \ldots, N, \\ l_n &= \tfrac{1}{2}(\beta(n+1) - \alpha(n+1)^2), && n = 0, \ldots, N-2. \end{aligned} \tag{45}$$

At each time step $m+1$ the approximate solution can be obtained by solving the system of linear equations:

$$Av^{m+1} = b^m, \;\; m = \overline{0, M-1}. \tag{46}$$

Hence in each time step, the implicit method provides a set of equations which must be solved simultaneously for all $n = 1, 2, \ldots, N-1$.

The values v_n^0, v_0^m, v_N^m with $n = 0, \ldots, N$ and $m = 0, \ldots, M$ are known from initial and boundary conditions.

2.4. The semi-implicit method. We use a backward difference approximation for the time derivative, a forward difference approximation for the first order S derivative and a symmetric central difference approximation for the second order S derivative

$$\frac{\partial V}{\partial t}(n\Delta S, (m+1)\Delta t) = \frac{v_n^{m+1} - v_n^m}{\Delta t} + O(\Delta t), \tag{47}$$

$$\frac{\partial V}{\partial S}(n\Delta S,(m+1)\Delta t)=\frac{v_{n+1}^{m+1}-v_n^{m+1}}{\Delta S}+O(\Delta S), \tag{48}$$

$$\frac{\partial^2 V}{\partial S^2}(n\Delta S,(m+1)\Delta t)=\frac{v_{n+1}^{m+1}-2v_n^{m+1}+v_{n-1}^{m+1}}{(\Delta S)^2}+O((\Delta S)^2). \tag{49}$$

The discretized form of equation (10) is

$$\begin{aligned}&\frac{v_n^{m+1}-v_n^m}{\Delta t}-\frac{1}{2}\sigma^2 n^2(\Delta S)^2\frac{v_{n+1}^{m+1}-2v_n^{m+1}+v_{n-1}^{m+1}}{(\Delta S)^2}\\ &-rn\Delta S\frac{v_{n+1}^{m+1}-v_n^{m+1}}{\Delta S}+rv_n^{m+1}=0, n=\overline{1,N-1}, m=\overline{0,M-1}.\end{aligned} \tag{50}$$

The order of the method is $O(\Delta t+\Delta S)$.

The term v_n^m from time step m is evaluated using the terms v_{n-1}^{m+1}, v_n^{m+1}, v_{n+1}^{m+1} from the time step $m+1$:

$$\begin{aligned}v_n^m=&-\frac{1}{2}\sigma^2 n^2\Delta t v_{n-1}^{m+1}+(1+r\Delta t+rn\Delta t+\sigma^2 n^2\Delta t)v_n^{m+1}\\ &-(rn\Delta t+\frac{1}{2}\sigma^2 n^2\Delta t)v_{n+1}^{m+1}, n=\overline{1,N-1}, m=\overline{0,M-1}.\end{aligned} \tag{51}$$

Using the notations (27) the equation becomes

$$v_n^m=-\frac{1}{2}\alpha n^2 v_{n-1}^{m+1}+(1+\beta+\beta n+\alpha n^2)v_n^{m+1}-(\beta n+\frac{1}{2}\alpha n^2)v_{n+1}^{m+1}, \tag{52}$$

for $n=\overline{1,N-1}$ and $m=\overline{0,M-1}$.

Using notations (43) and (44) with

$$\begin{array}{ll} d_n=1+\beta+\beta n+\alpha n^2, & n=1,\dots,N-1,\\ u_n=-(\beta(n-1)+\frac{1}{2}\alpha(n-1)^2), & n=2,\dots,N,\\ l_n=-\frac{1}{2}\alpha(n+1)^2, & n=0,\dots,N-2\end{array} \tag{53}$$

the system is:

$$Av^{m+1}=b^m,\ \ m=\overline{0,M-1}. \tag{54}$$

The values v_n^0, v_0^m, v_N^m with $n=0,\dots,N$ and $m=0,\dots,M$ are known from initial and boundary conditions.

The fully implicit and semi-implicit methods are unconditionally stable.

2.4.1. Implementation. The systems (46) or (54) has $N-1$ equations and as many unknowns, so there is a unique solution for v^{m+1} as long as the matrix A is non-singular.

To solve the systems (46) or (54) we can use direct methods or iterative methods.

Direct methods. An obvious way to solve both systems is to find the inverse matrix A^{-1} (if the matrix is non-singular), but there are faster methods. The matrix A is tridiagonal. So, it is an easy way to use the LU decomposition algorithm. It is possible, of course, to use the Gaussian elimination.

