Technical report, IDE0848 , November 25, 2008

# Pricing of European type options for Levy and conditionally Levy type models

Master's Thesis in Financial Mathematics

Stepan Sushko

MASTER'S PROGRAMME IN FINANCIAL MATHEMATICS

HÖGSKOLAN HALMSTAD

School of Information Science, Computer and Electrical Engineering
Halmstad University

# Pricing of European type options for Levy and conditionally Levy type models

*Stepan Sushko*

Halmstad University
Project Report IDE0848

Master's thesis in Financial Mathematics, 15 ECTS credits

Supervisor: Prof. Lioudmila Y. Vostrikova
Examiner: Prof. Ljudmila A. Bordag
External refrees: Prof. V. Roubtsov

November 25, 2008

Department of Mathematics, Physics and Electrical Engineering
School of Information Science, Computer and Electrical Engineering
Halmstad University

# Preface

I would like to say many thanks to my scientific supervisor L.U. Vostrikova for the help, useful advices and the hospitality in France, to L.A. Bordag for the organization of the interesting program on the field of Financial Mathematics and for the help with the trip to France. Also thanks to J. Kallsen for interesting discussions about problems of Financial Mathematics. It was the very interesting and useful experience in my life, so thanks to Sweden, to the staff and groupmattes in the Halmstad university as well.

**Abstract**

In this thesis we consider two models for the computation of option prices. The first one is a generalization of the Black-Scholes model. In this generalization the volatility $\sigma$ is not a constant. In the simplest case it changes at once at a certain time moment $\tau$. In some sense this is the conditionally Levy model. For this generalized Black-Scholes model have been theoretically obtained formulas for vanilla Call/Put option prices. Under the assumption of a good prediction of the parameter $\sigma$ the obtained numerical results fit the real dara better than standard Black-Scholes model.

Second model is an exponential Levy model, where a Levy process is the CGMY process. We use the finite-difference scheme for computations of option prices. As example we consider vanilla Call/Put, Double-Barrier and Up-and-out options. After the estimation of the parameters of the CGMY process by the method of moments we obtain options prices and calculate fitting error. This fitting error for the CGMY model is smaller than for the Black-Scholes model.

# Contents

# Chapter 1

# Introduction

Actually the modern Financial institutions use the Black-Scholes model, may be with some improvements, as a standard model for option pricing. But it is well-known that this model has a number of disadvantages and describe real situations rather bad. For instance, the well-known effects of "volatility smile" and "volatility clustering" contradict the assumption of constant volatility admitted in the BS model. Note also that in the case when we use the historical volatility it can have gaps. Another unrealistic assumption on the standard BS model is a constant interest rate. Moreover, the normal distribution don't reproduce heavy tails, high peak and realted effects.

So in this thesis we try to consider more realistic models.

The first model is the generalization of the Black-Scholes model. In this model we assume that parameters $\mu$, $r$ and $\sigma$ change over time. Consider two cases: in the first case parameters change at a deterministic time moment; in the second case parameters change at a random point of time.

The second model try to rectify disadvantages concerning the normal distribution. For this purpose we use a Levy process for the underlying asset price. More concreatly, we choose the extended CGMY process (see [3]). This process is not a purely jump process and has an independed diffusion component. Since this model is analytically untractable. So we use the finite-difference method (see [4] for details).

This thesis are organized as follows. In second and third chapters we give some important definitions from the probability theory, the theory of stochastic processes and some facts related to finance (see [12] for details). In Chapter 4 we introduce the generalized Black-Scholes model, prove two theorems concerning the option prices and presente numerical results. In Chapter 5 we describe the CGMY model, the numerical method and corresponding numerical results.

# Chapter 2

# Introduction to the theory of Levy processes

## 2.1 Filtered probability space

In this chapter we will define some notions and objects from the probability theory and theory of stochastic processes.

Firstly, let us define some **probability space**:

$$(\Omega, \mathcal{F}, P), \tag{2.1}$$

where $\Omega$ is a **space of elementary events** $\omega$,

$\mathcal{F}$ is the minimal $\sigma$-algebra of subsets of $\Omega$,

$P$ is a probability measure on $\mathcal{F}$.

We define a **flow of $\sigma$-algebras** as $\mathbf{F} = (\mathcal{F}_t)_{t\geq 0}$, where $\mathcal{F}_t = \bigcap_{s>t} \mathcal{F}_s$ and $\mathcal{F}_s \subset \mathcal{F}_t$. $(\Omega, \mathcal{F}, \mathbf{F} = (\mathcal{F}_t)_{t\geq 0}, P)$ is called a **filtered probability space** or a **stochastic basis**. We complete our stochastic basis. Let us denote by

$$E = (A \in \mathcal{F} | P(A) = 0), \tag{2.2}$$

then $\mathcal{F}^+$ is complection of $\mathcal{F}$ by E. And $\mathcal{F}_t^+$ is the smallest $\sigma$-algebra of $\mathcal{F}_t$ and $E$. A probability space

$$(\Omega, \mathcal{F}^+, \mathbf{F}^+ = (\mathcal{F}_t^+)_{t\geq 0}, P) \tag{2.3}$$

is called the **filtered probability space satisfied usual conditions**.

If we apply this notion to our field, then the flow of $\sigma$-algebras is an information on market and change market's variables. For instance, an available information about the price during the time period from zero up to today on some asset or a value of some index is $\mathcal{F}_t$. In our further explanations we work with the filtered probability space which satisfied usual conditions

$$((\Omega, \mathcal{F}, ((\mathcal{F}_t)_{t\geq 0}), P)). \tag{2.4}$$

It is convinient to suppose that the probability measure $P$ belongs to a family of probability measures $\mathcal{P}$.

## 2.2 Definitions and notions important for models based on Levy processes

**Definition 1** *A stochastic process $X = (X_t)_{t\geq 0}$ on $((\Omega, \mathcal{F}, ((\mathcal{F}_t)_{t\geq 0}), P))$, $E|X_t| < \infty$ is called:*

1. *A martingale, if $E(X_t|\mathcal{F}_s) = X_s$,$P - a.s.$,*
2. *A supermartingale, if $E(X_t|\mathcal{F}_s) \leq X_s$,$P - a.s.$,*
3. *A submartingale, if $E(X_t|\mathcal{F}_s) \geq X_s$,$P - a.s.$.*

**Definition 2** *A mapping $T : \Omega \to \overline{R}_+$ such that $\{T \leq t\} \in \mathcal{F}_t$ for all $t \in R_+$ is called a* ***stopping time****.*

**Definition 3** *Stochastic process $X = X_t$ is called* ***a local martingale*** *if there exist a localizing sequence of Markov stopping times $(\tau_k)_{k\geq 0}$ ($\{\omega : \tau_k \leq t\} \in \mathcal{F}_t$), such that $\tau_k \leq \tau_{k+1}$ ($P - a.s.$) and such that $\tau_k \to \infty$, $P - a.s.$ if $k \to \infty$ and each localized sequence:*

$$X^{\tau_k} = X_{\tau_k \wedge t} \tag{2.5}$$

*is a martingale.*

**Definition 4** *A process $X$ is called a* ***semimartingale*** *if it has the following decomposition:*

$$X_t = X_0 + A_t + M_t,$$

*where $A_t$ is a process of a bounded variation and $M_t$ is a local martingale.*

**Definition 5** *A process $X = X_t$ is called* ***a cadlag process*** *if it has the right-continuous paths with a left limit.*

*For each cadlag process we define two processes $X_{t-}$ and $\Delta X$, such that*

1. *$X_{0-} = X_0$,*

2. $X_t = \lim_{s \to\ t} X_s$,

3. $\Delta X_t = X_t - X_{t-}$.

**Definition 6** ***A quadratic covariation*** *of two semimartingales $X$ and $Y$ denoted by $[X, Y]$ is a following process:*

$$[X, Y]_t = X_t Y_t - \int_0^t X_{s-} dY_s - \int_0^t Y_{s-} dX_{s-} - X_0 Y_0. \tag{2.6}$$

*We can rewrite the last expression in a differential form:*

$$d(X, Y) = X_- dY + Y_- dX + d[X, Y], \tag{2.7}$$

*this is an analog of the usual differentiation by parts.*

**Definition 7** ***A quadratic variation*** *of a semimartingale $X$ is a process denoted by $[X, X]$*

$$[X, X]_t = X_t^2 - 2 \int_0^t X_{s-} dX_s - X_0^2. \tag{2.8}$$

Let the process $(M_t)_{t \geq 0}$ is a square integrable martingale, then from the Dub-Mayer decomposition follow that there exist a nondecreasing predictable process $\langle M, M \rangle$ such that the process $M^2 - \langle M, M \rangle$ is an uniformly integrable martingale. If the process $M_t$ is a local square integrable martingale then $M^2 - \langle M, M \rangle$ is a local square integrable martingale. If we have two $L_2$ martingales $M$ and $N$ then their angle brackets

$$\langle M, N \rangle = \frac{1}{4}(\langle M + N, M + N \rangle - \langle M - N, M - N \rangle). \tag{2.9}$$

It is evident, that $\langle M, N \rangle$ is a predictable process with a bounded variation and $MN - \langle M, N \rangle$ is a local martingale, $MN - [M, N]$ is local martingale too.

**Definition 8** *Let us consider a measurable space $(E, \mathcal{E})$, then a family of nonnegative measures on $(R_+ \times E, \mathcal{B}(R_+) \otimes \mathcal{E})$*

$$\mu = \{\mu(dt, dx; \omega); \omega \in \Omega\} \tag{2.10}$$

*is called a **random measure** on $R_+ \times E$, if it satisfies a condition: $\mu(\{0\} \times E; \omega) = 0$ for every $\omega \in \Omega$.*

**Example.** (The Poisson random measure is a major example of the random measures):

If $A \in \mathcal{B}((R_+)) \otimes \mathcal{E}$ is such that

$$m(A) = E(\mu(A;\omega)) \tag{2.11}$$

and $m(A)$ is a $\sigma$-finite positive measure and measure of the intensity $m$ is such that $m(t \times E) = 0$ for each $t \in R_+$ then the random measure $\mu$ is called **a Poisson random measure**, if $m(dt, dx) = dtF(dx)$ ($F$-is a positive $\sigma$-finite measure) then $\mu$ is the homogeneous Poisson measure. This measure of the intensity $m$ is such that $m(t \times E) = 0$ for each $t \in R_+$ then the random measure $\mu$ is called the **Poisson random measure**, if $m(dt, dx) = dtF(dx)$ ($F$-is positive $\sigma$-finite measure) then $\mu$ is a homogeneous Poisson measure.

**Definition 9** *Let us consider a random measure $\mu$, if a process $W * \mu$ (a stochastic integral w.r.t. measure $\mu$) is optional for each optional function $W(t, \omega, x)$ then $\mu$ is called an **optional (predictable) random measure**.*

**Definition 10** *An optional random measure $\mu$ is called $\widetilde{\mathcal{P}} - \sigma$**-finite** if a random variable $V * \mu_\infty$ is integrable where $V$ is a strictly positive predictable function on $\widetilde{\Omega}$.*

**Definition 11** *A predictable measure $\nu$ is called a **compensator** of an optional $\widetilde{\mathcal{P}} - \sigma$-finite measure $\mu$ if*

1. *For each nonnegative $\widetilde{\mathcal{P}}$-measurable function $W$ on $(R_+, \Omega, E)$ holds $E(W * \nu)_\infty = E(W * \mu)_\infty$*

2. *$W * \mu - W * \nu$ is a local martingale and $|W| * \mu$ is a local integrable for each $\widetilde{\mathcal{P}}$-measurable function $W$ on $(R_+ \times \Omega \times E)$ and $\nu$ is unique and exists.*

Note that for a Poisson random measure its compensator is a measure of the intensity $m$.

A **jump-measures** $N_p$ of Levy processes $X = (X_t)_{t \geq 0}$ are other particular cases of the random measures (including semimartingales). Let $(b, \sigma^2, \nu)$ is a characteristic triplet of $X$ connected to the truncation function $g(x) = xI(|x| \leq 1)$, $p$ is a point process on $\mathrm{R}\backslash\{0\}$ such that $p_t = \Delta X_t$, then $N_p(du, dx)$ **is a counting measure** of the process $p_t$:

$$N_p((0,t], A) = \#\{s \in D_p \cap (0,t] : p_s \in A,\ for A \in \mathcal{B}(\mathrm{R}\backslash\{0\})\}, \tag{2.12}$$

where $D_p = \{t > 0 : \Delta X_t \neq 0\}$.

We denote by $\hat{N}_p(dudx)$ the compensator of $N_p(dudx)$. $N_p$ is a homogeneous Poisson random measure, hence

$$\hat{N}_p(dudx) = du\nu(dx). \tag{2.13}$$

Using this Poisson measure we can define a well-known **Levy-Ito decomposition of** $X_t$

$$X_t = \sigma + bt + \int_{(0,t]} \int_{|x|\leq 1} x\widetilde{N}_p(dudx) + \int_{(0,t]} \int_{|x|>1} xN_p(dudx), \tag{2.14}$$

where $\widetilde{N}_p(dudx) = N_p(dudx) - \hat{N}_p(dudx)$.

Let the bounded function $g(x)$ is **a truncation function**, i.e. it is equal to $x$ in the neighborhood of zero and it has finite support. Often we take $g(x) = xI_{\{|x|\leq 1\}}$ and $H$ is a semimartingale with the triplet $(B(g), C, \nu)$, then for $H$ exist the decomposition

$$H = H^0 + B(g) + H^c + g * (\mu^H - \nu^H) + (x - g(x)) * \mu^H, \tag{2.15}$$

which is called **a canonical decomposition of semimartingale** $H$.

We formulate the following result for semimartingales.

**Theorem 1** *.(Girsanov's theorem for semimartingales)[12]*

*If the semimartingale $H$ has a canonical decomposition with the triplet $(B, C, \nu)$ w.r.t. $P$ and $\widetilde{P} \ll^{loc} P$ and the probability density process $Z_t = \frac{\widetilde{P}_t}{P_t}, t \geq 0$, then the triplet $(\widetilde{B}, \widetilde{C}, \widetilde{\nu})$ of $H$ w.r.t. to $\widetilde{P}$ is determine by*

1. $\widetilde{B} = B + \beta C + g(x)(Y - 1) * \nu$;
2. $\widetilde{C} = C$;
3. $\widetilde{\nu} = Y\nu$,

*where*

$$\beta = \frac{d\langle Z^c, X^c\rangle}{d\langle X^c, X^c\rangle}, \quad Y = E_\mu^P\left(\frac{Z}{Z_-} I(Z_- > 0)|\widetilde{P}\right), \tag{2.16}$$

*where $E_\mu^P$ is an averaging by the measure $\theta_\mu^P = E(W * \mu)$ for all measurable functions $W = W(\omega, t, x)$.*

The next theorem is a consequence of the previous result of Girsanov, but for diffusion processes. Before we start with formulation of this theore we give a definnition of a Wiener process.

**Definition 12** *A Wiener process* $W = (W_t)_{t \geq 0}$ *that starting from 0 is a process which is varified as*

1. $W_0 = 0$;
2. *For any sequence* $0 \leq t_0 < t_1 < \ldots < t_n$ *and* $n \geq 1$ *the random variables* $W_{t_0}, W_{t_1} - W_{t_0}, \ldots, W_{t_n} - W_{t_{n-1}}$ *are independent;*
3. $Law(W_t - W_s) = N(0, t-s)$.

A consequence of this definition is a property $E(W_t W_s) = \min(t, s)$.

Let's denote an nonanticipating functionals $a(t,x)$, $\widetilde{a}(t,x)$ and $b(t,x) > 0$ which satisfies following conditions

$$\int_0^t b(s,X)dW_s \text{ exists } P - a.s. \text{ for } t \leq T, \tag{2.17}$$

$$\int_0^T |a(s,X)|ds < \infty, \ (P - a.s.). \tag{2.18}$$

$$\int_0^T |\widetilde{a}(s,X)|ds < \infty, (P - a.s.), \tag{2.19}$$

and

$$\int_0^T \left(\frac{\widetilde{a}(s,X) - a(s,X)}{b(s,X)}\right)^2 ds < \infty, (P - a.s.). \tag{2.20}$$

Beside we denote $Z = (Z_t)_{t \geq 0}$ as the following process

$$Z_t = \exp \int_0^t \left(\frac{\widetilde{a}(s,X) - a(s,X)}{b(s,X)}\right) dW_s - \frac{1}{2}\int_0^t \left(\frac{\widetilde{a}(s,X) - a(s,X)}{b(s,X)}\right)^2 ds. \tag{2.21}$$

After that we can formulate following theorem.

**Theorem 2** ***(Girsanov's theorem for diffusion processes)***

*Let* $X$ *is diffusion process with the differential*

$$dX_t = a(t,X)dt + b(t,X)dW_t, \tag{2.22}$$

*where* $W$ *is a Wiener process. If* $E(Z_T) = 1$ *we can consider process* $Z$ *as density of probability, so we can change the probability measure* $P$ *to measure* $P^*$

$$\frac{dP^*}{dP} = Z_T, \tag{2.23}$$

*then with respect to the new measure* $P^*$ *the process* $X$ *has a following differential*

$$dX_t = \widetilde{a}(t,X)dt + b(t,X)d\widetilde{W}_t, \tag{2.24}$$

*where* $\widetilde{W}$ *is a Wiener process w.r.t.* $P^*$.

**Corollary 1:**

Let $B_t$ be a Wiener process with respect to $P$, $W_t$ is a Wiener process with respect to $P^*$, where $P$ and $P^*$ are equivalent measures such that

$$\frac{dP_t^*}{dP_t} = e^{\alpha B_T + \frac{1}{2}\alpha^2 T} = e^{\alpha(W_T - \alpha T) + \frac{1}{2}\alpha^2 T} = e^{\alpha W_T - \frac{1}{2}\alpha^2 T},$$

then

$$E(f(B_t + \alpha t, t \leq T)) = E^*(\frac{dP}{dP^*} f(B_t + \alpha t, t \leq T))$$

$$= E^*(e^{\alpha W_T - \frac{1}{2}\alpha^2 T} f(W_t, t \leq T)) = E(e^{\alpha B_T - \frac{1}{2}\alpha^2 T} f(B_t, t \leq T)),$$

where $E^*$ is the expectation with respect to $P^*$, $E$ is the expectation with respect to $P$.

# Chapter 3

# Levy processes and their use

## 3.1 Necessary facts about Levy processes

**Definition 13** *A real valued stochastic process $X = (X_t)_{t \geq 0}$ that starts from zero is called **a Levy process** if it satisfies following conditions*

1. $X_0 = 0, P - a.s.;$
2. *It has independent increments: for any sequence $0 \leq t_0 < t_1 < \ldots < t_n$ and $n \geq 1$ random variables $X_{t_0}, X_{t_1} - X_{t_0}, \ldots, X_{t_n} - X_{t_{n-1}}$ are independent;*
3. *Stationary increments: for any $s \geq 0$ and $t \geq 0$*

$$Law(X_{t+s} - X_s) = Law(X_t - X_0).$$

A Levy process has following properties:

1. The stochastic continuity: for any $t \geq 0$ and $\epsilon > 0$

$$\lim_{s \to t} P(|X_s - X_t| > \epsilon) = 0;$$

2. The trajectories of process $X_t$ are right-continuous functions with left-hand limits for $t > 0$.

**Definition 14** *A random variable $X$ is called **infinitely divisible**, if there exists a sequence of independent identically distributed random variables $X_{n1}, \ldots, X_{nn}$ such that $Law(X) = Law(X_{n1} + \ldots + X_{nn})$.*

*Let $b \in R$, $c \geq 0$ and $\nu(dx)$ is a positive measure on $R$ such that $\nu(0) = 0$, $\int_R (|x|^2 \wedge 1)\nu(dx) < \infty$. For this type of random variables a characteristic function is given by:*

$$\varphi(\theta) = Ee^{i\theta X} = e^{\psi(\theta)}, \tag{3.1}$$

*where*

$$\psi(\theta) = i\theta b - \frac{1}{2}\theta^2 c + \int_R (e^{i\theta x} - 1 - \theta x I_{\{|x|\leq 1\}})\nu(dx). \tag{3.2}$$

It is important to mention that the triplet of characteristics $(b, c, \nu)$ is unique for each $\varphi(\theta)$. Note that the Levy processes as processes with stationary independent increments have an infinitely divisible distribution for each t, i.e. it's characteristic function is determined by the Levy-Khinchin formula. We put

$$\varphi_t(\theta) = Ee^{i\theta X_t} = e^{it\psi(\theta)}, \tag{3.3}$$

where $X_t$ is a Levy process, then using the Levy-Khinchin formula

$$\psi_t(\theta) = i\theta B_t - \frac{1}{2}\theta^2 C_t + \int_R (e^{i\theta x} - 1 - \theta x I_{\{|x|\leq 1\}})\nu_t(dx). \tag{3.4}$$

The triplet $(B_t, C_t, \nu(dx))$ can be interpreted in the following way:

1. $B_t$ is an average of $X_t$;

2. $C_t$ is a variation of a continuous part of $X_t$;

3. $v_t(dx)$ is a probability of jumps with different sizes.

As a result we can say that since $X_t$ is a stationary and has independent increments we obtain next equality

$$\varphi_{t+s}(\theta) = \varphi_t(\theta)\varphi_s(\theta). \tag{3.5}$$

Because of the last expression and the uniqueness of the triplet it is clear that $B_t = tB$,$C_t = tC$,$\nu_t(dx) = t\nu(dx)$. Hence we have

$$\varphi_t(\theta) = e^{t\psi(\theta)}, \tag{3.6}$$

where

$$\psi(\theta) = i\theta B - \frac{1}{2}\theta^2 C + \int_R (e^{i\theta x} - 1 - \theta x I_{\{|x|\leq 1\}})\nu(dx). \tag{3.7}$$

Let us give some information about the Markov property. If we fix any moment of time, then the future behavior of the process doesn't dependend on the past and depends on the current value of the process. The family of Markov processes is very useful and mostly used in practice.

**Theorem 3** *(The simple Markov property) Let $f \in L^\infty(R^d)$, $X$ is a Levy process. Then the following equality hold*

$$E_{X_t}(\omega)[f(X_h)] = E_x[f(X_{t+h})|\mathcal{F}_\sqcup](\omega). \tag{3.8}$$

**Theorem 4** *(The Markov property) Let $\tau$ be a stopping time with respect to $\mathcal{F}_t$ such that $P(\tau < \infty) > 0$ then*

$$E_{X_\tau}[f(X_h)] = E_x[f(X_{\tau+h})|\mathcal{F}_\tau](\omega), \text{ for any } h \geq 0. \tag{3.9}$$

**Theorem 5** *(The Infinitesimal generator) Let $X$ be a Levy process, if $f$ is a function on the Schwartz space of rapidly decreasing functions then*

$$Af(x) = -\langle B, f'(x)\rangle + \frac{1}{2}\sum_{i,j=1,\dots,d} C_{ij} f''_{ij}(x) + \int_{R^d} (f(x+y) - f(x) - I_{|y|<1}\langle y, f'(x)\rangle)\nu(dy). \tag{3.10}$$

**Theorem 6** *(The Inverse Kolmogorov equation) Let $X$ be a Levy process, $f$ is a function on the Schwartz space of rapidly decreasing functions. If we denote*

$$u(t,x) = E^x[f(X_t)], \tag{3.11}$$

*than $u(t,x)$ satisfy conditions of the following problem*

$$\frac{\partial u}{\partial t} = Au, \ t > 0, \ x \in R^n, \quad u(0,x) = f(x), \ x \in R^n. \tag{3.12}$$

## 3.2 Models in Finance

Nowardays the main and the most used and the widely known model for the option pricing is the Black-Scholes model. This model is very convenient and is relatively simple for computations. Using this model in many situations we can obtain results which are enough good correlated with the real world.

But it is well-known that this model has some disadvantages by the reflection of the behavior of financial assets such as

- the clustering effect,
- the smile/smirk of volatility,
- jumps in the price process,
- gaps in the historical volatility and etc.

That is the reason why people try to develop new models which describe the real world more precise. Models based on Levy processes are the models of such type.

**Definition 15** *(The Dolean-Dade exponent)*

*Let $X = (X_t, \mathcal{F}_t)_{t\geq 0}$ and $Y = (Y_t, \mathcal{F}_t)_{t\geq 0}$ be cadlag processes, then we can consider the Dolean-Dade equation:*

$$dY_t = Y_{t-} dX_t, Y_0 = 1 \tag{3.13}$$

*or in the integral form:*

$$Y_t = 1 + \int_0^t Y_{s-} dX_s. \tag{3.14}$$

*Using Ito's formula for semimartingales we can find the solution of equation 3.13. Let us denote*

$$X_t^1 = X_t - X_0 - \langle X^c, X^c \rangle_t, \tag{3.15}$$

$$X_t^2 = \prod_{0<s\leq t} (1 + \Delta X_s) e^{-\Delta X_s}, \tag{3.16}$$

$$X_0^2 = 1 \tag{3.17}$$

*and $F(x_1, x_2) = e^{x_1} x_2$, then the process $\mathcal{E}(X) = (\mathcal{E}(X)_t, \mathcal{F}_t)_{t\geq 0}$ which is equal to*

$$\mathcal{E}(X) = F(X_t^1, X_t^2) = e^{X_t - X_0 - \langle X^c, X^c \rangle_t} \prod_{0<s\leq t} (1 + \Delta X_s) e^{-\Delta X_s} \tag{3.18}$$

*is a semimartingale and is the unique solution of the Dolean equation. The process $\mathcal{E}(X)_t$ is called* ***the stochastic exponent (Dolean's exponent).***

Note that in our context it is naturally to assume that today's price on some asset $S_t$ belongs to $\mathcal{F}_t$.

In models which are based on exponential Levy processes the price is determined by

$$S_t = S_0 e^{L_t} = S_0 \mathcal{E}(L_t), \tag{3.19}$$

where $L_t$ is a Levy process. Sometimes, $L_t$ is tranformed into the following form

$$L_t = \sigma W_t + H_t. \tag{3.20}$$

In this case $L_t$ is a sum of the continuous and discontinuous components.

In the discrete case there exist two ways for the representation of the price process $S_n (\in \mathcal{F}_n)$.

The first way is called **compound returns**. In this case for $S_n$ presentation the "common" exponent is used

$$S_n = S_0 e^{H_n}, \tag{3.21}$$

where $H_n, n \geq 0$ is a sum of $h_i(\omega) \in \mathcal{F}_i$. Hence we have

$$H_n = ln \frac{S_n}{S_0} \tag{3.22}$$

and

$$h_n = ln \frac{S_n}{S_{n-1}} \text{ (returns).} \tag{3.23}$$

The second way is called **simple returns**. On this way we use the stochastic exponent. If we take $\hat{h}_n = \frac{\Delta S_n}{S_{n-1}}$ and $\hat{H}_n = \sum_{1 \leq k \leq n} \hat{h}_k$, we can rewrite our formula for the asset price $S_t$ in the form

$$S_n = S_0 \prod_{1 \leq k \leq n} (1 + \hat{h}_k). \tag{3.24}$$

We can continue our calculations and rewrite the last expression as

$$S_n = S_0 \prod_{1 \leq k \leq n} (1 + \Delta \hat{H}_k) = S_0 e^{\hat{H}_n} \prod_{1 \leq k \leq n} (1 + \Delta \hat{H}_k) e^{-\Delta \hat{H}_k}, \tag{3.25}$$

and now we can use the Dolean-Dade exponent

$$\mathcal{E}(H_n) = e^{\hat{H}_n} \prod_{1 \leq k \leq n} (1 + \Delta \hat{H}_k) e^{-\Delta \hat{H}_k}. \tag{3.26}$$

This means that we have the second formula for $S_n$

$$S_n = S_0 \mathcal{E}(\hat{H})_n. \tag{3.27}$$

The explicit formula for a Levy process is known only in some cases, for example:

1. Brownian motion + Compound Poisson process model;
2. Logstable process model;
3. VG+CGMY;
4. Hyperbolic model;
5. GH model;
6. Models based on process subordinated to the Wiener process.

We can compute the option price for a Levy process based model using the Fourier transformation (FT) in two following ways:

1. Using the real data of the underlying asset (Eberlein, Keller, Prause 1995);

2. Using the real data of the option prices.

Another approach for computatation option prices using Levy process based models is solving of the corresponding Partial integro-differential equation. For this purpose can be used following numerical methods, for example

1. Finite diference method;

2. Galerkin method.

## 3.3 Martingale Measures

Recall that an option price is defined as

$$C(T, K) = E^* \left[ \frac{S_T}{B_T} \right], \tag{3.28}$$

where $E^*$ is the expectation with respect to a martinagale measure $P^*$.

In the case of models which are based on a Levy process $S_t$ has the following form

$$S_t = S_0 e^{L_t} = S_0 \mathcal{E}(L_t), \tag{3.29}$$

where $L_t$ is a Levy process.

But this type of models are not complete, hence the martingale measure is not unique. There exist two main ways to compute this expectation.

The first way: try to estimate parametres of a Levy process (the triplet) under a matringale measure using the real data from the market for option prices and then calculate the expectation with respect to a martingale measure.

The second way: estimate parameters of a Levy process under the physical measure $P$, choose some martingale measure and then using explicit expressions recalculate parameters under the physical measure to parameters with respect to the martingale measure.

The martingale measures is divided in two groups:
Esscher transformed martingale measures, where are

- Mean Correcting MM;

- Esscher MM;

Minimal distance martingale measures:

- Variance Optimal MM;
- Minimal Entropy MM;
- Utility Based MM.

Now we give an example of conditions and expressions for the recalculation of the parameters of the characteristic triplet of a Levy process.

Consider a Minimal Entropy martingale measure. Assume that the following condition holds

$$\int_{R\setminus\{0\}} |x| I_{|x|\leq 1} \nu(dx) < \infty, \tag{3.30}$$

then the characteristic triplet of $L_t$ is given by $(\gamma_0, \sigma^2, \nu)$ and $L_t$ has the explicit expression

$$L_t = \sigma W_t + \gamma_0 + \int_0^t \int_{R\setminus\{0\}} x N_p(dsdx). \tag{3.31}$$

using this consideration we formulate a theorem.

**Theorem 7** *Let the process $L_t$ be defined by equation 3.31 and $\theta^*$ can be found from the equation*

$$\gamma_0 + \left(\frac{1}{2} + \theta^*\right)\sigma^2 + \int_{R\setminus\{0\}} (e^x - 1) e^{\theta^*(e^x-1)} \nu(dx) = r, \tag{3.32}$$

*and measure $P^*$ is defined by the Radon-Nikodim derivative*

$$\frac{dP^*}{dP} = exp\{\theta^* \sigma W_t - \frac{1}{2}(\theta^*\sigma)^2 t + \int_0^t \int_{R\setminus\{0\}} \theta^*(e^x - 1) N_p(dsdx)$$

$$- \int_0^t \int_{R\setminus\{0\}} (e^{\theta^*(e^x-1)} - 1)\nu(dx)ds\}, \tag{3.33}$$

*then*

1. *$P^*$ is MEMM of the process $S_t$.*
2. *The process $L_t$ is a Levy process w.r.t. $P^*$ with the characteristic triplet $(\gamma_0^*, \sigma^2, \nu^*)$*

$$\gamma_0^* = \gamma_0 + \theta^* \sigma^2,$$
$$\nu^*(dx) = e^{\theta^*(e^x-1)} \nu(dx),$$

(see [10]).

# Chapter 4

# The Black-Scholes model with changing parameters

## 4.1 The case of deterministic time of change of the parameters

First of all let us recall what is the standard Black-Scholes model. Let $W$ be a Wiener process on some filtered probability space $(\Omega, \mathcal{F}_t, \mathbf{F}, P)$ with the filtration satisfying usual conditions. Then in the model the price of underlying assets is given by a geometric Brownian motion, namely

$$S_t = S_0 e^{(\mu - \frac{\sigma^2}{2})t + \sigma W_t},$$

$$B_t = B_0 e^{rt}, \tag{4.1}$$

where a drift $\mu$ and a volatility $\sigma > 0$ of a risky asset are supposed to be deterministic constants. $S_0$ is the initial value of $S$, $r$ is an interest rate, $B_0$ is the initial value of a bank account.

The price of vanilla Call/Put option is given by

$$C(S_T, K) = e^{-rT} E_P^*((S_T - K)^+), \tag{4.2}$$

$$P(S_T, K) = e^{-rT} E_P^*((K - S_T)^+), \tag{4.3}$$

where the expectation is taken under the measure $P^*$. We notice that in the case of a geometric Brownian motion based models the martingale measure is unique. Let $P_t, P_t^*$ be the restriction of $P, P^*$ on $F_t$. The Radon-Nikodim derivative $Z_t$ of $P_t^*$ with respect to $P_t$ is given by

$$\frac{dP_t^*}{dP_t} = Z_t = exp\{-\frac{\mu - r}{\sigma} W_t - \frac{1}{2}\left(\frac{\mu - r}{\sigma}\right)^2 t\}. \tag{4.4}$$

According to the Black-Scholes formula

$$C(S_T, K) = S_0 N(d_1) - Ke^{-rT} N(d_2), \tag{4.5}$$

$$P(S_T, K) = Ke^{-rT} N(-d_2) - S_0 N(-d_1), \tag{4.6}$$

where $N(\cdot)$ is a distribution function of the standard $\mathcal{N}(0,1)$ random variable.