We use the fully implicit method for an European put option with $T = 0.25, E = 10.0, r = 0.1, \sigma = 0.4, M = 2000, N = 200$.

The matrix A for this example is strictly diagonally dominant, i.e.

$$|A_{i,i}| > \sum_{i \neq j} |A_{i,j}| . \tag{55}$$

Then A is non-singular.

The system (46) is solved using the LU decomposition algorithm wich was adapted to the special form of the matrix A.

S	P_{approx}	P_{exact}
4.0	5.753102	5.753100
8.0	1.902102	1.902434
10.0	0.668360	0.669390
16.0	0.005419	0.005386
20.0	$1.170806e-004$	$1.129336e-004$

P_{exact} is calculated, in particular case when the interest rate and volatility are constant, using the relation

$$P(S,t) = Ee^{-r(T-t)}N(-d_2) - SN(-d_1) \tag{56}$$

with $N(x)$, d_1 and d_2 from relations (35) and (36).

We obtain the solution that we can see in Figure 3.

In Figure 4 we observe that the absolute errors are the largest near the strike price $E = 10.0$, where $\partial P/\partial S$ is discontinuous. In Figure 5 we observe the relative error.

Iterative methods. The Jacobi method, Gauss-Seidel method or Succesive OverRelaxation(SOR) are such methods. If the discretization grid contains many points an iterative algorithm becomes slow. But pricing derivatives should be computed very quickly. It is therefore necessary to sacrifice some of the accuracy, both in the step sizes and in the error tolerance level of the iteration, in order to produce a result within a reasonable time.

To solve the general system of linear equations

$$Ax = b, \tag{57}$$

we consider the following decomposition of the matrix $A \in \mathbb{R}^{N\times N}$:

$$A = L + U + D, \tag{58}$$

where L is the lower triangular part of A with zeros on the diagonal, U is the upper triangular part of A with zeros on the diagonal and D is diagonal part of A.

The Jacobi method. In the Jacobi method we write

$$D \cdot x^{(k+1)} = -(L+U)\cdot x^{(k)} + b. \tag{59}$$

Let us consider the system (54). The system matrix A is, in this case, strictly diagonally dominant, for any numerical datas . Hence, the matrix is non-singular and the Jacobi method converges. The matrix A admits the decomposition above.

The approximate solution for the step time $m+1$ is

$$v^{m+1,k+1} = D^{-1}\cdot(-(L+U)\cdot v^{m+1,k} + b^m), \tag{60}$$

or,

$$\begin{aligned} v_n^{m+1,k+1} &= \frac{1}{1+\beta+\beta n+\alpha n^2}(v_n^{m,k} + \frac{1}{2}\alpha n^2 v_{n-1}^{m+1,k} \\ &+ (\beta n + \frac{1}{2}\alpha n^2) v_{n+1}^{m+1,k}), \end{aligned} \tag{61}$$

for $n = 1, \dots, N-1$.

This procedure is then iterated until convergence. The ”stopping criterion” considered is

$$\|v_n^{m+1,k+1} - v_n^{m+1,k}\| < \epsilon, \tag{62}$$

where ϵ is a prescribed precision.

The Gauss-Seidel method. In this method we make use of updated values of x as soon as they become available.

The Gauss-Seidel method coresponds to the matrix decomposition

$$(L+D)\cdot x^{(k+1)} = -U\cdot x^{(k)} + b. \tag{63}$$

For system (54), the $(k+1)$ iteration is:

$$v^{m+1,k+1} = (L+D)^{-1}\cdot(-U\cdot v^{m+1,k} + b^m). \tag{64}$$

Let us write the relation above by components:

$$\begin{aligned} v_n^{m+1,k+1} &= \frac{1}{1+\beta+\beta n+\alpha n^2}(v_n^{m,k} + \frac{1}{2}\alpha n^2 v_{n-1}^{m+1,k+1} \\ &+ (\beta n + \frac{1}{2}\alpha n^2)v_{n+1}^{m+1,k}), \end{aligned} \tag{65}$$

for $n = 1, \ldots, N-1$.

The Gauss-Seidel method converges for the same reasons as the Jacobi method.

Both methods, Jacobi and Gauss-Seidel, are not very used in practice because they converge too slowly.