We consider two generalized B-S models when parameters $r_1$, $\mu_1$ and $\sigma_1$ change on $r_2$, $\mu_2$ and $\sigma_2$ at a certain deterministic or random moment $\tau$. If it is a random moment then we suppose that $\tau$ is a stopping time. In this case the price of a risky asset is given by

$$S_t = \begin{cases} S_0 e^{(\mu_1 - \frac{\sigma_1^2}{2})t + \sigma_1 W_t}, \text{ if } 0 \le t \le \tau, \\ S_\tau e^{(\mu_2 - \frac{\sigma_2^2}{2})(t-\tau) + \sigma_2 (W_t - W_\tau)}, \text{ if } \tau < t \le T; \end{cases} \tag{4.7}$$

and

$$B_t = \begin{cases} B_0 e^{r_1 t} ,\text{if } 0 \le t \le \tau, \\ B_\tau e^{r_2 (t-\tau)}, \text{ if } \tau < t \le T. \end{cases} \tag{4.8}$$

Our goal is to calculate the price of vanilla Call/Put options, namely (for the simplicity we assume that $B_0 = 1$):

$$C(S_T, K) = \frac{1}{B_T} E_{P^*}((S_T - K)^+), \tag{4.9}$$

$$P(S_T, K) = \frac{1}{B_T} E_{P^*}((K - S_T)^+). \tag{4.10}$$

**Theorem 8** *Suppose that $\tau$ is a deterministic time ($\tau \in [0,T]$) of a change of the parameters $r_1$, $\mu_1$ and $\sigma_1$ to $r_2$, $\mu_2$ and $\sigma_2$, in the generalized B-S model. Then, there exist the unique equivalent measure $P^*$ such that the Radon-Nikodim derivative of*

$$\frac{dP_t^*}{dP_t} = Z_t = \begin{cases} \exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_t - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 t\}, \\ \text{if } 0 \le t \le \tau, \\ \exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_\tau - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 \tau - \frac{\mu_2 - r}{\sigma_2}(W_t - W_\tau) - \frac{1}{2}\left(\frac{\mu_2 - r}{\sigma_2}\right)^2 (t-\tau)\}, \\ \text{if } \tau < t \le T. \end{cases} \tag{4.11}$$

*Under $P^*$ we have the equivalence in law*

$$S_t = \begin{cases} S_0 e^{\sigma_1 W_t^* + (r_1 - \frac{\sigma_1^2}{2})t}, & if\ 0 \le t \le \tau, \\ S_\tau e^{\sigma_2 (W_t^* - W_\tau^*) + (r_2 - \frac{\sigma_2^2}{2})(t-\tau)}, & if\ \tau < t \le T; \end{cases} \tag{4.12}$$

*where $W^*$ is a Wiener process w.r.t $P^*$.*

*In this case*

$$C(S_T, K) = S_0 N(d_3) - K e^{-r_1\tau - r_2(T-\tau)} N(d_4), \tag{4.13}$$

$$P(S_T, K) = K e^{-r_1\tau - r_2(T-\tau)} N(-d_4) - S_0 N(-d_3), \tag{4.14}$$

*where*

$$d_3 = \frac{ln\frac{S_0}{K} + (r + \frac{\sigma_1^2}{2})\tau + (r + \frac{\sigma_2^2}{2})(T-\tau)}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}}, \tag{4.15}$$

$$d_4 = \frac{ln\frac{S_0}{K} + (r - \frac{\sigma_1^2}{2})\tau + (r - \frac{\sigma_2^2}{2})(T-\tau))}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}} \tag{4.16}$$

*and $N(\cdot)$ is distribution function of a standard normal variable.*

**Proof:** Let us define the process $Z$ using 4.11.

Now we prove that the process $Z$ is a martingale. Consider two possible cases: $t \le \tau$ and $t > \tau$. Let $t \le \tau$, in this case the process $Z$ has a representation

$$Z_t = \exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_t - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 t\}, \tag{4.17}$$

then it's expectation is

$$EZ_t = E[\exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_t - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 t\}] =$$

$$\exp\{-\frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 t\} E[\exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_t\}] =$$

$$\exp\{-\frac{a^2 t}{2}\} E[\exp\{a W_t\}]. \tag{4.18}$$

For simplicity, we denote $a = -\frac{\mu_1 - r_1}{\sigma_1}$. Now we calculate the last expectation (remember that $W_t \in \mathcal{N}(0, t)$)

$$E[e^{\{aW_t\}}] = \frac{1}{\sqrt{2\pi t}} \int_{-\infty}^{+\infty} e^{ax} e^{-\frac{1}{2}\frac{x^2}{t}} dx =$$

Transform the expression in the argument of the exponent to a complete square

$$= \frac{1}{\sqrt{2\pi t}} \int_{-\infty}^{+\infty} e^{-\frac{1}{2}(\frac{x^2}{t} - 2ax + a^2t) + \frac{a^2t}{2}} dx =$$

$$\frac{e^{\frac{a^2t}{2}}}{\sqrt{2\pi t}} \int_{-\infty}^{+\infty} e^{-\frac{1}{2}(\frac{x^2}{t} - 2ax + a^2t)} d(\frac{x}{\sqrt{t}} - a\sqrt{t}) = e^{\frac{a^2t}{2}}. \tag{4.19}$$

Therefore we proved that $EZ_t = 1$.

In the case when $t > \tau$ we have

$$EZ_t = E[Z_\tau \exp\{-\frac{\mu_1 - r_1}{\sigma_1}(W_t - W_\tau) - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 (t - \tau)\}].$$

Like in the first case we denote

$$a_1 = -\frac{\mu_1 - r_1}{\sigma_1},\ a_2 = -\frac{\mu_2 - r_2}{\sigma_2},$$

hence

$$EZ_t = \exp\{-\frac{a_1^2}{2} - \frac{a_2^2}{2}\}E[\exp\{a_1 W_\tau\}\exp\{a_2(W_t - W_\tau)\}].$$

There after we use the property of independent increments of a Wiener process

$$EZ_t = \exp\{-\frac{a_1^2}{2}\}E[e^{\{a_1 W_\tau\}}]\exp\{-\frac{a_2^2}{2}\}E[e^{\{a_2(W_t - W_\tau)\}}].$$

Using results from the first case (4.18, 4.19) we obtain that $EZ_t = 1$. Therefore we define the Radon-Nikodim derivative using the process $Z$.

On the second step, we prove equation 4.12 (the representation for the process $S$) with respect to the probability measure $P^*$.

If the probability measure $P^*$ is defined using the density process $Z$ as the Radon-Nikodim derivative, then according to the Girsanov theorem for diffusion processes

$$W_t = W_t^* - \int_0^t (\frac{\mu_1 - r_1}{\sigma_1})ds = W_t^* - \frac{\mu_1 - r_1}{\sigma_1}t, \tag{4.20}$$

where $W^*$ is a Wiener process with respect to the measure $P^*$. Hence we can rewrite expression in a power of an exponent in the representation of $S$ when $t \le \tau$ as

$$(\mu_1 - \frac{\sigma_1^2}{2})t + \sigma_1 W_t = (\mu_1 - \frac{\sigma_1^2}{2})t + \sigma_1(W_t^* - \frac{\mu_1 - r_1}{\sigma_1}) =$$

$$(r_1 - \frac{\sigma_1^2}{2})t + \sigma_2 W_t^*.$$

After an analogous operation we obtain the necessary expression for the case when $t > \tau$, i.e. we have

$$S_t = \begin{cases} S_0 e^{\sigma_1 W_t - \frac{\sigma_1^2}{2} t}, & \text{if } 0 \le t \le \tau, \\ S_\tau e^{\sigma_2(W_t - W_\tau) - \frac{\sigma_2^2}{2}(t-\tau)}, & \text{if } \tau < t \le T. \end{cases} \tag{4.21}$$

Now we can start the calculations of the option price for the model B-S with changing parameters. An option price for a Call option is given by

$$C(S_T, K) = \frac{1}{B_T} E^*((S_T - K)^+), \tag{4.22}$$

where $E^*$ is an expectation w.r.t. the martingale measure ($P^*$) and we can rewrite it as

$$E^*((S_T - K)^+) = E^*((S_T - K) I_{\{S_T > K\}}) \tag{4.23}$$

$$E^*(S_T I_{\{S_T > K\}}) - K E^*(I_{\{S_T > K\}}) = E_1 - K E_2. \tag{4.24}$$

Firstly, we will find an expectation $E_2$

$$E_2 = E^*(I_{S_T > K}) = E^*\left( I_{\left\{ S_0 exp\{d + \sigma_1 W_\tau^* + \sigma_2(W_T^* - W_\tau^*)\} > K \right\}} \right),$$

where

$$d = (r_1 - \frac{\sigma_1^2}{2})\tau + (r_2 - \frac{\sigma_2^2}{2})(T - \tau). \tag{4.25}$$

From $P^*$ and $W^*$ we can proceed to $P$ and $W$, but the expectation remains the same and it will be equal to

$$P(\sigma_1 W_\tau^* + \sigma_2(W_T^* - W_\tau^*) > ln\frac{K}{S_0} - d)$$

To compute this probability we find a variance $\sigma_3^2$ using a property of independent increments of $W$ again

$$\sigma_3^2 = V[\sigma_1 W_\tau + \sigma_2(W_T - W_\tau)] = V[\sigma_1 W_\tau] + V[\sigma_1(W_T - W_\tau)] = \sigma_1^2\tau + \sigma_1(T-\tau) \tag{4.26}$$

Hence

$$P(\sigma_1 W_\tau^* + \sigma_2(W_T^* - W_\tau^*) > ln\frac{K}{S_0} - d) = P(G > \frac{1}{\sigma_3}(ln\frac{K}{S_0} - d)) = N(d_4),$$

where $G \in \mathcal{N}(0,1)$, $N(\cdot)$ is the distribution function of the standard normal variable and $d_4$ is given by

$$d_4 = \frac{ln\frac{S_0}{K} + (r - \frac{\sigma_1^2}{2})\tau + (r - \frac{\sigma_2^2}{2})(T-\tau))}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}}.$$

Now we calculate expectation $E_1$. It is defined by the expression

$$E_1 = E^*[S_T I_{\{S_T > K\}}]$$

to simplify notations for this expectation we again proceed from $P^*$ and $W^*$ to $P$ and $W$, then using $d$ (4.25) we obtain

$$E^*[S_T I_{\{S_T > K\}}] =$$

$$S_0 e^d E[\exp\{\sigma_1 W_\tau + \sigma_2(W_T - W_\tau)\} I_{\left\{S_0 e^d \exp\{\sigma_1 W_\tau + \sigma_2(W_T - W_\tau) > K\}\right\}}]$$

or, using $G \in \mathcal{N}(0,1)$ and $\sigma_3$ (4.26)

$$= S_0 e^d E\left[e^{\sigma_3 G} I\{\sigma_3 G > ln\frac{K}{S_0} - d\}\right]$$

then by the definition of the expectation

$$= \frac{S_0 e^d}{\sqrt{2\pi}} \int_{\frac{1}{\sigma_3}(ln\frac{K}{S_0} - d)}^{+\infty} e^{\sigma_3 x - \frac{1}{2}\frac{x^2}{\sigma_3}} dx$$

after that we reduce the expression in the power of the exponent under integral into complete square

$$= \frac{S_0 e^d}{\sqrt{2\pi}} \int_{\frac{1}{\sigma_3}(ln\frac{K}{S_0} - d)}^{+\infty} e^{-\frac{1}{2}(x-\sigma_3)^2 + \frac{\sigma_3^2}{2}} dx = \frac{S_0 e^{d + \frac{\sigma_3^2}{2}}}{\sqrt{2\pi}} \int_{\frac{1}{\sigma_3}(ln\frac{K}{S_0} - d)}^{+\infty} e^{-\frac{1}{2}(x-\sigma_3)^2} dx$$

$$= \frac{S_0 e^{d + \frac{\sigma_3^2}{2}}}{\sqrt{2\pi}} \int_{\frac{1}{\sigma_3}(ln\frac{K}{S_0} - d) - \sigma_3}^{+\infty} e^{-\frac{1}{2}(x-\sigma_3)^2} d(x-\sigma_3)$$

after an easy simplification we obtain the following result

$$S_0 e^{r_1\tau + r_2(T-\tau)} N(d_3),$$

where

$$d_3 = \frac{ln\frac{S_0}{K} + (r + \frac{\sigma_1^2}{2})\tau + (r + \frac{\sigma_2^2}{2})(T-\tau)}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}}.$$

The final result of the theorem (4.13) for a Call option

$$C(S_T) = S_0 N(d_3) - K e^{-r_1\tau - r_2(T-\tau)} N(d_4).$$

Easy to obtain an analogous formula for the price of a Put option using a Put-Call parity, which is based on economical considerations (see [6], section 8.4)

$$C(S_T, K) + Ke^{-rT} = P(S_T, K) + S_0$$

or in our case of changing parameters

$$C(S_T, K) + Ke^{-r_1\tau - r_2(T-\tau)} = P(S_T, K) + S_0. \tag{4.27}$$

Recall a property of the distribution function $N(\cdot)$ of a standard normal random variable

$$N(d) + N(-d) = 1.$$

From the Put-Call parity (4.27) and obtained formula for a Call option (4.13) we get

$$P(S_T, K) = -S_0 + Ke^{-r_1\tau - r_2(T-\tau)} + C(S_T, K) =$$

$$-S_0(1 - N(d_3)) + Ke^{-r_1\tau - r_2(T-\tau)}(1 - N(d_4))$$

$$Ke^{-r_1\tau - r_2(T-\tau)} N(-d_4) - S_0 N(-d_3).$$

Now we will generalize our results for the case when $\tau$ is a random variable.

## 4.2 The case of a random time of the parameters changing

**Theorem 9** *Suppose that $\tau$ is a stopping time, $\tau \in [0,T]$, of parameters changing $r_1$, $\mu_1$ and $\sigma_1$ to $r_2$, $\mu_2$ (independent of a Wiener process) and $\sigma_2$ in the generalized B-S model. Then there exists the unique equivalent measure $P^*$ such that the Radon-Nikodim derivative of $P^*$ with respect to $P$ is given by the same expression 4.11 like in previous case, i.e.*

$$\frac{dP_t^*}{dP_t} = Z_t = \begin{cases} \exp\{-\frac{\mu_1-r_1}{\sigma_1}W_t - \frac{1}{2}\left(\frac{\mu_1-r_1}{\sigma_1}\right)^2 t\}, \\ \text{if } 0 \le t \le \tau, \\ \exp\{-\frac{\mu_1-r_1}{\sigma_1}W_\tau - \frac{1}{2}\left(\frac{\mu_1-r_1}{\sigma_1}\right)^2 \tau - \frac{\mu_2-r}{\sigma_2}(W_t - W_\tau) - \frac{1}{2}\left(\frac{\mu_2-r}{\sigma_2}\right)^2 (t-\tau)\}, \\ \text{if } \tau < t \le T. \end{cases} \tag{4.28}$$

*Under $P^*$ we have the same equivalence in law as in the first case (4.12)*

$$S_t = \begin{cases} S_0 e^{\sigma_1 W_t^* + (r_1 - \frac{\sigma_1^2}{2} t)}, \ \text{if } 0 \le t \le \tau, \\ S_\tau e^{\sigma_2 (W_t^* - W_\tau^*) + (r_2 - \frac{\sigma_2^2}{2}(t-\tau))}, \ \text{if } \tau < t \le T; \end{cases} \tag{4.29}$$

*where $W^*$ is Wiener process w.r.t $P^*$, but in the case of random time of the parameters changing*

$$C(S_T, K) = S_0 E[N(d_3(\tau))] - KE[e^{-r_1\tau - r_2(T-\tau)} N(d_4(\tau))],$$

$$P(S_T, K) = KE[e^{-r_1\tau - r_2(T-\tau)} N(-d_4(\tau))] - S_0 E[N(-d_3(\tau))],$$

*where*

$$d_3 = \frac{ln\frac{S_0}{K} + (r + \frac{\sigma_1^2}{2})\tau + (r + \frac{\sigma_2^2}{2})(T-\tau)}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}},$$

$$d_4 = \frac{ln\frac{S_0}{K} + (r - \frac{\sigma_1^2}{2})\tau + (r - \frac{\sigma_2^2}{2})(T-\tau))}{\sqrt{\sigma_1^2\tau + \sigma_2^2(T-\tau)}}$$

*and $N(\cdot)$ is the distribution function of the standard normal variable, $E$ is an expectation with respect to the measure $P$.*

**Proof:** Let us define the process $Z$ corresponding to the theorem conditions.

Now we prove that the process $Z$ is a martingale. Consider two possible cases: $t \leq \tau$ and $t > \tau$.

Let $t \leq \tau$, in this case the process $Z$ has a representation

$$Z_t = \exp\{-\frac{\mu_1 - r_1}{\sigma_1} W_t - \frac{1}{2}\left(\frac{\mu_1 - r_1}{\sigma_1}\right)^2 t\}. \quad (4.30)$$

For a simplicity we will denote

$$a_1 = -\frac{\mu_1 - r_1}{\sigma_1},\ a_2 = -\frac{\mu_2 - r_2}{\sigma_2}, \quad (4.31)$$

then using the results from the previous theorem an expectation is given by

$$EZ_t = E[\exp\{a_1 W_t - \frac{1}{2}a_1^2 t\}] = \exp\{-\frac{1}{2}a_1^2 t\}E[\exp\{-a_1 W_t\}] = 1. \quad (4.32)$$

In the case when $t > \tau$ we have

$$EZ_t = E[\exp\{-a_1 W_\tau - a_1^2\tau - a_2(W_t - W_\tau) - a_2^2(t-\tau)\}]$$

$$= E[\exp\{-a_1^2\tau - a_2^2(t-\tau)\}E[exp\{-a_2(W_t - W_\tau) - a_1 W_\tau\}|\tau]]. \quad (4.33)$$

Using the independence of increments

$$EZ_t = E[\exp\{-a_1^2\tau - a_2^2(t-\tau)\}E[exp\{-a_2(W_t - W_\tau)\}|\tau]E[exp\{-a_1 W_\tau\}|\tau]]$$

hence using the results from the first case we obtain next

$$EZ_t = E[\exp\{-a_1^2\tau - a_2^2(t-\tau)\}\exp\{a_1^2\tau + a_2^2(t-\tau)\}] = 1$$

Therefore we can define the Radon-Nikodim derivative using the process $Z$.

On the second step, we should prove the expression (4.29) of the process $S$ with respect to the probability measure $P^*$. The proof of this step is identical to the proof in previous Theorem 8.

Now we calculate an option price for the B-S model with changing parameters. An option price for a Call option is given by

$$C(S_T, K) = E^*\left(\frac{(S_T - K)^+}{B_T}\right), \quad (4.34)$$

where $E^*$ is an expectation w.r.t. the martingale measure ($P^*$) (4.28) and it can be rewrited in the following way

$$E^*\left(\frac{(S_T - K)^+}{B_T}\right) = E^*\left(\frac{(S_T - K)}{B_T} I_{\{S_T > K\}}\right)$$

$$= E^* \left( \frac{S_T}{B_T} I_{\{S_T>K\}} \right) - KE^* \left( \frac{1}{B_T} I_{\{S_T>K\}} \right) = E_1 - KE_2. \quad (4.35)$$

Firstly, we find the expectation $E_2$, recall that we take $B_0 = 1$ in all our calculations

$$E_2 = E^*(\exp\{-r_1\tau - r_2(T-\tau)\} I_{S_T>K})$$

$$= E^* \left( \exp\{-r_1\tau - r_2(T-\tau)\} E^* \left[ I_{\left\{S_0 \exp\{d(\tau) + \sigma_1 W_\tau^* + \sigma_2(W_T^* - W_\tau^*)\} > K\right\}} | \tau \right] \right)$$

where $d(\tau) = (r_1 - \frac{\sigma_1^2}{2})\tau + (r_2 - \frac{\sigma_2^2}{2})(T-\tau)$.

The inside conditional expectation can be computed identically to $E_2$ in the previous Theorem 8, hence we obtain

$$E_2 = E^*[\exp\{-r_1\tau - r_2(T-\tau)\} N(d_4(\tau))] = E^*[f(\tau)]. \quad (4.36)$$

We move from an expectation with respect to the martingale measure $P^*$ to an expectation with respect to the physical measure $P$ using the density process $Z$. This is important, because the distribution function of $\tau$ can be estimated on real data and it will be the distribution of $\tau$ w.r.t. the measure $P$.

$$E_2 = E[\exp\{a_1 W_\tau + a_2(W_T - W_\tau)\} - \frac{a_1^2}{2}\tau - \frac{a_2^2}{2}(T-\tau)\} f(\tau)]$$

$$= E[\exp\{-\frac{a_1^2}{2}\tau - \frac{a_2^2}{2}(T-\tau)\} E[\exp\{a_1 W_\tau + a_2(W_T - W_\tau)\} | \tau] f(\tau)].$$

Because of the independence of $W_T - W_\tau$ and $W_\tau$

$$E_2 = E[\exp\{-\frac{a_1^2}{2}\tau - \frac{a_2^2}{2}(T-\tau)\} f(\tau) E[\exp\{a_1 W_\tau\} | \tau] E[\exp\{a_2(W_T - W_\tau)\}].$$

Using the previous results for the conditional expectation we see that

$$E_2 = E[\exp\{-r_1\tau - r_2(T-\tau)\} N(d_4(\tau))].$$

Now we calculate an expectation $E_1$ (4.35). It is defined as

$$E_1 = E^* \left[ \frac{S_T}{B_T} I_{\{S_T>K\}} \right]$$

$$= E^*[\exp\{-r_1\tau - r_2(T-\tau)\} E^*[S_T I_{\{S_T>K\}} | \tau]], \quad (4.37)$$

where the inside expectation can be computed identically to the computation of $E_1$ in the previous Theorem 8, hence

$$E_1 = S_0 E^*[N(d_3(\tau))],$$

or after moving to the expectation with respect to the measure $P$

$$E_1 = S_0 E[\exp\{-\frac{a_1^2}{2}\tau - \frac{a_2^2}{2}(T-\tau)\} E[\exp\{a_1 W_\tau + a_2(W_T - W_\tau)\}|\tau] N(d_3(\tau))]$$

$$= E[\exp\{-\frac{a_1^2}{2}\tau - \frac{a_2^2}{2}(T-\tau)\} E[\exp\{a_1 W_\tau\}|\tau] E[\exp\{a_2(W_T - W_\tau)\}] N(d_3(\tau))],$$

or

$$E_1 = S_0 E[N(d_3(\tau))].$$

Finally we have

$$C(S_T, K) = S_0 E[N(d_3(\tau))] - K E[e^{-r_1\tau - r_2(T-\tau)} N(d_4(\tau))]. \qquad (4.38)$$

It is easy to obtain an analogous formula for the price of a Put option. An option price for a Put option is given by

$$P(S_T, K) = E^* \left( \frac{(K - S_T)^+}{B_T} \right), \qquad (4.39)$$

where $E^*$ is an expectation w.r.t. the martingale measure ($P^*$) (4.28) and it can be rewrited in the following way

$$E^* \left( \frac{(K - S_T)^+}{B_T} \right) = E^* \left( \frac{(K - S_T)}{B_T} I_{\{S_T < K\}} \right)$$

$$= K E^* \left( \frac{1}{B_T}(1 - I_{\{S_T > K\}}) \right) - E^* \left( \frac{S_T}{B_T}(1 - I_{\{S_T > K\}}) \right)$$

$$= K E^* \left( \frac{1}{B_T} \right) - K E^* \left( \frac{1}{B_T} I_{\{S_T > K\}} \right) - E^* \left( \frac{S_T}{B_T} \right) + E^* \left( \frac{S_T}{B_T} I_{\{S_T > K\}} \right).$$

Clearly, that $E^*\left(\frac{1}{B_T}\right) = E\left(\frac{1}{B_T}\right)$ and that $E^*\left(\frac{S_T}{B_T}\right) = S_0$ (by using the definition of a martingale measure), then

$$P(S_T, K) = K E\left[\frac{1}{B_T}\right] - K E_2 - S_0 + E_1$$

$$= K\left( E\left[\frac{1}{B_T}\right] - E\left[\frac{1}{B_T} N(d_4(\tau))\right]\right) - S_0\left(1 - E[N(d_3(\tau))]\right)$$

$$= K E\left[\frac{1}{B_T} N(-d_4(\tau))\right] - S_0 E[N(-d_3(\tau))].$$

Finally we have

$$P(S_T, K) = K E[e^{-r_1\tau - r_2(T-\tau)} N(-d_4(\tau))] - S_0 E[N(-d_3(\tau))]. \qquad (4.40)$$

## The computation of $E[N(d_3)]$, $E[N(d_4)]$.

To compute expectations $E[N(d_3)]$, $E[N(d_4)]$ we need the distribution of $\tau$. The value $\tau$ is the time of parameters change. In the real world it means some events on the market which is lead to the change of the volatility and the drift (for example, some crisis or in contrary some good reports of the companies or some another events). To determ the $\tau$ distribution we need to investigate the statistics of such changes and approximate it's distribution.

Then we calculate expectations in the formula of a price (4.38, 4.40) numerically, or we can use the following approximation of $N$ ([1])

$$N(x) = \begin{cases} 1 - N'(x)\left(a_1 \dfrac{1}{1+bx} + a_2 \dfrac{1}{(1+bx)^2} + \cdots + a_5 \dfrac{1}{(1+bx)^5}\right), \\ 1 - N(-x) \end{cases}$$

where $b$, $a_1$, $a_2$, $a_3$, $a_4$, $a_5$ are some known constants and

$$N'(x) = \frac{1}{\sqrt{2\pi}} e^{-\dfrac{x^2}{2}}.$$

## 4.3 The numerical results

Here for computations of call option prices was used a data from the Swedish market for stock prices, option prices and the volatility for H&M and Electrolux companies (for details see Appendix A). On plots (see Figure 4.1 - Figure 4.2) it is visible that the curve obtained by the Black-Scholes model with the deterministic time of changes (BSDTCM) of parameters situated closer to the real Bid-Ask spread than the curve obtained by the standard Black-Scholes model (BSM). This means that by the assumption of the right prediction of the $\sigma$ behavior we obtain prices, which are better matched the real prices on the market.

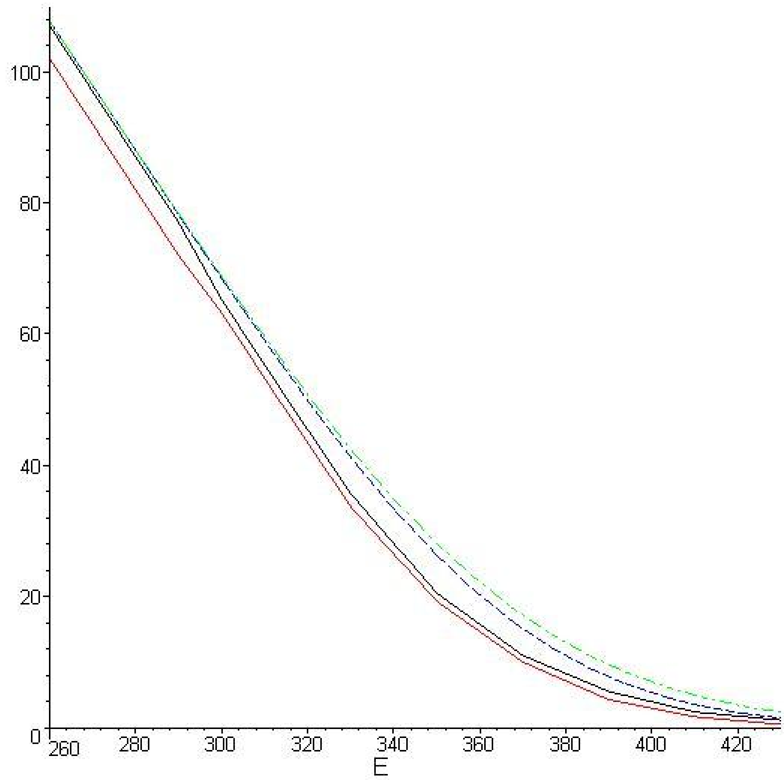


Figure 4.1: Strike price $E$ (horizontal axis) - Call price $C$ (vertical axis) of the H&M company on the 2nd of May ($T = 50/262$ expiration date is in July'08, $\tau = 9/262$ parameters changed at the 10th of May, $r_1 = r_2 = 0.0465, t = 0, S_0 = 365.5, \sigma_1 = 0.278, \sigma_2 = 0.238$). Bid - red line, Ask - black line, BS - green line, BSDCV - blue line.

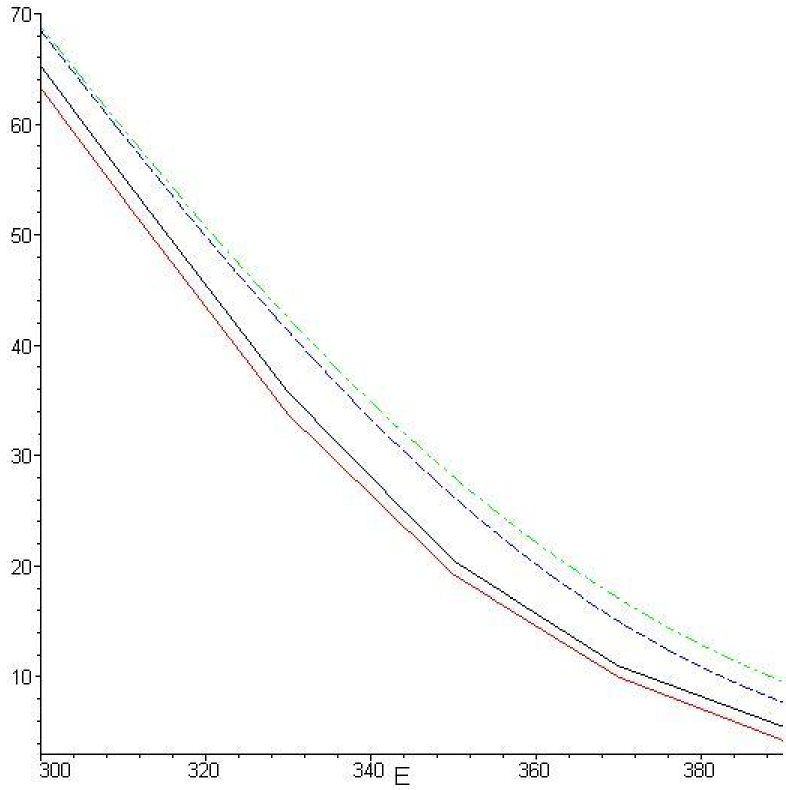


Figure 4.2: Strike price $300 \leq E \leq 390$ (horizontal axis) - Call price $C$ (vertical axis) of the H&M company on the 2nd of May ($T = 50/262, \tau = 9/262, r_1 = r_2 = 0.0465, t = 0, S_0 = 365.5, \sigma_1 = 0.278, \sigma_2 = 0.238$). Bid - red line, Ask - black line, BS - green line, BSDCV - blue line.

In following Figure 4.3 few 3D plots are presented, they shows changes of (BSM) price (on the left picture) and (BSDTCM) price (on the right picture) against $S/K$ ratio and the time to maturity for the Electrolux company. In Figure 4.4 the corresponding diference between BSM and BSDTCM prices are presented .

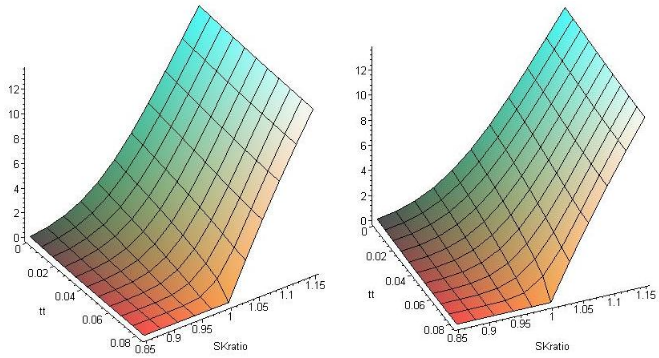


Figure 4.3: $S/K$ ratio - Time to maturity for the Black-Scholes model (the left picture) and for the Black-Scholes model with changing parameters (the right picture) for the Electrolux company (June'08).

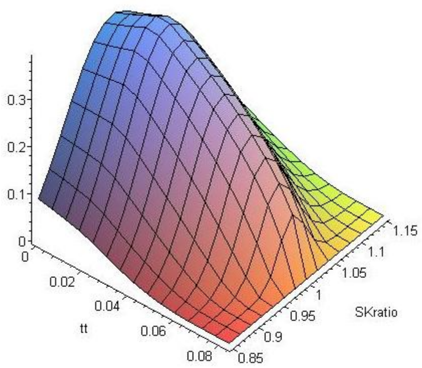


Figure 4.4: The difference Black-Scholes prices vs. the Black-Scholes prices with DCV for the Electrolux company (June'08).

# Chapter 5

# Numerical calculations for a European Call/Put, a Double Barrier Put and an Up-and-out Call option for the CGMY model

## 5.1 The partial Integro-Differential Equation (PIDE)

In this chapter 5 I investigate a model which is based on a Levy process to calculate option prices. In such type of models the price process has the form

$$S_t = S_0 e^{rt+X_t}, \tag{5.1}$$

where $X$ is a Levy process with the characteristic triplet $\langle \sigma, \gamma, \nu \rangle$. We define the infinitesimal operator of the process $X$ as

$$L^X f = \frac{\sigma^2}{2}\frac{\partial^2 f}{\partial x^2} + \gamma \frac{\partial f}{\partial x} + \int_{-\infty}^{\infty} (f(x+y) - f(x) - yI_{\{|y|\leq 1\}}\frac{\partial f}{\partial x}(x))\nu(dy). \tag{5.2}$$

From the assumption of the risk-neutrality of the price process $S$ follows that $\tilde{S}_t = S_t e^{-rt} = exp(X_t)$ should be a martingale. The Levy process $X$ is defined on the filtered probability space $((\Omega, \mathcal{F}, ((\mathcal{F}_t)_{t\geq 0}), P))$. If we assume that the measure $P$ is a martingale measure then the triplet of $X$ should admit following conditions

$$\int_{|y|>1} \nu(dy)e^y < \infty, \tag{5.3}$$

$$\gamma = \gamma(\sigma, \nu) = -\frac{\sigma^2}{2} - \int (e^y - 1 - yI_{\{|y|\leq 1\}})\nu(dy). \quad (5.4)$$

Now we define the Levy process $Y$, $Y_t = rt + X_t$ with the infinetisimal operator

$$Lf = L^X f + r\frac{\partial f}{\partial x}$$

$$= \frac{\sigma^2}{2}\frac{\partial^2 f}{\partial x^2} + \left(r - \frac{\sigma^2}{2}\right)\frac{\partial f}{\partial x} + \int_{-\infty}^{\infty} (f(x+y) - f(x) - (e^y - 1)\frac{\partial f}{\partial x}(x))\nu(dy). \quad (5.5)$$

Now we start the option pricing. The value of an option under a risk-neutral measure is defined by

$$C_t = E[e^{-r(T-t)}H(S_T)|\mathcal{F}_t], \quad (5.6)$$

where $H(S_T)$ is a payoff function. If we use the Markov property of a Levy processes, then we can rewrite the last equation (5.6) in the form

$$C_t = E[e^{-r(T-t)}H(S_T)|S_t = S]. \quad (5.7)$$

After substitutions $\tau = T - t$, $x = ln(S/S_0)$, $h(x) = H(S_0 e^x)$ and $u(\tau, x) = e^{rT}C(T - \tau, S_0 e^x)$ we obtain the inverse Kolmogorov equation

$$u(\tau, x) = E[h(x + Y_\tau)]. \quad (5.8)$$

By applying Ito's formula to $u(t, X_t)$ in the area $(0, T] \times R$ one can show [8] that it is a classical solution of the Cauchy problem

$$\frac{\partial u}{\partial \tau} = Lu, u(0, x) = h(x), \ x \in R. \quad (5.9)$$

The question of the existence and uniqueness of a solution of the Cauchy problem (5.9) have been studied in [13] in the case a non degenerate defusion component. In the case of pure jump models the uniqueness and existence were studied in [14], [15] in terms of a viscosity solution.

Results from [13] shows that if a payoff function verify Lipschitz conditions then prices of the European and barrier options can be obtain in terms of a viscosity solution of the Cauchy problem (5.9) defined on some domain.

Now we consider the Finite Difference approach for solving PIDE.

## 5.2 A Finite Difference method for solving PIDE (5.9)

**Truncation of a domain.**

To solve PIDE (5.9) numerically we localize the variables and the integral term on the bounded domain. Firstly, we truncate the domain on $x \in (-A, A)$. But an integral part is not local and we should know value $u$ not only in any point $x \in (-A, A)$ but as well on whole the set $\{x + y : y \in supp(\nu)\}$. Let $u(\tau, x) = g(\tau, x)$, if $x \notin (-A, A)$. We take two cases: $g = 0$, which is natural, for example, for a barrier option and $g(\tau, x) = h(x)$, which is asymptotically close to the solution at infinity.