The SOR method. This method is a practical useful algorithm. Add and substract $x^{(k)}$on the right hand side of equation (63) and write the Gauss-Seidel method as

$$x^{(k+1)} = x^{(k)} + D^{-1}\cdot(-L\cdot x^{(k+1)} - D\cdot x^{(k)} - U\cdot x^{(k)} + b). \tag{66}$$

Now we make the correction, defining

$$x^{(k+1)} = x^{(k)} + \omega D^{-1}\cdot(-L\cdot x^{(k+1)} - D\cdot x^{(k)} - U\cdot x^{(k)} + b). \tag{67}$$

Here ω is the overrelaxation parameter, and the method is called Succesive OverRelaxation (SOR)

$$(D+\omega L)x^{(k+1)} = ((1-\omega)D - \omega U)\cdot x^{(k)} + \omega b. \tag{68}$$

We recall the following results ([9], Chapter 17):

• The method is convergent only for $0 < \omega < 2$. If $0 < \omega < 1$, we speak of underrelaxation.

• Only overrelaxation, $1 < \omega < 2$, can give faster convergence than the Gauss-Seidel method (under certain mathematical restrictions)

• If ρ_J is the spectral radius of the Jacobi iteration, then the optimal choice for ω is given by

$$\omega = \frac{2}{1 + \sqrt{1 - \rho_J^2}}. \tag{69}$$

A good comparation between these three methods is related in [5].

For the system (54), we get $v^{m+1,k+1}$ using equation (68):

$$v^{m+1,k+1} = (D + \omega L)^{-1}(((1 - \omega)D - \omega U)v^{m+1,k} + \omega b^m). \tag{70}$$

We write the relation above by components:

$$v_n^{m+1,k+1} = (1 - \omega)v_n^{m+1,k} + \frac{\omega}{1 + \beta + \beta n + \alpha n^2}(v_n^{m,k} + \frac{1}{2}\alpha n^2 v_{n-1}^{m+1,k+1} + (\beta n + \frac{1}{2}\alpha n^2)v_{n+1}^{m+1,k}), \tag{71}$$

for $n = 1, \ldots, N - 1$, $m = \overline{0, M - 1}$.

For an European call option with $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000, N = 200$ we use the Gauss-Seidel method and the SOR method.

	Gauss-Seidel Method	SOR Method	
S	C_{approx}	C_{approx}	C_{exact}
4.0	$1.631276e - 006$	$2.696249e - 006$	$1.067322e - 006$
8.0	0.146176	0.158275	0.149335
10.0	0.906372	0.931284	0.916291
16.0	6.236436	6.249212	6.252287
20.0	10.223892	10.235624	10.247014

From this experiment, we notice that the number of iterations which is required to obtain the approximate solution is the same in both methods, as much in Gauss-Seidel method as well as in SOR method (in our case, the number of iterations is equal to 2001). The numerical analysis theory asserts that for a good choice of ω SOR should be much faster than Gauss-Seidel method. It is also known that the Gauss-Seidel method can be obtained from the SOR method for $\omega = 1$. The parameter of relaxation ω_{optim} is calculated using formula (69). In our example, $\omega_{optim} = 1.0483 \approx 1$. That is why the two methods are similar in this case. Figures 6 and 7 show graphical similarities. For a better approximation we must refine the grid (S, t). The time of execution is a criterion in the choice of one or another method of approximation. Refining the grid, the time is increasing and this is not acceptable. Hence, using iterative methods we lose in accuracy: a bigger error tolerance or a bigger time of execution. It is therefore better to use the direct method in order to solve the systems of linear equations.

3. Two-asset options: call on maximum option

3.1. The Black-Scholes equation. In this section we describe the Black-Scholes theory for options on more than one underlying asset. Rainbow Options refer to all options whose payoff depends on more than one underlying risky asset; each asset is referred to as a colour of the rainbow.

Consider an European option whose payoff depends on two assets S_1 and S_2. Let $V(S_1, S_2, t)$ be the option value.

The corresponding Black-Scholes equation is

$$\begin{aligned}\frac{\partial V}{\partial t}(S_1, S_2, t) + \frac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 V}{\partial S_1^2}(S_1, S_2, t) + \frac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 V}{\partial S_2^2}(S_1, S_2, t) \\ +\rho\sigma_1\sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2}(S_1, S_2, t) + rS_1 \frac{\partial V}{\partial S_1}(S_1, S_2, t) \\ +rS_2 \frac{\partial V}{\partial S_2}(S_1, S_2, t) - rV(S_1, S_2, t) = 0.\end{aligned} \tag{72}$$

where σ_1 is volatility of asset S_1, σ_2 is volatility of asset S_2 and ρ is the correlation coefficient between S_1 and S_2. Like in previous sections, r is the risk-free rate, E is the strike price of the rainbow option and T is the term to expiry of the rainbow option.