Note that an error exponentially growing with the domain decreases [4].

**Truncation of the integral.**

For the computation of an integral part we need to truncate a region of the integration. For the Levy process the truncation of a region of the integration means that we truncate large jumps. In other words we will define a new process $\tilde{X}$ with the triplet $\langle \tilde{\gamma}, \sigma, \nu I_{\{x \in [B_l, B_r]\}} \rangle$, where $\tilde{\gamma}$ is such that the martingale property of the price process holds

$$\tilde{\gamma} = -\frac{\sigma^2}{2} - \int_{B_l}^{B_r} (e^y - 1 - yI_{\{|y| \leq 1\}})\nu(dy). \tag{5.10}$$

In paper [4] you find an estimation of the truncation error, we mentione that the error is exponentially growing if the region decreases.

Now we consider the Finite Difference approach for a numerical solution of PIDE.

**An explicit-implicit scheme. A finite intensity case.**

Let us recall our Cauchy problem

$$\frac{\partial u}{\partial \tau} = Lu, \text{ on } (0, T]x(-A, A), \tag{5.11}$$

$$u(\tau, x) = g(\tau, x), \text{ when } x = (-\infty, A) \cup (A, \infty), \tag{5.12}$$

$$u(0, x) = h(x), \text{ when x } \in (-A, A), \tag{5.13}$$

where $L$ was define above in (5.5).

We split an operator $L$ into the integral and differential parts

$$\frac{\partial u}{\partial \tau} = Du + Iu. \tag{5.14}$$

By $D_\Delta$ we denote a finite-difference approximation of an operator $D$ and $I_\Delta$ is a trapezoidal quadrature approximation of $I$. For the approximation of $u$ time derivative we use following finite-difference scheme

$$\frac{u^{n+1} - u^n}{\Delta t} = D_\Delta u^{n+1} + I_\Delta u^n. \tag{5.15}$$

Note that we have the integral part $I_\Delta u^n$ in an explicit time stepping, because in this case we don't need to inverse a non sparse matrix $I_\Delta$ and it does not have any influence the stability.

We consider models with a finite intensive Levy measure $\nu(dx) = \lambda < \infty$. Then operators $D$ and $I$ can be written as

$$Du = \frac{\sigma^2}{2}\frac{\partial^2 u}{\partial x^2} - (\frac{\sigma^2}{2} - r + \alpha)\frac{\partial u}{\partial x} - \lambda u, \tag{5.16}$$

$$Iu = \int_{B_l}^{B_r} u(\tau, x + y)\nu(dy), \tag{5.17}$$

where $\alpha = \int_{B_l}^{B_r} (e^y - 1)\nu(du)$.

We define a grid on $[0, T] \times [-A, A]$: the time step size $\Delta t = T/M$: $\tau_n = n\Delta t$,$n = 0 \ldots M$; the space step size on $(-A, A)$ is $\Delta x = 2A/N$: $x_i = -A + i\Delta x$,$i = 0 \ldots N$.

We use following approximations

$$\int_{B_l}^{B_r} u(\tau, x_i + y)\nu(dx) \approx \sum_{j=K_l}^{K_r} \nu_j u_{i+j}, \tag{5.18}$$

$$\lambda \approx \widetilde{\lambda} = \sum_{j=K_l}^{K_r} \nu_j, \tag{5.19}$$

$$\alpha \approx \widetilde{\alpha} = \sum_{j=K_l}^{K_r} (e^{y_j} - 1)\nu_j, \tag{5.20}$$

where $\nu_j = \int_{(j-1/2)\Delta x}^{(j+1/2)\Delta x} \nu(dy)$.

A derivative in the space direction we will approximate using the following discretization

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_i \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x)^2},$$

$$\left(\frac{\partial u}{\partial x}\right)_i \approx \begin{cases} \frac{u_{i+1}-u_i}{\Delta x}, \ if \ \sigma^2/2 - t + \hat{\alpha} < 0 \\ \frac{u_i - u_{i-1}}{\Delta x}, \ if \ \sigma^2/2 - t + \hat{\alpha} \geq 0 \end{cases}$$

The choice between two cases for approximations of the first-order derivative is based on stability conditions.

The initial conditions are

$$u_i^0 = h(x_i), \ i \in \{0, \ldots, N\}. \tag{5.21}$$

The boundary conditions for $n = 0, \ldots, M-1$ are

$$u_i^0 = g(0, x_i), \ if \ i \notin \{0, \ldots, N\}. \tag{5.22}$$

**The Explicit-implicit scheme: the Infinite intensity case.**

For the approximation of PIDE in a case when $\nu(R) = +\infty$ we have to improve the method described above. The idea is to substitute small jumps (that are smaller then $\epsilon$) for the Brownian motion. For this purpose we change characteristic triplet $(\gamma, \sigma, \nu)$ of our Levy process $X$ from $Y_t = rt + X_t$ (recall that $S_t = S_0 exp\{Y_t\}$) to $(\gamma(\epsilon), \sqrt{\sigma^2 + \sigma^2(\epsilon)}, \nu I_{|x| \geq \epsilon})$, where

$$\sigma^2(\epsilon) = \int_{-\epsilon}^{\epsilon} y^2 \nu(dy),$$

by the martingale conditions

$$\gamma(\epsilon) = -\frac{\sigma^2 + \sigma^2(\epsilon)}{2} - \int_{|y| \geq \epsilon} (e^y - 1 - yI_{|y| \leq})\nu(dy).$$

After such manipulations $Y_\tau^\epsilon$ has jumps of a finite intensity and we can consider the following Cauchy problem

$$\frac{\partial u^\epsilon}{\partial \tau} = L^\epsilon u^\epsilon, \text{ on } (0, T] \times R, \tag{5.23}$$

$$u^\epsilon(0, x) = h(x), \text{ when x } \in R, \tag{5.24}$$

where

$$Lf = \frac{\sigma^2 + \sigma^2(\epsilon)}{2}\frac{\partial^2 f}{\partial x^2} + (r - \frac{\sigma^2 + \sigma^2(\epsilon)}{2} - \alpha(\epsilon))\frac{\partial f}{\partial x} - \lambda(\epsilon) f(x) + \int_{|y|\geq\epsilon} f(x+y)\nu(dy)$$

and $\alpha(\epsilon) = \int_{|y|\geq\epsilon}(e^y - 1)\nu(dy)$, $\lambda(\epsilon) = \int_{|y|\geq\epsilon}\nu(dy)$ and $h = 2A/N$.

We can use the method that was described for the finite intensity case.

## 5.3 The method of Moments for the estimation of the Levy measure parameters of the CGMY process

For the numerical calculations of the option prices based on the finite-difference scheme given above we use, so-called, an extended CGMY process. An extended CGMY process is a Levy process with the triplet of characteristics $(\gamma, \sigma, \nu)$, where $\sigma$ is not equal to zero (see [3]) and the Levy measure has the following form

$$\nu(x) = C\left(exp(-G|x|)I_{x<0} + exp(-M|x|)I_{x>0}\right)/|x|^{1+Y}, \tag{5.25}$$

where $C > 0$, $G, M \geq 0$ and $Y < 2$.

If we want to use this process we should be able to estimate the five parameters - $\gamma$, $C$, $G$, $M$ and $Y$. For this estimation we use the method of moments (MM) (see [11]).

There are exist two kinds of the method of moments - the Classical and the Generalized method of moments (or the Characteristic function method). The first one can be apply if a random variable has moments of any degree. The second one can be apply if a random variable do not possess moments. Now we take a short explanation of MM.

1. **Moments and sample moments**. If $Z$ is some random variable then moments are defined by

$$m_k = E[Z^k], \quad k = 1, 2, ...n.$$

   Assume that we have the sample data $(z_i)_{i=1}^n$ of this r.v. then corresponding sample moments are given by

$$\tilde{m}_k = \frac{1}{n}\sum_{i=1}^{N} z_i^k.$$

2. **The classical moment equation**. The main idea of the MM is to solve following system of equations

$$m_k = \tilde{m}_k, \;\; k = 1, ..., N,$$

where $N$ is a number of parameters which we have to estimate. We can transform this equation using properties of the characteristic functions. A characteristic function is defined by

$$A(u) = E[e^{iuZ}] = exp(\tilde{A}(u)).$$

If we denote by

$$A^{(k)} = \frac{d^k A(u)}{du^k}, \;\; \tilde{A}^{(k)} = \frac{d^k \tilde{A}(u)}{du^k}$$

we rewrite moment equations using well-known formulas

$$m_k = E[Z^k] = \frac{1}{i^k} E[(iZ)^k] = \frac{1}{i^k} A^{(k)}(0)$$

into our moment equations

$$\tilde{m_k} = \frac{1}{i^k} A^{(k)}(0), k = 1, 2, ..., N.$$

3. **The generalized moment equation**. In the case when we can't use the classical moment equation we can use the generalized moment equation

$$A(u) = lim\tilde{A}(u),$$

where $\tilde{A}(u)$ is a sample characteristic function which is defined by

$$\tilde{A}(u) = \frac{1}{n} \sum_{i=1}^{n} e^{iuz_i}.$$

Now we apply the Method of Moments for the estimation of parameters of the extended CGMY process using the data on the behavior of the stock price of the H&M company on the Swedish market in 1200 point (from the 1st of January 2004 to the 18th of October 2008), see Figure 5.1.

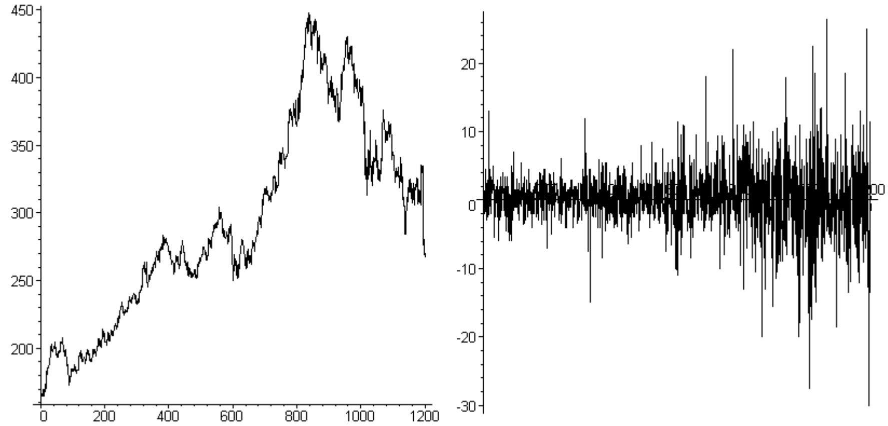


Figure 5.1: The stock price behavior and logreturns for the H&M company.

We skip some details which are not concern to our case (for details see [11]) and give formulas for the estimators $\tilde{\gamma}$, $\tilde{C}$, $\tilde{G}$, $\tilde{M}$ and $\tilde{Y}$, which are obtained by the Method of Moments.

1. **The estimation of** $Y$. The estimator $\tilde{Y}$ of $Y$ is given by
$$\tilde{Y} = \frac{ln(|Re[ln\tilde{A}(u_1)]|)}{lnu_1},$$
where $u_1$ is a large number.

2. **The estimation of** $\gamma$. It is well-known that $\gamma = m_1$, then the estimator of $\gamma$ is given by
$$\tilde{\gamma} = \tilde{m}_1.$$

3. **The estimation of** $C$. The estimator of $C$ is obtained as a solution of the following equation
$$\frac{Re[ln(\tilde{A}(u_2))]}{u_2^{\tilde{Y}}} = 2\tilde{C}\Gamma(-\tilde{Y})cos(\frac{\sigma^2}{2}\tilde{Y}),$$
where $u_2$ is a large number.

4. **The estimation of $G$ and $M$.** For the estimation of $G$ and $M$ we should solve the system of the two moment equations

$$\tilde{C}\tilde{\Gamma}(-\tilde{Y})\tilde{Y}(\tilde{Y}-1)[\tilde{G}^{(\tilde{Y}-2)}+\tilde{M}^{(\tilde{Y}-2)}]=h_2, \tag{5.26}$$

$$\tilde{C}\tilde{\Gamma}(-\tilde{Y})\tilde{Y}(\tilde{Y}-1)(\tilde{Y}-2)[\tilde{M}^{(\tilde{Y}-3)}-\tilde{G}^{(\tilde{Y}-3)}]=h_3, \tag{5.27}$$

where $h_2=\tilde{m}_2-\tilde{m}_1^2$, $h_3:=\tilde{m}_3-3\tilde{m}_2\tilde{m}_1+2\tilde{m}_1^3$. For the convergence of the Newton's method was found using the program software in Maple (see Appendix B) the approximation of the solution as the cross-point of two implicit plots (see Figure 5.2).

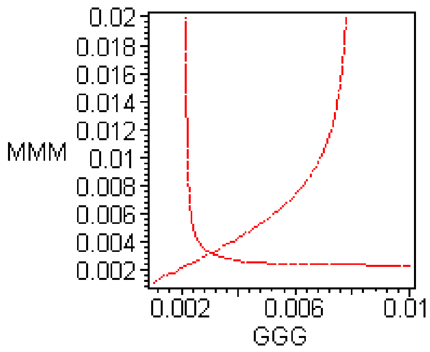


Figure 5.2: The graphical solution of the system of equations (5.26)-(5.27) w.r.t. variables $G$ and $M$.

## 5.4 The numerical results

For the numerical calculations using finite-difference scheme described above have been developed the program software in Delphi 7 (See Appendix C). Delphi have been choosed for our purposes because this software environment allow to create quick programs with very comfortable and flexible interface. Besides Delphi it is more convenient to write large and complicate programs, because the code of the program can be better structurize than in such application mathematical packages as Mathematica or Matlab.

To describe my results I start from the program software description. A general view of the program is presented in Figure 5.3.

Structurally the main window consists of the six blocks with the parameters, the two buttons for the computation, the one block with results and the two plot windows.

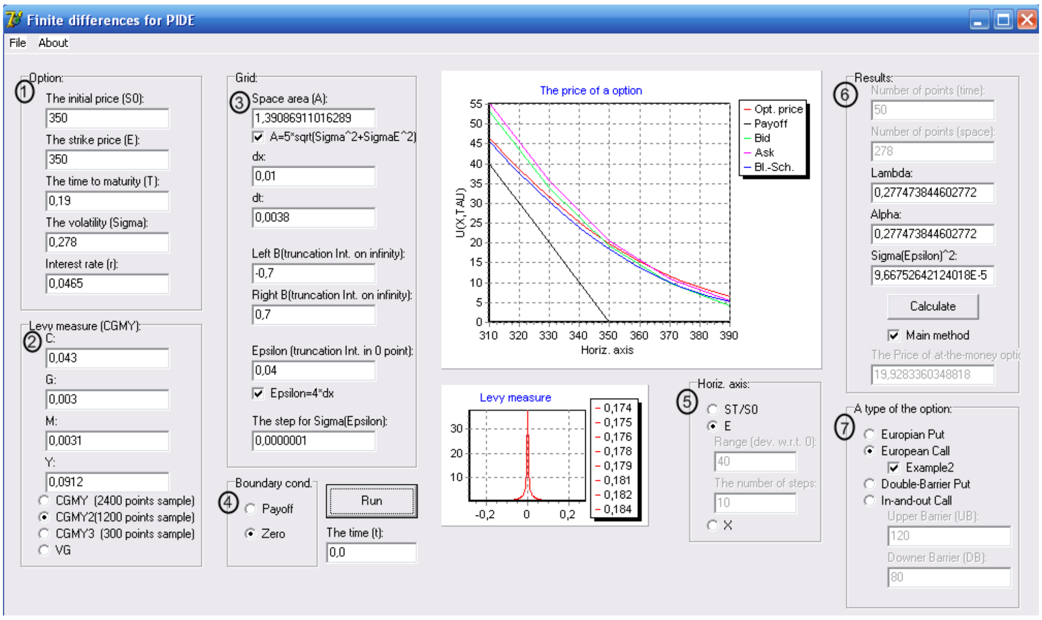


Figure 5.3: A general view of the program for the option price calculations by the PIDE method.

- The first block is the block of the option parameters, i.e., an initial price of an underlying asset ($S_0$), a strike price ($E$), a time to maturity ($T$), a volatility ($\sigma$) and an interest rate ($r$).

- The next four block contains parameters of our finite-difference scheme and parameters of our model. The second block include the "radio button" for choosing of the parameters set for the Levy measure (can be choosed CGMY with the 2400 points sample, CGMY with the 1200 points sample and CGMY with the 300 points sample) and four fields with the corresponding parameters.

- The third block include parameters of the scheme, i.e., a space area ($A$), a step size ($dt$ and $dx$), the left and right borders of the integration domain of the integral term in PIDE ($B_l$ and $B_r$), the truncation area of integrals around zero ($\epsilon$) and a step for calculation $\sigma(\epsilon)$. Also it include two the check boxes. If the first "check box" is checked then a space area ($A$) is automatically compute in according to the formula $A = 5\sqrt{\sigma^2 + \sigma^2(\epsilon)}$. If the second check box is checked then the aplication automatically support the ratio between $\epsilon$ and $dx$ equals four.

- In the fourth block we can choose a type of the boundary conditions: zero or payoff functions.

- In the fifth block the control parameters of the main plot are situated. On the horizontal axes we choose the ratio $S_T/S_0$, the strike price $E$ or the variable $x$.

- In the six block we can choose a type of the option (European Call/Put, Double-barrier Put and In-and-out Call).

To compute the option price with parameters $\lambda$, $\alpha$ and $\sigma(\epsilon)$, please, press the button "Run". To compute only the parameters $\lambda$, $\alpha$ and $\sigma(\epsilon)$, please, press the button "Calculate".