The solution domain is $\{S_1 \in [0, \infty), S_2 \in [0, \infty), t \in [0, T]\}$ and the final condition is

$$V(S_1, S_2, T) = Payoff(S_1, S_2). \tag{73}$$

Examples of rainbow options (with two assets) include:

• Call on maximum option

$$Payoff(S_1, S_2) = \max(\max(S_1, S_2) - E, 0); \tag{74}$$

• Put on maximum option

$$Payoff(S_1, S_2) = \max(E - \max(S_1, S_2), 0); \tag{75}$$

• Call on minimum option

$$Payoff(S_1, S_2) = \max(\min(S_1, S_2) - E, 0); \tag{76}$$

• Put on minimum option

$$Payoff(S_1, S_2) = \max(E - \min(S_1, S_2), 0); \tag{77}$$

• Best of two assets or cash option

$$Payoff(S_1, S_2) = \max(S_1, S_2, E) \tag{78}$$

(for details see [7], [10], [11]).

We make the same change of variables like in Section 1. From relations (8) and (9) we obtain the following form of equation (72):

$$\begin{aligned} &\frac{\partial V}{\partial t}(S_1, S_2, t) - \frac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 V}{\partial S_1^2}(S_1, S_2, t) - \frac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 V}{\partial S_2^2}(S_1, S_2, t) \\ &-\rho\sigma_1\sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2}(S_1, S_2, t) - rS_1 \frac{\partial V}{\partial S_1}(S_1, S_2, t) \\ &-rS_2 \frac{\partial V}{\partial S_2}(S_1, S_2, t) + rV(S_1, S_2, t) = 0. \end{aligned} \tag{79}$$

The final condition (73) becomes initial condition:

$$V(S_1, S_2, 0) = Payoff(S_1, S_2). \tag{80}$$

In the following we consider a call on maximum option. The option value satisfies equation (79) (we renote V by C) and the initial condition is

$$C(S_1, S_2, 0) = \max(\max(S_1, S_2) - E, 0). \tag{81}$$

The boundary conditions are:

• If $S_1 = 0$ and $S_2 = 0$, the option value is $C(0, 0, t) = 0$.

• If $S_1 = 0$ and $S_2 \neq 0$, the option value C depends only on S_2 and t :

$$\frac{\partial C}{\partial t}(S_2, t) - \frac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 C}{\partial S_2^2}(S_2, t) - rS_2 \frac{\partial C}{\partial S_2}(S_2, t) + rC(S_2, t) = 0. \tag{82}$$

This is the usual Black-Scholes formula in S_2.

• If $S_1 \neq 0$ and $S_2 = 0$, the option value C depends only on S_1 and t :

$$\frac{\partial C}{\partial t}(S_1,t) - \frac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 C}{\partial S_1^2}(S_1,t) - rS_1 \frac{\partial C}{\partial S_1}(S_1,t) + rC(S_1,t) = 0. \tag{83}$$

This is the usual Black-Scholes formula in S_1.

• If $S_1 \to \infty$ and $S_2 \to \infty$, the option value is approximative equal to S_1 or S_2.

• If $S_1 \to \infty$ and S_2 is finite, the option value is approximative equal to S_1.

• If S_1 is finite and $S_2 \to \infty$, the option value is approximative equal to S_2.

The exact solution of equation (79), obtained in particular conditions (σ_1, σ_2 and r - constants), is

$$\begin{aligned} C(S_1,S_2,t) &= S_1\left[N(\delta_1) - N_2(-d_1,\delta_1;\rho_1)\right] \\ &+ S_2\left[N(\delta_2) - N_2(-d_2,\delta_2;\rho_2)\right] \\ &+ Ee^{-r(T-t)}N_2(-d_1 + \sigma_1\sqrt{T-t}, -d_2 \\ &+ \sigma_2\sqrt{T-t};\rho) - Ee^{-r(T-t)}, \end{aligned} \tag{84}$$

where

$$d_{1,2} = \frac{\ln\frac{S_{1,2}}{E} + \frac{\sigma_{1,2}^2}{2}(T-t)}{\sigma\sqrt{T-t}}, \qquad \delta_{1,2} = \frac{\ln\frac{S_{1,2}}{S_{2,1}} + \frac{\sigma^2}{2}(T-t)}{\sigma\sqrt{T-t}}$$

$$\rho_{1,2} = \frac{\rho\sigma_{2,1} - \sigma_{1,2}}{\sigma}, \quad \sigma^2 = \sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2$$

and N is the is the cumulative distribution function for a standardised normal random variable and N_2 is the cumulative bivariate distribution for two correlated standardised normal random variables.

3.2. The numerical approximation. We have three variables S_1, S_2 and t, so we are going to divide the S_1-axis into equally spaced nodes with step ΔS_1, the S_2-axis into equally spaced nodes with step ΔS_2 and the t-axis into equally spaced nodes with step Δt.

Since the equation cannot be solved numerically for $S_1 \in [0,\infty)$ and $S_2 \in [0,\infty)$, we have to choose reasonable upper bounds for S_1 and S_2, for example four times the value of the exercise price E.

Hence the space $[0, S_{\max}] \times [0, S_{\max}] \times [0, T]$ is approximated by a grid

$$(m\Delta S_1, n\Delta S_2, k\Delta t) \in [0, M\Delta S_1] \times [0, N\Delta S_2] \times [0, K\Delta t]\,, \tag{85}$$

where $m = 0, \ldots, M$, $n = 0, \ldots, N$ and $k = 0, \ldots, K$. In which follows, we denote

$$v_{m,n}^k = C\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) \tag{86}$$

for the value of $C(S_1 S_2, t)$ at the mesh point $(m\Delta S_1, n\Delta S_2, k\Delta t)$.

We use a forward difference approximation of the time derivative, a central difference approximation of the first order S_1 and S_2 derivative and a symmetric central difference approximation of the second order S_1 and S_2 derivative, i.e.

$$\frac{\partial V}{\partial t}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) = \frac{v_{m,n}^{k+1} - v_{m,n}^k}{\Delta t} + O(\Delta t), \tag{87}$$

$$\frac{\partial V}{\partial S_1}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) = \frac{v_{m+1,n}^k - v_{m-1,n}^k}{2\Delta S_1} + O((\Delta S_1)^2), \tag{88}$$

$$\frac{\partial V}{\partial S_2}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) = \frac{v_{m,n+1}^k - v_{m,n-1}^k}{2\Delta S_2} + O((\Delta S_2)^2), \tag{89}$$

$$\begin{aligned} &\frac{\partial^2 V}{\partial S_1 \partial S_2}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) \\ &= \frac{v_{m+1,n+1}^k - v_{m+1,n-1}^k - v_{m-1,n+1}^k + v_{m-1,n-1}^k}{4\Delta S_1 \Delta S_2} \\ &+ O((\Delta S_1)^2, (\Delta S_2)^2), \end{aligned} \tag{90}$$

$$\frac{\partial^2 V}{\partial S_1^2}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) = \frac{v_{m+1,n}^k - 2v_{m,n}^k + v_{m-1,n}^k}{(\Delta S_1)^2} + O((\Delta S_1)^2), \tag{91}$$

$$\frac{\partial^2 V}{\partial S_2^2}\left(m\Delta S_1, n\Delta S_2, k\Delta t\right) = \frac{v_{m,n+1}^k - 2v_{m,n}^k + v_{m,n-1}^k}{(\Delta S_2)^2} + O((\Delta S_2)^2). \tag{92}$$

The explicit scheme for the PDE (79) is:

$$\begin{aligned}
&\frac{v_{m,n}^{k+1}-v_{m,n}^{k}}{\Delta t}-\frac{1}{2}\sigma_1^2S_1^2\frac{v_{m+1,n}^{k}-2v_{m,n}^{k}+v_{m-1,n}^{k}}{(\Delta S_1)^2}\\
&-\frac{1}{2}\sigma_2^2S_2^2\frac{v_{m,n+1}^{k}-2v_{m,n}^{k}+v_{m,n-1}^{k}}{(\Delta S_2)^2}\\
&-\rho\sigma_1\sigma_2S_1S_2\frac{v_{m+1,n+1}^{k}-v_{m+1,n-1}^{k}-v_{m-1,n+1}^{k}+v_{m-1,n-1}^{k}}{4\Delta S_1\Delta S_2}\\
&-rS_1\frac{v_{m+1,n}^{k}-v_{m-1,n}^{k}}{2\Delta S_1}-rS_2\frac{v_{m,n+1}^{k}-v_{m,n-1}^{k}}{2\Delta S_2}+rv_{m,n}^{k}=0,
\end{aligned}\tag{93}$$

for $n=\overline{1,N-1}$, $m=\overline{0,M-1}$ and $k=\overline{0,K-1}$.