The last, seventh block, represent results such as the price of the at-the-money option, parameters of the model $\lambda$, $\alpha$, $\sigma(\epsilon)$, the number of steps in time and space areas.

On the main plot Figure 5.3 the result of computations given as curves of: the option price according to our model, the Black-Scholes price, the payoff and the Bid-Ask curves (for the data on the option price and the behavior of the stock price of the H&M company).

Now we proceed to numerical results. Using the finite-difference scheme described above we have got prices for four types of options: the vanilla Call/Put, the double-barrier Put and the up-and-out Call. All computations made for the three sets of parameters of the Levy measure: for the CGMY with 2400 sample points (CGMY1), for the CGMY with 1200 sample points (CGMY2) and for the CGMY with 300 sample points (CGMY3). Recall that in the CGMY model the Levy measure is defined as

$$\nu(x) = C\left(exp(-G|x|)I_{x<0} + exp(-M|x|)I_{x>0}\right)/|x|^{1+Y}. \tag{5.28}$$

In computations using the formula (5.28) following parameters are used: $C = 0.0502$, $G = 0.0029$, $M = 0.0031$, $Y = 0.1079$ (CGMY1), $C = 0.043$, $G = 0.003$, $M = 0.0031$, $Y = 0.0912$ (CGMY2) and $C = 0.0422$, $G = 0.0128$, $M = 0.0131$, $Y = 0.0895$ (CGMY3).

For the computation of barrier option prices we make a little change in our initial-boundary problem (5.11-5.13). For instance, an Up-and-out Call ([6]) has two parameters like vanilla Call: the time to maturity ($T$), the strike price ($E$) and an additional parameter the value of the upper boundary ($U$). While the price of an underlying asset $S_t$ is smaller than $U$, then the payoff of the Up-and-out Call is the same as the payoff of the vanilla Call, but if the price $S_t$ become above or equal to the upper barrier $U$ the payoff of the Up-and-out Call option is equal to zero. The payoff function in the moment $T$ is given by

$$H_T = (S_T - E)^+ I_{\{T<\theta\}},$$

where $\theta = inf\{t \geq 0 | S_t \geq U\}$ is the first moment at which the barrier is crossed. The price of the barrier option is defined by

$$C_b(t,S) = e^{-r(T-t)} E[H(Se^{Y_{T-t}}) I_{T<\theta_t}],$$

where $\theta_t = inf s \geq t | Se^{Y_{s-t}} \geq U$ is the first time after $t$ at which the barrier $U$ is crossed. This means that for the Up-and-out Call we obtain that $u_b(\tau, x) = e^{r\tau} C_b(T-\tau, S_0 e^x)$ is a solution of the following initial-boundary value problem

$$\frac{\partial u}{\partial \tau} = Lu, \quad on(0,T] \times (-\infty, log(u/S_0)),$$

$$u(0,x) = h(x), \quad x < log(U/S_0),$$

$$u(\tau, x) = 0, \quad x \geq log(U/S_0)$$

and for the double-barrier option of the following problem

$$\frac{\partial u}{\partial \tau} = Lu, \quad on(0,T] \times (-\infty, log(u/S_0)),$$

$$u(0,x) = h(x), \quad x \in (log(D/S_0), log(U/S_0)),$$

$$u(\tau, x) = 0, \quad x \leq log(D/S_0) \; and \; x \geq log(U/S_0),$$

where $D$ is the lower barrier.

To check the quality of the CGMY model we computed prices of a Call option for the H&M company using the Black-Scholes model (we used the implied volatility from Table 5.1) and the CGMY model. Then we calculate the fitting error using the data at the 2nd of May'08 about Call option prices for the H&M company from the Table 5.1 below.

We define the fitting error of the model by

$$\epsilon = \frac{1}{L} \sum_l \frac{|C_l - \eta_l|}{\eta_l}, \tag{5.29}$$

where $C_l$ is a theoretical price and $\eta_l$ is a real price (we took the average price between the bid and ask prices from the Table 5.1 above). The results of the fitting error computation is shown in Table 5.2.

From this analysis it is obviosly that the fitting error of the CGMY model is smaller than the fitting error of the Black-Sholes model, especially for the case CGMY3. In Table 5.3 we present results of the fitting analysis for the strike prices which are close to the spot price $S_0$ (for $E$ equal to 310, 330, 350, 370, 390).

Evidently, that the fitting error of the CGMY model is better than the Black-Scholes fitting error by one order of magnitude. Note that for the

| Strike | Bid | Ask | BS implied volatility for Bid | BS implied volatility for Ask |
|---|---|---|---|---|
| 260 | 102 | 107 | 0.706 | 0.841 |
| 270 | 92 | 97 | 0.643 | 0.771 |
| 280 | 82 | 87 | 0.569 | 0.703 |
| 290 | 72 | 77 | 0.52 | 0.635 |
| 300 | 63.25 | 65.25 | 0.489 | 0.532 |
| 310 | 53.25 | 55.25 | 0.428 | 0.469 |
| 330 | 33.75 | 35.75 | 0.308 | 0.351 |
| 350 | 19.25 | 20.5 | 0.268 | 0.288 |
| 370 | 10 | 11 | 0.254 | 0.269 |
| 390 | 4.25 | 5.5 | 0.24 | 0.261 |
| 410 | 1.65 | 2.4 | 0.232 | 0.261 |
| 430 | 0.5 | 1.15 | 0.225 | 0.268 |

Table 5.1: The real data set for the option prices for the H&M company at the 2nd of May'08.

computation of an option price for each strike price we use the corresponding BS implied volatility from the Table 5.1.

Results of the computation of the at-the-money option prices (using the numerical method considered above for the three different Levy measures CGMY1, CGMY2 and CGMY3) are collected in the Table 5.4. The computations are done for the implied volatility, which is equal to 0.278. The plot of Call option prices is given on Figure 5.4.

| Model | Fitting error |
|---|---|
| CGMY1 | 0.0050 |
| CGMY2 | 0.0090 |
| CGMY3 | 0.0096 |
| Bl.-Sch. | 0.0420 |

Table 5.2: The results of the fitting error computations (5.29) for the Call option prices with parametres from Table 5.1, $r = 0.0465$, $T = 0.14$, $dx = 0.01$, $dt = 0.02$, $\epsilon = 0.04$ (the parameter of the finite-difference scheme), $S_0 = 356$, for $E$ equal to $260 \ldots 430$.

| Model | Fitting error |
|---|---|
| CGMY1 | 0.0050 |
| CGMY2 | 0.0090 |
| CGMY3 | 0.0096 |
| Bl.-Sch. | 0.0420 |

Table 5.3: The results of the fitting error computations (5.29) for the Call option prices with parametres from Table 5.1, $r = 0.0465$, $T = 0.14$, $dx = 0.01$, $dt = 0.02$, $\epsilon = 0.04$ (the parameter of the finite-difference scheme), $S_0 = 356$, for $E$ equal to 310, 330, 350, 370, 390.

| Model | Put | Call | Double-barrier put | Up-and-out call |
|---|---|---|---|---|
| CGMY1 | 11.89 | 20.21 | 5.65 | 5.63 |
| CGMY2 | 11.7 | 20.01 | 5.66 | 5.66 |
| CGMY3 | 11.67 | 19.99 | 5.66 | 5.66 |
| Bl.-Sch. | 10.65 | 18.99 | 5.67 | 5.82 |
| Bid | 14.75 | 19.25 | - | - |
| Ask | 16.6 | 20.5 | - | - |

Table 5.4: The results of the computation by the CGMY model and the Black-Scholes model for the at-the-money options on H&M company stock with the same parameters as for Table 5.2.

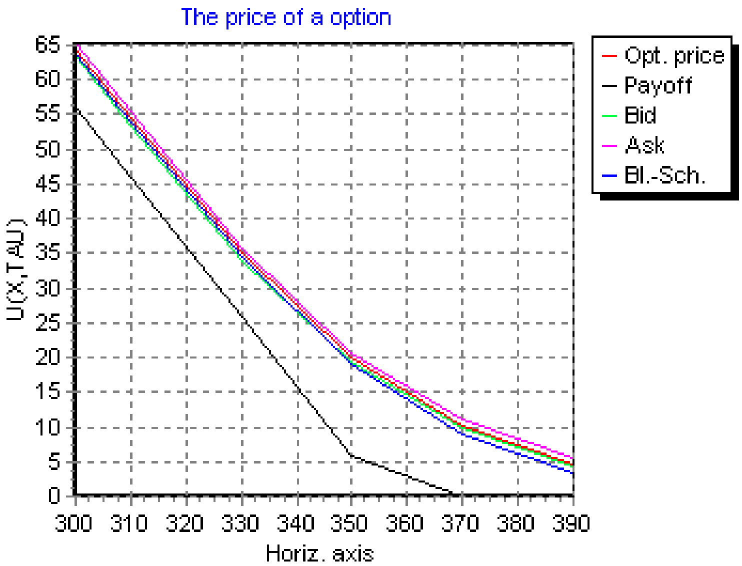


Figure 5.4: The results of the computation of the Call option (on June'08) prices for the H&M company on the 2nd of May'08 (for the spot price equal to 356) .

# Chapter 6

# Conclusions

Let us summarize the obtained results.

Our main theoretical results are contained in Theorems 8 and 9. In both cases the price of a vanilla Call/Put option is expressed with the use of the Radom-Nykodim derivative of the martingale measure with respect to the physical measure. In Theorem 8 we assume that the time moment, when the parameters of the model change, is deterministic. In Theorem 9 the mentioned time moment is random. In the Chapter 4 we regard the generalizations of the BS model for which I prove two theorems (Theorem 8, Theorem 9). The main tool which was used in proofs of this two theorems is the Girsanov's theorem. We can consider more complicate models with several parameter jumps or even a model, where the volatilty is described by a general predictable random process $\sigma(t, \omega)$.

Our computational results can be described as follows. We compare the option prices from the Swedish market with those obtained by the Black-Scholes model as well as by our models. As concerning the first model under assumption of a good prediction of $\sigma$ it gives the results better than the Black-Scholes.

Lets us turn to the second model. There, following ([4]), we obtain the partial integro-differential equation for option price. A special feature of our approach is the use of the method of moments for the parameter evaluation of the CGMY process. We implemententent the finite-difference scheme for solving of the mentioned partial integro-differential equation.

For numerical experiments we use the asset and option prices of the H&M company. The results for vanilla Call/Put prices, Double-Barrier Put and Up-and-out Call options are presented in Section 5.4. This results show that the CGMY model fits the real data better than the Black-Scholes model in the sence of avarage relative deviation, especially for the cases when the strike price is close to the spot price.

# Bibliography

[1] M. Abramovitz and I. Stegun (1972), *Hand book of Mathematical Functions.* Dover Publications, New York, USE.

[2] J. Bertoin (1996), *L'evy processes.* Cambridge University Press, Great Britain.

[3] P. Carr, H. Geman, D.B. Madan and M. Yor, (2000). *The Fine Structure of Asset Returns: An Empirical Investigation.* Journal of Business, vol. 75, no. 2.

[4] R. Cont and E. Voltchkova, (2003) *A finite difference scheme for option pricing in jump diffusion and exponential Levy models.* Internal Report CMAP number 513, September 2003. SIAM J. Numer. Anal. vol. 43, no. 4, pp. 1596-1626. CMAP-Ecole Polytechnique, France.

[5] T. Fujiwara and Y. Miyahara, (2003). *The Minimal Entropy Martingale Measures for Geometric L'evy Processes.* Finance and Stochastics 7(2003), pp.509-531.

[6] J.C. Hull *Options, Futures, and Other Derivatives. 5th edition.* Prentice Hall, USA.

[7] J. Jacod and A. Shiryaev (1980), *Limit Theorems for Stochastic Processes.* Berlin: Springer- Verlag.

[8] Y. Miyahara, (2004). *A Note on Esscher Transformed Martingale Measures for Geometric L'evy Processes.* Discussion Papers in Economics, Nagoya City University No. 379, pp. 1-14.

[9] Y. Miyahara, (2005), *[GLP & MEMM] Pricing Models and Related Problems.*, in Proceedings of SPA-05 (Ritsumeikan Univ.).

[10] Y. Miyahara and A. Novikov, (2002). *Geometric L'evy Process Pricing Model.* Proceedings of Steklov Mathematical Institute, vol. 237 (2002), pp.176-191.

[11] Y. Miyahara, (2002). *Estimation of Levy Processes.* Discussion papers in economics, Nagoya City University No.318, pp.1-36.

[12] A.N. Shiryaev (1999), *Essentials of Stochastic Finance: Facts, Models, Theory.* World Scientific, www.worldscientific.com.

[13] M.G. Garroniand, J.L.Menaldi (2002), *Second Order Elliptic Integro-Dierential Problems.* Chapman&Hall/CRC, Boca Raton, FL.

[14] G.Barles, R.Buckdahn and E.Pardoux (1997), *BSDEs and integral-partial differentialequations.* Stochastics Stochastics Rep., 60, pp.5783.

[15] H.Pham (1998) *Optimal stopping of controlled jump-diffusion processes: A viscosity solution approach.* J.Math. Systems Estim. Control, 8, pp.127.

# Appendix

**(A) The Black-Scholes model computations.**

In this Apppendix is given the text of the program solution in the Maple software for the computation of option prices using the finite-difference scheme for the Black-Scholes model with the deterministic time of changing parameters.

```
with(plots);C_real_Bid:=[[260,102],[270,92],[280,82],[290,72],
[300,63.25],[310,53.25],[330,33.75],[350,19.25],[370,10],
[390,4.25],[410,1.65],[430,0.5]];
> C_real_Ask:=[
> [260,107],
> [270,97],
> [280,87],
> [290,77],
> [300,65.25],
> [310,55.25],
> [330,35.75],
> [350,20.5],
> [370,11],
> [390,5.5],
> [410,2.4],
> [430,1.15]
> ];
> C_real_Bid_2:=[[300,63.25],[310,53.25],[330,33.75],
[350,19.25],[370,10],[390,4.25]];
> C_real_Ask_2:=[
> [300,65.25],
> [310,55.25],
> [330,35.75],
> [350,20.5],
> [370,11],
```

```
> [390,5.5]
> ];
> r:=0.0465; Sigma:=0.278; T:=50/262; t:=0; E:=350;
S0:=365.5;tau:=9/262;Sigma_1:=Sigma;Sigma_2:=0.238;
> d1 := (r,Sigma,E,S,T,t) ->
(ln(S/E)+(r+(Sigma^2)/2)*(T-t))/(Sigma*(T-t)^(1/2));
> d2 := (r,Sigma,E,S,T,t) ->
(ln(S/E)+(r-(Sigma^2)/2)*(T-t))/(Sigma*(T-t)^(1/2));
> d3_H := (r,Sigma_1,Sigma_2,E,S,T,t) ->
(ln(S/E)+(r+(Sigma_1^2)/2)*(tau-t)*Heaviside(tau-t)+
(r+(Sigma_2^2)/2)*(T-t*Heaviside(t-tau)-
tau*Heaviside(tau-t)))/
 (sqrt(Sigma_1^2*(tau-t)*Heaviside(tau-t)+
 Sigma_2^2*(T-t*Heaviside(t-tau)-tau*Heaviside(tau-t))));
> d3 := (r,Sigma_1,Sigma_2,E,S,T,t) ->
(ln(S/E)+(r+(Sigma_1^2)/2)*tau+(r+(Sigma_2^2)/2)*
(T-t-tau))/(sqrt(Sigma_1^2*tau+Sigma_2^2*(T-t-tau)));
> d4_H := (r,Sigma_1,Sigma_2,E,S,T,t) ->
(ln(S/E)+(r-(Sigma_1^2)/2)*(tau-t)*Heaviside(tau-t)+
(r-(Sigma_2^2)/2)*(T-t*Heaviside(t-tau)
-tau*Heaviside(tau-t)))/
(sqrt(Sigma_1^2*(tau-t)*Heaviside(tau-t)+Sigma_2^2*(T-
t*Heaviside(t-tau)-tau*Heaviside(tau-t))));
> d4 := (r,Sigma_1,Sigma_2,E,S,T,t) ->
(ln(S/E)+(r-(Sigma_1^2)/2)*tau+(r-(Sigma_2^2)/2)*
(T-t-tau))/ (sqrt(Sigma_1^2*tau+Sigma_2^2*(T-t-tau)));

> N := (x) -> Int(exp(-(y^2)/2)/(2*3.1458)^(1/2),
y=-infinity..x);

> C := (r,Sigma,E,S,T,t) ->
S*N(d1(r,Sigma,E,S,T,t))-E*exp(-r*(T-t))*
N(d2(r,Sigma,E,S,T,t));
> P := (r,Sigma,E,S,T,t) ->
E*exp(-r*(T-t))*N(-d2(r,Sigma,E,S,T,t))-
S*N(-d1(r,Sigma,E,S,T,t));
> C_BSRCV := (r,Sigma_1,Sigma_2,E,S,T,t) ->
S*N(d3(r,Sigma_1,Sigma_2,E,S,T,t))-
E*exp(-r*(T-t))*N(d4(r,Sigma_1,Sigma_2,E,S,T,t));
> P_BSRCV := (r,Sigma_1,Sigma_2,E,S,T,t) ->
E*exp(-r*(T-t))*N(-d3(r,Sigma_1,Sigma_2,E,S,T,t))-
```

```
S*N(-d4(r,Sigma_1,Sigma_2,E,S,T,t));
> C_BSRCV_H := (r,Sigma_1,Sigma_2,E,S,T,t) ->
S*N(d3_H(r,Sigma_1,Sigma_2,E,S,T,t))-
E*exp(-r*(T-t))*N(d4_H(r,Sigma_1,Sigma_2,E,S,T,t));
> P_BSRCV_H := (r,Sigma_1,Sigma_2,E,S,T,t) ->
E*exp(-r*(T-t))*N(-d3_H(r,Sigma_1,Sigma_2,E,S,T,t))-
S*N(-d4_H(r,Sigma_1,Sigma_2,E,S,T,t));

> CB:=plot({C_real_Bid},color=[red],linestyle=[solid]):
> CA:=plot({C_real_Ask},color=[black],linestyle=[solid]):
>
> CBS:=plot({C(r,Sigma,EE,S0,T,t)},EE=260..430,
numpoints=20,color=[green],linestyle=[dashdot]):
> CBSRCV:=plot({C_BSRCV_H(r,Sigma_1,Sigma_2,EE,S0,T,t)},
EE=260..430,numpoints=20,color=[blue],linestyle=[dash]):
> display(CB,CA,CBS,CBSRCV);

> plot3d(C_BSRCV_H(r,Sigma_1,Sigma_2,E,S,T,tt),
S=260..430,tt=0..T,numpoints=150);
> plot3d(C_BSRCV_H(r,Sigma_1,Sigma_2,EE,S0,T,tt)-
C(r,Sigma,EE,S0,T,tt),EE=75..125,tt=0..1/12,numpoints=300);
> plot3d(C_BSRCV_H(r,Sigma_1,Sigma_2,S0/SKratio,S0,T,tt)-
C(r,Sigma,S0/SKratio,S0,T,tt),tt=0..1/12,SKratio=0.85..1.15,
numpoints=200);
> plot3d(C_BSRCV_H(r,Sigma_1,Sigma_2,S0/SKratio,S,T,tt)-
C(r,Sigma,S0/SKratio,S,T,tt),S=60..140,SKratio=0.85..1.15,
numpoints=200);
> plot3d(C(r,Sigma,S0/SKratio,S0,T,tt),
tt=0..1/12,SKratio=0.85..1.15,numpoints=100);
> plot3d(C_BSRCV_H(r,Sigma_1,Sigma_2,S0/SKratio,S0,T,tt),
tt=0..1/12,SKratio=0.85..1.15,numpoints=100);
```