For $m=0,\dots,M$ and $n=0,\dots,N$, let us denote

$$\begin{aligned}
&A(m)=\frac{1}{2}\sigma_1^2m^2\Delta t,\ B(m)=\frac{1}{2}rm\Delta t,\ C(n)=\frac{1}{2}\sigma_2^2n^2\Delta t,\\
&D(n)=\frac{1}{2}rn\Delta t,\ E(m,n)=\frac{1}{4}\rho\sigma_1\sigma_2mn\Delta t,\ F=r\Delta t.
\end{aligned}$$

Hence equation (93) becomes

$$\begin{aligned}
v_{m,n}^{k+1}&=v_{m,n}^{k}+A(m)(v_{m+1,n}^{k}-2v_{m,n}^{k}+v_{m-1,n}^{k})\\
&+C(n)(v_{m,n+1}^{k}-2v_{m,n}^{k}+v_{m,n-1}^{k})\\
&+E(m,n)(v_{m+1,n+1}^{k}-v_{m+1,n-1}^{k}-v_{m-1,n+1}^{k}+v_{m-1,n-1}^{k})\\
&+B(m)(v_{m+1,n}^{k}-v_{m-1,n}^{k})+D(n)(v_{m,n+1}^{k}-v_{m,n-1}^{k})-Fv_{m,n}^{k},
\end{aligned}\tag{94}$$

with $m=1,\dots,M-1$, $n=1,\dots,N-1$ and $k=0,\dots,K-1$.

Values $v_{m,n}^0$, $v_{0,0}^k$,$v_{0,n}^k$, $v_{m,0}^k$, $v_{M,n}^k$, $v_{m,N}^k$, $v_{M,N}^k$ with $m=0,\dots,M$, $n=0,\dots,N$ and $k=0,\dots,K$ are known from initial and boundary conditions.

For example, from (82) and (83) we obtain:

$$\begin{aligned}
v_{0,n}^{k+1}&=v_{0,n}^{k}+C(n)(v_{0,n+1}^{k}-2v_{0,n}^{k}+v_{0,n-1}^{k})\\
&+D(n)(v_{0,n+1}^{k}-v_{0,n-1}^{k})-Fv_{0,n}^{k},
\end{aligned}\tag{95}$$

for $n=1,\dots,N-1$, $k=0,\dots,K-1$ and

$$\begin{aligned}
v_{m,0}^{k+1}&=v_{m,0}^{k}+A(m)(v_{m+1,0}^{k}-2v_{m,0}^{k}\\
&+v_{m-1,0}^{k})+B(n)(v_{m+1,0}^{k}-v_{m-1,0}^{k})-Fv_{m,0}^{k},
\end{aligned}\tag{96}$$

for $m=1,\dots,M-1$, $k=0,\dots,K-1$.

3.2.1. Implementation. The explicit method is easy to implement with any programming language capable to store arrays of data. Using formula (94), we can calculate all of the values for the next time step one by one.

We consider an European call on maximum option with: $T = 0.5, E = 10.0, r = 0.1, \sigma_1 = 0.2, \sigma_2 = 0.2, \rho = 0.1$.We get the following results:

S_1	S_2	C_{approx}	C_{exact}
4.0	8.0	0.0658	0.0657
8.0	16.0	6.4879	6.4878
10.0	4.0	0.8221	0.8278
10.0	10.0	1.3280	1.3342
16.0	16.0	7.6950	7.6970
20.0	8.0	10.4877	10.4877
20.0	16.0	10.6863	10.6871

C_{exact} is calculated using the relations (84).

In Figures 8 and 9 we obtain the approximate solution for the numerical datas above.

When dealing with options on two assets, the stability problem becomes more complicated. For our example, the method is stable. The t-step, Δt, must be choosed depending on σ_1, σ_2, M and N. The number of time steps, K, must be significantly biger than M and N.We choose $M = N = 100$ and, from some calculus, $K = 401$.

Making S_1 or S_2 constant we obtain the solutions from Figure 10 and Figure 11.

Conclusions. As we have seen in the Sections 2 and 3, using the explicit finite differences method for solving partial differential equations puts severe constraints on the size of the time step. One way to overcome this problem is to use implicit finite differences schemes. The finite differences methods are suitable for solving financial problems with two or three random factors. For more random factors, the Monte Carlo simulation becomes a better method, which also works in the case of complex payoffs.