# Appendix

**(B) The CGMY Levy measure.**

In this Apppendix the text of the program solution is given in Maple software for the estimation of parameters of the CGMY process by the method of moments.

```
> with(linalg):with(plots):
> phi_theor := u -> exp(I*(b+C*Gamma(-Y)*Y*(M^(Y-1)-
G^(Y-1)))*u+C*Gamma(-Y)*((M-I*u)^Y-M^Y+(G+I*u)^Y-G^Y));
>
> n:=2200: sigma:=0.278:
> Delta:=array(1..n):Z:=array(1..n,1..4):ZZ:=array(1..n):
> Z:=readdata("c:\DataHM_010100-181008.txt",string,string,
string,float,4):
> for i from 1 to n by 1 do Delta[i]:=Z[i+1,4]-Z[i,4]:
end do:
> listplot(Delta,symbol=circle);

> for kk from 1 to n by 1 do ZZ[kk]:=Z[kk,4] end do:
>
> listplot(ZZ,symbol=circle);

> phi := (u,n) -> sum(exp(I*u*Delta[k]),k=1..n)/n;

> Y:=eval(ln(abs(Re(ln(phi(u,n)))))/eval(ln(100000)),
u=100000):
> Y:=simplify(Y);
>
> u:=100000:

> _EnvAllSolutions := true:
> C:=solve(Re(ln(phi(u,n)))/(u^Y)=
```

```
2*CCC*GAMMA(-Y)*cos(Y*sigma^2/2),CCC);

> b:=0:m2:=0:m3:=0:
> for i from 1 to n by 1 do b:=b+Z[i,4]; end do: b:=b/n;
> for i from 1 to n by 1 do m2:=m2+Z[i,4]^2; end do: m2:=m2/n;
> for i from 1 to n by 1 do m3:=m3+Z[i,4]^3; end do: m3:=m3/n;
> m1:=b:
> h2:=m2-m1^2;h3:=m3-3*m2*m1+2*m1^3;
Plotting error, empty plot
> plot({(h2/(C*GAMMA(-Y)*Y*(Y-1))-M^(Y-2))^(2-Y),
(h3/(C*GAMMA(-Y)*Y*(Y-1)*(Y-2))+M^(Y-3))^(3-Y)},M=0.1..1);

> plot({(-h3/(C*GAMMA(-Y)*Y*(Y-1)*(Y-2))+G^(Y-3))^(3-Y),
(h2/(C*GAMMA(-Y)*Y*(Y-1))-G^(Y-2))^(2-Y)},G=0.2..0.99);

> implicitplot(C*GAMMA(-Y)*Y*(Y-1)*(Y-2)*(MMM^(Y-3)-
GGG^(Y-3))=h3,GGG=0.035..1,MMM=0.035..1,axes=boxed);

> implicitplot({C*GAMMA(-Y)*Y*(Y-1)*(MMM^(Y-2)+GGG^(Y-2))=h2,
C*GAMMA(-Y)*Y*(Y-1)*(Y-2)*(MMM^(Y-3)-GGG^(Y-3))=h3},
GGG=0.001..0.01,MMM=0.001..0.01,axes=boxed);

>
> fsolve({C*GAMMA(-Y)*Y*(Y-1)*(GGG^(Y-2)+M^(Y-2))=h2,
C*GAMMA(-Y)*Y*(Y-1)*(Y-2)*(M^(Y-3)-GGG^(Y-3))=h3},{GGG,M},
{GGG=0.001..0.01,M=0.001..0.01});

              {M = 0.003069195904, GGG = 0.002862222171}

> G:=GGG:
> print(b,C,G,M,Y);

           246.3536364, 0.05020207653, GGG, M, 0.1079459547
```

# Appendix

**(C) The finite-difference scheme for the CGMY model**

In this Apppendix is given the text of the program solution in the Delphi software for the computation of option prices using the finite-difference scheme for the CGMY model.

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
   Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, Menus, Grids, TeeProcs,
   TeEngine, Chart,
  Series, normaldistr;

type
  TForm1 = class(TForm)
    GroupBox1: TGroupBox;
    LabeledEdit5: TLabeledEdit;
    LabeledEdit6: TLabeledEdit;
    LabeledEdit7: TLabeledEdit;
    LabeledEdit8: TLabeledEdit;
    GroupBox2: TGroupBox;
    LabeledEdit1: TLabeledEdit;
    LabeledEdit3: TLabeledEdit;
    LabeledEdit4: TLabeledEdit;
    MainMenu1: TMainMenu;
    File1: TMenuItem;
    Exit1: TMenuItem;
    About1: TMenuItem;
```

```
Button1: TButton;
GroupBox3: TGroupBox;
LabeledEdit9: TLabeledEdit;
LabeledEdit10: TLabeledEdit;
LabeledEdit13: TLabeledEdit;
LabeledEdit15: TLabeledEdit;
LabeledEdit16: TLabeledEdit;
LabeledEdit14: TLabeledEdit;
LabeledEdit11: TLabeledEdit;
GroupBox4: TGroupBox;
LabeledEdit17: TLabeledEdit;
LabeledEdit18: TLabeledEdit;
LabeledEdit19: TLabeledEdit;
Chart1: TChart;
Series1: TLineSeries;
Chart2: TChart;
Series2: TLineSeries;
Button3: TButton;
RadioButton1: TRadioButton;
RadioButton2: TRadioButton;
CheckBox1: TCheckBox;
CheckBox2: TCheckBox;
LabeledEdit23: TLabeledEdit;
LabeledEdit24: TLabeledEdit;
Series3: TLineSeries;
GroupBox5: TGroupBox;
RadioButton5: TRadioButton;
RadioButton6: TRadioButton;
GroupBox6: TGroupBox;
RadioButton3: TRadioButton;
RadioButton4: TRadioButton;
LabeledEdit2: TLabeledEdit;
LabeledEdit12: TLabeledEdit;
LabeledEdit21: TLabeledEdit;
LabeledEdit22: TLabeledEdit;
GroupBox7: TGroupBox;
RadioButton11: TRadioButton;
RadioButton10: TRadioButton;
RadioButton9: TRadioButton;
RadioButton12: TRadioButton;
RadioButton13: TRadioButton;
```

```
    LabeledEdit20: TLabeledEdit;
    LabeledEdit25: TLabeledEdit;
    Series4: TLineSeries;
    Series5: TLineSeries;
    Series6: TLineSeries;
    LabeledEdit26: TLabeledEdit;
    LabeledEdit27: TLabeledEdit;
    CheckBox4: TCheckBox;
    RadioButton7: TRadioButton;
    RadioButton8: TRadioButton;
    RadioButton14: TRadioButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure CheckBox1Click(Sender: TObject);
    procedure CheckBox2Click(Sender: TObject);
    procedure LabeledEdit1Change(Sender: TObject);
    procedure LabeledEdit12Change(Sender: TObject);
    procedure LabeledEdit2Change(Sender: TObject);
    procedure LabeledEdit10Change(Sender: TObject);
    procedure LabeledEdit21Change(Sender: TObject);
    procedure LabeledEdit22Change(Sender: TObject);
    procedure LabeledEdit24Change(Sender: TObject);
    procedure RadioButton2Click(Sender: TObject);
    procedure RadioButton1Click(Sender: TObject);
    procedure RadioButton7Click(Sender: TObject);
    procedure RadioButton13Click(Sender: TObject);
    procedure RadioButton3Click(Sender: TObject);
    procedure RadioButton4Click(Sender: TObject);
    procedure CheckBox3Click(Sender: TObject);
    procedure RadioButton11Click(Sender: TObject);
    procedure RadioButton12Click(Sender: TObject);
    procedure RadioButton9Click(Sender: TObject);
    procedure RadioButton10Click(Sender: TObject);
    procedure RadioButton8Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
```

```
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

const Nmax=600; Mmax=200;
type
  ArrMax=array[0..Nmax] of real;
  Arr   =array[0..Mmax,-2..Nmax+2] of real;
  ArrNu =array[-Nmax*2..Nmax*2] of real;

function K(x,C,G,M,Y:real):real;
begin

  if Form1.RadioButton1.Checked or Form1.RadioButton7.Checked
     or Form1.RadioButton8.Checked then begin
    if x<0 then
      K:=C*Exp(-G*Abs(x))/(exp(ln(Abs(x))*(1 + Y)))
         else
          begin
            if x<>0 then K:=C*Exp(-M*Abs(x))/
            (exp(ln(Abs(x))*(1 + Y))) else K:=1000000000;
          end;
                                     end;
  if Form1.RadioButton14.Checked then K:=0;

end;

function Nu(Epsilon:real;K_l,K_r:integer;
dx,C,G,M,Y:real):ArrNu;
var i:integer; x1,x2,x3,x4,x5,dxx,nuInt:real;
begin
  x1:=0;  x2:=0;  x3:=0;  x4:=0;  x5:=0;
  for i:=K_l to K_r do
   if (abs((i-1/2)*dx)>=Epsilon)
   AND (abs((i+1/2)*dx)>=Epsilon) then
     begin
```

```
        dxx:=dx/4;
        x1:=(i-1/2)*dx;
        x2:=(i-1/2)*dx+dxx;
        x3:=(i-1/2)*dx+2*dxx;
        x4:=(i-1/2)*dx+3*dxx;
        x5:=(i+1/2)*dx;


        nuInt:=0;
        nuInt:=nuInt+((x2-x1)/6)*( K(x1,C,G,M,Y)+
          4*K((x1+x2)/2,C,G,M,Y)+K(x2,C,G,M,Y) );

        nuInt:=nuInt+((x3-x2)/6)*( K(x2,C,G,M,Y)+
          4*K((x2+x3)/2,C,G,M,Y)+K(x3,C,G,M,Y) );

        nuInt:=nuInt+((x4-x3)/6)*( K(x3,C,G,M,Y)+
          4*K((x3+x4)/2,C,G,M,Y)+K(x4,C,G,M,Y) );

        nuInt:=nuInt+((x5-x4)/6)*( K(x4,C,G,M,Y)+
          4*K((x4+x5)/2,C,G,M,Y)+K(x5,C,G,M,Y) );
        Nu[i]:=nuInt;
      end else Nu[i]:=0;
end;

function Kl(dx,Bl:real):integer;
var K:integer;
begin
  K:=0;
  while (K-1/2)*dx>Bl do K:=K-1;
  Kl:=K;
end;

function Kr(dx,Br:real):integer;
var K:integer;
begin
  K:=0;
  while ((K+1/2)*dx)<Br do K:=K+1;
  Kr:=K;
end;

{function SigmaEpsSquare_Main(Epsilon,dx,C,G,M,Y:real):real;
```

```
var s:real; j,K_l,K_r:integer; NuInt:ArrNu;
begin
  j:=0;s:=0;
  K_l:=Kl(dx,-Epsilon);
  K_r:=Kr(dx,Epsilon);
  NuInt:=Nu(Epsilon,K_l,K_r,dx,C,G,M,Y);
  for j:=K_l to K_r do
    s:=s+NuInt[j];


  s:=0;x:=-Epsilon;
  while x<=Epsilon do
   begin
     s:=s+((x*x*K(x,C,G,M,Y)+
       (x+dx2)*(x+dx2)*K(x+dx2,C,G,M,Y))/2)*dx2;
     x:=x+dx2;
   end;
  SigmaEpsSquare:=s;
end;           }

function LambdaEps_Main(Epsilon,Abig:real;
N:integer;dx,C,G,M,Y,Bl,Br:real):real;
var s:real; j,K_l,K_r:integer; NuInt:ArrNu;
begin
  j:=0;s:=0;
  K_l:=Kl(dx,Bl);
  K_r:=Kr(dx,Br);
  NuInt:=Nu(Epsilon,K_l,K_r,dx,C,G,M,Y);
  for j:=K_l to K_r do
    if abs(j*dx)>=epsilon then s:=s+NuInt[j];
  LambdaEps_Main:=s;
end;

function AlfaEps_Main(epsilon,A:real;
N:integer;dx,C,G,M,Y,Bl,Br:real):real;
var s:real;j,K_l,K_r:integer; NuInt: ArrNu;
begin
  j:=0;s:=0;
  K_l:=Kl(dx,Bl);
  K_r:=Kr(dx,Br);
  NuInt:=Nu(Epsilon,K_l,K_r,dx,C,G,M,Y);
```

```
  for j:=K_l to K_r do
    if abs(j*dx)>=epsilon then s:=s+(exp(j*dx)-1)*NuInt[j];
  AlfaEps_Main:=s;
end;

function SigmaEpsSquare(Epsilon,dx2,C,G,M,Y:real):real;
var s,x:real;
begin
  s:=0;x:=-Epsilon;
  while x<=Epsilon do
   begin
     s:=s+((x*x*K(x,C,G,M,Y)+
       (x+dx2)*(x+dx2)*K(x+dx2,C,G,M,Y))/2)*dx2;
     x:=x+dx2;
   end;
  SigmaEpsSquare:=s;
end;

function LambdaEps(Epsilon:real;dx2,C,G,M,Y,Bl,Br:real):real;
var s:real; x:real;
begin
  s:=0;
  x:=Bl;
  while x<=Br do
   begin
     if abs(x)>=Epsilon then s:=s+((K(x,C,G,M,Y)+
       K(x+dx2,C,G,M,Y))/2)*dx2;
     x:=x+dx2;
   end;
  LambdaEps:=s;
end;

function AlfaEps(Epsilon:real;dx2,C,G,M,Y,Bl,Br:real):real;
var x,s:real;
begin
  s:=0;
  x:=Bl;
  while x<=Br do
   begin
     if abs(x)>=Epsilon then s:=s+((exp(x)-1)*K(x,C,G,M,Y)+
       (exp(x+dx2)-1)*K(x+dx2,C,G,M,Y))*dx2/2;
```

```
    x:=x+dx2;
   end;
  AlfaEps:=s;
end;

function h(x,E,S0,UB,DB:real):real;
begin
  if Form1.RadioButton9.Checked then
    if (E-S0*exp(x))>0 then h:=(E-S0*exp(x)) else h:=0;
  if Form1.RadioButton10.Checked then
    if (S0*exp(x)-E)>0 then h:=(S0*exp(x)-E) else h:=0;
  if Form1.RadioButton11.Checked then
    if ((E-S0*exp(x))>0) and (x<ln(UB/S0)) and (x>ln(DB/S0))
       then h:=(E-S0*exp(x)) else h:=0;
  if Form1.RadioButton12.Checked then
    if ((S0*exp(x)-E)>0) and (x<ln(UB/S0))
      then h:=(S0*exp(x)-E) else h:=0;
end;

function g(x,E,S0,U,D:real):real;
begin
  if Form1.RadioButton5.Checked then g:=h(x,E,S0,U,D);
  if Form1.RadioButton6.Checked then g:=0;//sin(x);
  g:=0;
end;

procedure MetodAlgebrProg(N:integer;a:real; b:real;
c:real;var d,x:ArrMax);
var
  ksi,etta:ArrMax;
  i:integer;
begin

  for i:=1 to N do x[i]:=0;
  b:=-b;
  ksi[0] :=c/b;
  etta[0]:=-d[0]/b;
  for i:=1 to N do
   begin
    ksi[i] :=c/(b-a*ksi[i-1]);
    etta[i]:=(d[i]-a*etta[i-1])/(a*ksi[i-1]-b);
```