Acknowledgements. We thanks Professor Viorel Arnăutu for his support in preparing this paper.

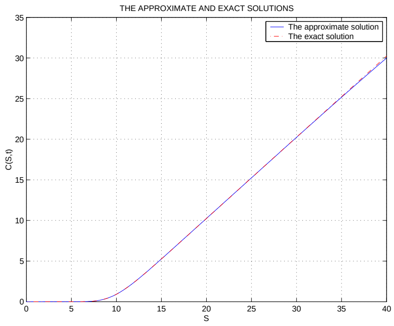

Figure 1: European call option: the explicit method. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$,$M = 2000$, $N = 200$

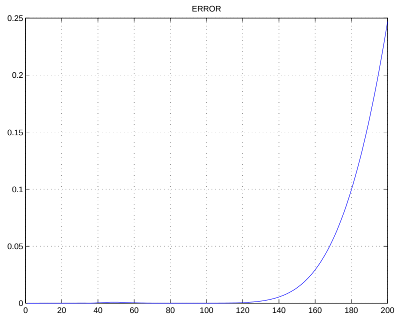

Figure 2: European call option: errors. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

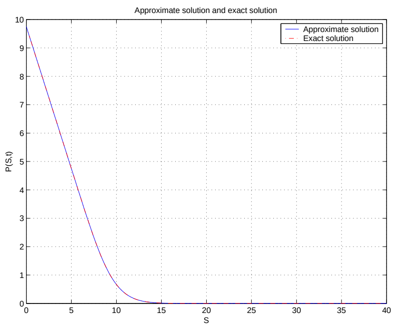

Figure 3: European put option: the fully implicit method. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

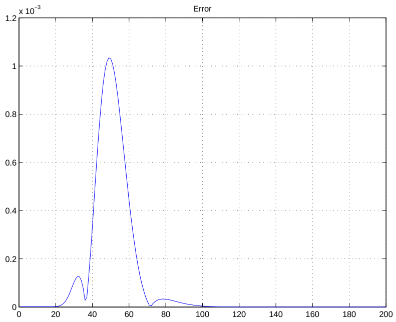

Figure 4: European put option: absolute error. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

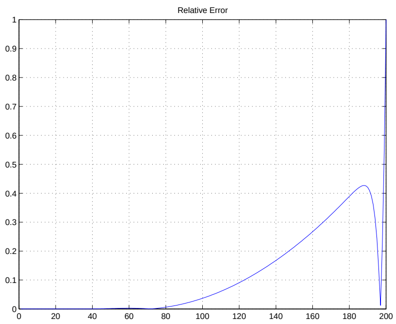

Figure 5: European put option: relative error. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

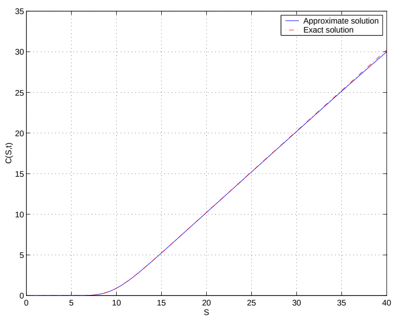

Figure 6: European call option: the Gauss-Seidel method. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

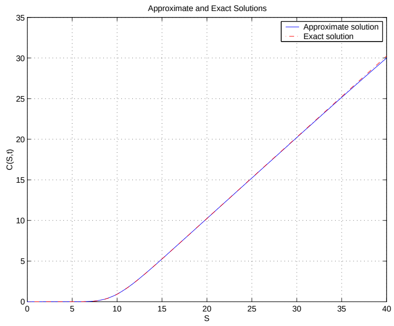

Figure 7: European call option: the SOR Method. $T = 0.25$, $E = 10.0$, $r = 0.1$, $\sigma = 0.4$, $M = 2000$, $N = 200$

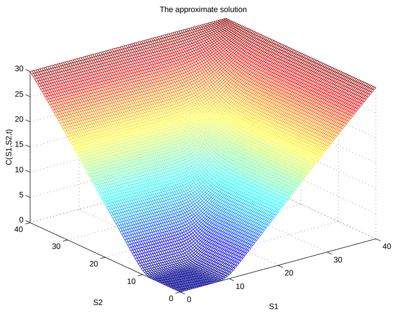

Figure 8: European call on max option: the explicit method. $T = 0.5$, $E = 10.0$, $r = 0.1$, $\sigma_1 = 0.2$, $\sigma_2 = 0.2$, $\rho = 0.1$, $M = 100$, $N = 100$

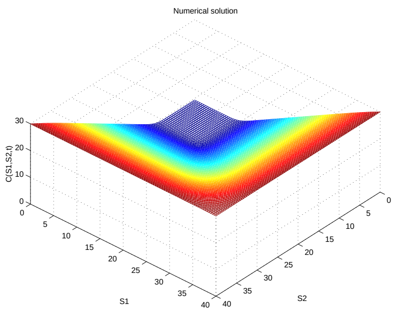

Figure 9: European call on max option: the explicit method(another view). $T = 0.5$, $E = 10.0$, $r = 0.1$, $\sigma_1 = 0.2$, $\sigma_2 = 0.2$, $\rho = 0.1$, $M = 100$, $N = 100$

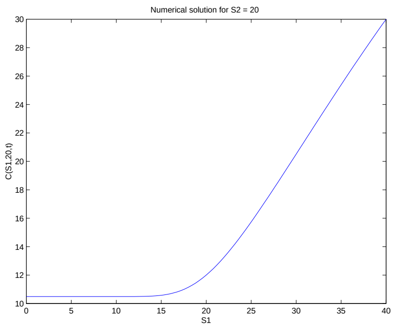

Figure 10: European call on max option: S2 = constant. $T = 0.5, E = 10.0, r = 0.1, \sigma_1 = 0.2, \sigma_2 = 0.2, \rho = 0.1, M = 100, N = 100$

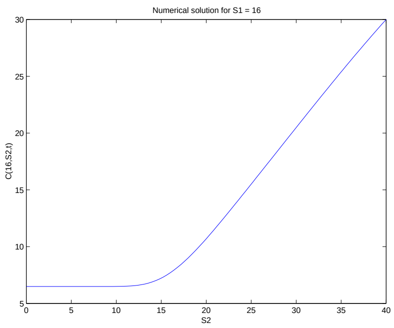

Figure 11: European call on max option: S1 = constant. $T = 0.5, E = 10.0, r = 0.1, \sigma_1 = 0.2, \sigma_2 = 0.2, \rho = 0.1, M = 100, N = 100$

REFERENCES

1. Arnăutu, V.; Neittaanmäki, P. – *Discretization estimates for an elliptic control problem*, Numer. Funct. Anal. Optim., 19 (1998), 431–464.
2. Arnăutu, V.; Neittaanmäki, P. – *Optimal Control From Theory to Computer Programs*, Solid Mechanics and its Applications, 111, Kluwer Academic Publishers Group, Dordrecht, 2003.
3. Black, F.; Scholes, M. – *The pricing of options and corporate liabilities*, Journal of Political Economy, 81 (1973), 637–654.
4. Clewlow, L.; Strickland, C. – *Implementing Derivatives Models*, John Wiley & Sons, Chicester, 1998.
5. Dura, G.; Ştefănescu, R. – *Numerical algorithms for a second order ellpitic BVP*, An. Ştiinţ. Univ. "Al. I. Cuza" Iaşi, Mat. (N.S.), 53 (2007), 103–118.
6. Ikonen, S. – *Black-Scholes-yhtälöstä ja sen numeerisesta ratkaisemisesta differenssimenetelmällä*, MD Thesis, University of Jyväskylä, Finland, 2001.

7. Johnson, H. – *Options on the minimum or the maximum of several assets*, Journal of Finance and Quantitative Analysis, 22(1987), 277-283.

8. Kerman, Jouni – *Numerical Methods for Option Pricing: Binomial and Finite-difference Approximation*, Courant Institute of Mathematical Science, New York University, 2002.

9. Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. – *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, 1990.

10. Rubinstein, M. – *Somewhere over the rainbow*, Risk Magazine, 4 (1995), 63–66.

11. Stulz, R. – *Options on the minimum or the maximum of two risky assets*, Journal of Financial Economics, 10(1982), 161–185.

12. Wilmott, P.; Howison, S.; Dewynne, J. – *The Mathematics of Financial Derivatives. A Student Introduction,* Cambridge University Press, Cambridge, 1995.

Received: 2.III.2009

Department of Mathematics,
"Al.I. Cuza" University,
Bd. Carol I, no 11, 700506 Iaşi,
ROMANIA
anamaria.mosneagu@uaic.ro