```
   end;
  ksi[N]:=0;

  i:=0;x[N]:=etta[N];
  for i:=N-1 downto 0 do x[i]:=ksi[i]*x[i+1]+etta[i];
end;

procedure FD_PIDE(var u:Arr;N,M:integer;
E,S0,UB,DB,Bl,Br,Abig,Sigma2,dx2,CC,GG,MM,YY,Epsilon,T,r
,SigmaEps2,LambdaEps,AlfaEps:real);
var
  a,b,c,s,dx,dt,xx,x:real;
  d,MAlgPr:ArrMax;
  NuInt:ArrNu;
  q:boolean;
  K_l,K_r,i,j,k:integer;
begin
  j:=0;s:=0;
  dx:=2*Abig/N;
  dt:=T/M;
  K_l:=Kl(dx,Bl);
  K_r:=Kr(dx,Br);

  if ((Sigma2+SigmaEps2)/2-r+AlfaEps)<0 then q:=true
                                      else q:=false;

  if q then
      begin
        a:=-(Sigma2+SigmaEps2)/(2*dx*dx);
        b:=(Sigma2+SigmaEps2)/(dx*dx)-
         ((Sigma2+SigmaEps2)/2-r+AlfaEps)/dx+LambdaEps+1/dt;
        c:=-(Sigma2+SigmaEps2)/(2*dx*dx)+
         ((Sigma2+SigmaEps2)/2-r+AlfaEps)/dx;
      end
          else
      begin
        a:=-(Sigma2+SigmaEps2)/(2*dx*dx)-
         ((Sigma2+SigmaEps2)/2-r+AlfaEps)/dx;
        b:=(Sigma2+SigmaEps2)/(dx*dx)+
         ((Sigma2+SigmaEps2)/2-r+AlfaEps)/dx+LambdaEps+1/dt;
        c:=-(Sigma2+SigmaEps2)/(2*dx*dx);
```

```
    end;

NuInt:=Nu(Epsilon,K_l,K_r,dx,CC,GG,MM,YY);

for j:=1 to M do
 begin
   for i:=0 to N do
    begin
      s:=0;

      if Form1.RadioButton11.Checked then
       begin
         x:=-Abig+i*2*Abig/N;
         if (x>=ln(UB/S0)) and (x<=ln(DB/S0))
           then u[j-1,i]:=0;
       end;

      if Form1.RadioButton12.Checked then
       begin
         x:=-Abig+i*2*Abig/N;
         if x>=ln(UB/S0) then u[j-1,i]:=0;
       end;


      for k:=K_l to K_r do begin
         xx:=0;
         if ((i+k)>N) or ((i+k)<0)
            then xx:=NuInt[k]*g(-Abig+dx*(i+k),E,S0,UB,DB)
            else xx:=NuInt[k]*u[j-1,i+k];
         s:=s+xx;

                          end;

      d[i]:=u[j-1,i]/dt+s;
    end;

   MetodAlgebrProg(N,a,b,c,d,MAlgPr);
   for i:=0 to N do u[j,i]:=MAlgPr[i];

 end;
```

```
end;



// ---------------Black-Scholes-------------
function Sigma(t,a,b,c:real):real;
begin
  Sigma:=a*t*t+b*t+c;
end;

function dt_T(t,Tbig,a,b,c:real):real;
var d:real;
begin
  if t>0 then
  d:=sqrt(a*a*(exp(5*ln(Tbig))-exp(5*ln(t)))+
         2*a*b*(exp(4*ln(Tbig))-exp(4*ln(t)))+
         (b*b+2*a*c)*(exp(3*ln(Tbig))-exp(3*ln(t)))+
         2*b*c*(Tbig*Tbig-t*t)+c*c*(Tbig-t))
           else
  d:=sqrt(a*a*(exp(5*ln(Tbig)))+
         2*a*b*(exp(4*ln(Tbig)))+
         (b*b+2*a*c)*(exp(3*ln(Tbig)))+
         2*b*c*(Tbig*Tbig)+c*c*(Tbig));
  dt_T:=d;
end;

function d_one(t,Tbig,K,St,r,a,b,c:real):real;
var d:real;
begin
  d:=dt_T(t,Tbig,a,b,c)-(ln(K/St)-(r*(Tbig-t)-
    Sigma(Tbig,a,b,c)*Sigma(Tbig,a,b,c)*Tbig/2+
    Sigma(t,a,b,c)*Sigma(t,a,b,c)*t/2))/dt_T(t,Tbig,a,b,c);
  d_one:=d;
end;

function d_two(t,Tbig,K,St,r,a,b,c:real):real;
var STb2,STs2,d,dtT,d4,d3:real;
begin
  STb2:=Sigma(Tbig,a,b,c)*Sigma(Tbig,a,b,c);
  STs2:=Sigma(t,a,b,c)*Sigma(t,a,b,c);
  dtT:=dt_T(t,Tbig,a,b,c);
```

```
  d:=-(ln(K/St)-(r*(Tbig-t)-STb2*Tbig/2+STs2*t/2))/dtT;
  d_two:=d;
end;

function CallPrice(t,Tbig,K,St,r,a,b,c:real):real;
var CP,F1,F2,I1,I2:real;
begin
  F1:=(1+Erf(d_one(t,Tbig,K,St,r,a,b,c)/sqrt(2)))/2;
  F2:=(1+Erf(d_two(t,Tbig,K,St,r,a,b,c)/sqrt(2)))/2;
  I1:=St*F1*exp((dt_T(t,Tbig,a,b,c)*dt_T(t,Tbig,a,b,c)-
    Sigma(Tbig,a,b,c)*Sigma(Tbig,a,b,c)*Tbig+
    Sigma(t,a,b,c)*Sigma(t,a,b,c)*t)/2);
  I2:=K*exp(-r*(Tbig-t))*F2;
  CP:=I1  -  I2;
  CallPrice:=CP;
end;

function PutPrice(t,Tbig,K,St,r,a,b,c:real):real;
var PP,F1,F2,I1,I2:real;
begin
  F1:=(1+Erf(-d_two(t,Tbig,K,St,r,a,b,c)/sqrt(2)))/2;
  F2:=(1+Erf(-d_one(t,Tbig,K,St,r,a,b,c)/sqrt(2)))/2;
  I1:=K*exp(-r*(Tbig-t))*F1;
  I2:=St*F2*exp((dt_T(t,Tbig,a,b,c)*dt_T(t,Tbig,a,b,c)-
    Sigma(Tbig,a,b,c)*Sigma(Tbig,a,b,c)*Tbig+
    Sigma(t,a,b,c)*Sigma(t,a,b,c)*t)/2);
  PP:=I1  -  I2;
  PutPrice:=PP;
end;

//-------------------------------------------------------



procedure TForm1.Button1Click(Sender: TObject);
var
  N,M,i,ii,jj,j:integer;
  E0,dx_e,y,s,E,Bl,Br,Abig,Sigma2,dx2,CC,GG,MM,YY,
    Epsilon,T,S0,r,Lambda,x,tau,NumberESteps,RangeE:real;
  SigmaE,LambdaE,AlfaE,tSmall:real;
```

```
  u:Arr;
  d1,d2,CP,UB,DB:real;
  q1,q2,q3,q4,q5:Boolean;
begin
  N :=StrToInt(LabeledEdit12.Text);
  M :=StrToInt(LabeledEdit2.Text);
  E :=StrToFloat(LabeledEdit9.Text);
  Bl:=StrToFloat(LabeledEdit3.Text);
  Br:=StrToFloat(LabeledEdit4.Text);
  Abig:=StrToFloat(LabeledEdit1.Text);
  Sigma2:=StrToFloat(LabeledEdit13.Text)*
          StrToFloat(LabeledEdit13.Text);
  dx2:=StrToFloat(LabeledEdit14.Text);
  CC:=StrToFloat(LabeledEdit5.Text);
  GG:=StrToFloat(LabeledEdit6.Text);
  MM:=StrToFloat(LabeledEdit7.Text);
  YY:=StrToFloat(LabeledEdit8.Text);
  Epsilon:=StrToFloat(LabeledEdit11.Text);
  T:=StrToFloat(LabeledEdit10.Text);
  S0:=StrToFloat(LabeledEdit15.Text);
  r:=StrToFloat(LabeledEdit16.Text);
  tSmall:=StrToFloat(LabeledEdit24.Text);
  UB:=StrToFloat(LabeledEdit26.Text);
  DB:=StrToFloat(LabeledEdit27.Text);

  NumberESteps:=StrToFloat(LabeledEdit25.Text);
  RangeE:=StrToFloat(LabeledEdit20.Text);

  if CheckBox4.Checked then LambdaE:=
    LambdaEps_Main(Epsilon,Abig,N,2*Abig/N,CC,GG,MM,YY,Bl,Br)
                       else LambdaE:=
    LambdaEps(Epsilon,dx2*10,CC,GG,MM,YY,Bl,Br);
  LabeledEdit17.Text:=FloatToStr(LambdaE);
  if CheckBox4.Checked then AlfaE:=
     AlfaEps_Main(Epsilon,Abig,N,2*Abig/N,CC,GG,MM,YY,Bl,Br)
                       else AlfaE:=
     AlfaEps(Epsilon,dx2*10,CC,GG,MM,YY,Bl,Br);
  LabeledEdit18.Text:=FloatToStr(LambdaE);

  SigmaE:=SigmaEpsSquare(Epsilon,dx2,CC,GG,MM,YY);
  LabeledEdit19.Text:=FloatToStr(SigmaE);
```

```
for i:=1 to N-1 do u[0,i]:=h(-Abig+i*2*Abig/N,E,S0,UB,DB);
for i:=0 to M do
 begin
   u[i,-2]:=g(-Abig,E,S0,UB,DB);
   u[i,-1]:=g(-Abig,E,S0,UB,DB);
   u[i,0]:=g(-Abig,E,S0,UB,DB);
   u[i,N]:=g( Abig,E,S0,UB,DB);
   u[i,N+1]:=g( Abig,E,S0,UB,DB);
   u[i,N+2]:=g( Abig,E,S0,UB,DB);
 end;

if RadioButton2.Checked then
  FD_PIDE(u,N,M,E,S0,UB,DB,Bl,Br,Abig,0,dx2,CC,GG,MM,YY,
   Epsilon,T,r,SigmaE{*SigmaE},LambdaE,AlfaE)
                                                    else
  FD_PIDE(u,N,M,E,S0,UB,DB,Bl,Br,Abig,Sigma2,dx2,CC,GG,
   MM,YY,Epsilon,T,r,SigmaE{*SigmaE},LambdaE,AlfaE);

jj:=0;
while (jj*T/M)<tSmall do jj:=jj+1;
if jj<>0 then jj:=jj-1;

i:=0;
while (-Abig+i*2*Abig/N)<0 do i:=i+1;
LabeledEdit23.Text:=FloatToStr(exp(-r*tau)*u[M-jj,i]);

LabeledEdit23.Text:=FloatToStr(exp(-r*tau)*u[M-jj,i]);

Chart2.Series[0].Clear;
Chart2.Series[1].Clear;
Chart2.Series[2].Clear;
Chart2.Series[3].Clear;
Chart2.Series[4].Clear;

ii:=0;
while (-Abig+ii*2*Abig/N)<0 do ii:=ii+1;
tau:=T-tSmall;

if not RadioButton13.Checked then
```

```
  begin

for i:=0 to N do
 begin
   x:=-Abig+i*2*Abig/N;

   if RadioButton3.Checked and (u[M-jj,i]>=Epsilon) then
    begin
      Chart2.Series[0].AddXY(x,exp(-r*tau)*u[M-jj,i]);
      if not RadioButton11.Checked and
         not RadioButton12.Checked then
           Chart2.Series[1].AddXY(x,h(x,E,S0,UB,DB));
    end;
   if (RadioButton4.Checked) and (abs(u[M-jj,i])>=0.01)
    then
    begin
      if (abs(E-exp(x)*S0)<=RangeE) then
        Chart2.Series[0].AddXY(exp(x)*S0,
          exp(-r*tau)*u[M-jj,i]);
      if (abs(E-exp(x)*S0)<=RangeE) and
        not RadioButton11.Checked and
        not RadioButton12.Checked then
          Chart2.Series[1].AddXY(exp(x)*S0,h(x,E,S0,UB,DB));
      if RadioButton9.Checked and
        not RadioButton14.Checked and
        (abs(E-exp(x)*S0)<=RangeE) then
          Chart2.Series[4].AddXY(exp(x)*S0,
           PutPrice(tSmall,T,E,exp(x)*S0,r,0,0,sqrt(Sigma2)));
      if RadioButton10.Checked and
        not RadioButton14.Checked and
        (abs(E-exp(x)*S0)<=RangeE) then
          Chart2.Series[4].AddXY(exp(x)*S0,
          CallPrice(tSmall,T,E,exp(x)*S0,r,0,0,sqrt(Sigma2)));
    end;
 end;

 if (RadioButton11.Checked or RadioButton12.Checked or
     RadioButton14.Checked) then
   begin
     q1:=RadioButton1.Checked;
     q2:=RadioButton2.Checked;
```

```
q3:=RadioButton7.Checked;
q4:=RadioButton8.Checked;
q5:=RadioButton14.Checked;
RadioButton14.Checked:=True;
for i:=1 to N-1 do u[0,i]:=
  h(-Abig+i*2*Abig/N,E,S0,UB,DB);
for i:=0 to M do
 begin
   u[i,-2]:=g(-Abig,E,S0,UB,DB);
   u[i,-1]:=g(-Abig,E,S0,UB,DB);
   u[i,0]:=g(-Abig,E,S0,UB,DB);
   u[i,N]:=g( Abig,E,S0,UB,DB);
   u[i,N+1]:=g( Abig,E,S0,UB,DB);
   u[i,N+2]:=g( Abig,E,S0,UB,DB);
 end;

 FD_PIDE(u,N,M,E,S0,UB,DB,Bl,Br,Abig,Sigma2,dx2,
   0,0,0,0,Epsilon,T,r,0,0,0);

 for i:=0 to N do
   begin
     x:=-Abig+i*2*Abig/N;

     if RadioButton3.Checked and (u[M-jj,i]>=Epsilon)
      then
       begin
         Chart2.Series[4].AddXY(x,exp(-r*tau)*u[M-jj,i]);
         if not RadioButton11.Checked and
            not RadioButton12.Checked then
            Chart2.Series[1].AddXY(x,h(x,E,S0,UB,DB));
       end;
     if (RadioButton4.Checked) and (abs(u[M-jj,i])>=0.01)
      then
       begin
         if abs(E-exp(x)*S0)<=RangeE then
           Chart2.Series[4].AddXY(exp(x)*S0,
           exp(-r*tau)*u[M-jj,i]);
         if (abs(E-exp(x)*S0)<=RangeE) and
            not RadioButton11.Checked and
            not RadioButton12.Checked then
              Chart2.Series[1].AddXY(exp(x)*S0,
```

```
                h(x,E,S0,UB,DB));
        end;
    end;
    RadioButton1.Checked:=q1;
    RadioButton2.Checked:=q2;
    RadioButton7.Checked:=q3;
    RadioButton8.Checked:=q4;
    RadioButton14.Checked:=q5;
end;
 end else begin
    if RadioButton10.Checked then begin
 Chart2.Series[2].AddXY(310,53.25);
 Chart2.Series[2].AddXY(330,33.75);
 Chart2.Series[2].AddXY(350,19.25);
 Chart2.Series[2].AddXY(370,10);
 Chart2.Series[2].AddXY(390,4.25);

 Chart2.Series[3].AddXY(310,55.25);
 Chart2.Series[3].AddXY(330,35.75);
 Chart2.Series[3].AddXY(350,20.5);
 Chart2.Series[3].AddXY(370,11  );
 Chart2.Series[3].AddXY(390,5.5 );
                                   end;

 E0:=E-RangeE;
 y:=E0;
 dx_e:=2*RangeE/NumberESteps;
 while y<=E+RangeE do begin
   for i:=1 to N-1 do u[0,i]:=
     h(-Abig+i*2*Abig/N,y,S0,UB,DB);
   for i:=0 to M do
    begin
      u[i,-2]:=g(-Abig,y,S0,UB,DB);
      u[i,-1]:=g(-Abig,y,S0,UB,DB);
      u[i,0]:=g(-Abig,y,S0,UB,DB);
      u[i,N]:=g( Abig,y,S0,UB,DB);
      u[i,N+1]:=g( Abig,y,S0,UB,DB);
      u[i,N+2]:=g( Abig,y,S0,UB,DB);
    end;

   if RadioButton2.Checked {or RadioButton8.Checked}
```

```
         then
          FD_PIDE(u,N,M,E,S0,UB,DB,Bl,Br,Abig,0,dx2,CC,GG,MM,
           YY,Epsilon,T,r,SigmaE{*SigmaE},LambdaE,AlfaE)
                                                        else
          FD_PIDE(u,N,M,E,S0,UB,DB,Bl,Br,Abig,Sigma2,dx2,CC,
           GG,MM,YY,Epsilon,T,r,SigmaE{*SigmaE},LambdaE,AlfaE);

        jj:=0;
        while (jj*T/M)<tSmall do jj:=jj+1;
        if jj<>0 then jj:=jj-1;
        i:=0;
        while (-Abig+i*2*Abig/N)<0 do i:=i+1;

        if (u[M-jj,i]>=Epsilon) then begin
        Chart2.Series[0].AddXY(y,exp(-r*tau)*u[M-jj,i]);
        Chart2.Series[1].AddXY(y,h(0,y,S0,UB,DB));
        if RadioButton9.Checked  then
          Chart2.Series[4].AddXY(y, PutPrice(tSmall,T,y,S0,
            r,0,0,sqrt(Sigma2)));
        if RadioButton10.Checked then
          Chart2.Series[4].AddXY(y,CallPrice(tSmall,T,y,S0,
            r,0,0,sqrt(Sigma2)));
                                    end;
        y:=y+dx_e;
                     end;
    end;



  Form1.Chart1.Series[0].Clear;
  for i:=0 to N do
   begin
     x:=-Abig/5+i*2*Abig/(N*5);
     if (K(x,CC,GG,MM,YY)<=300) and (K(x,CC,GG,MM,YY)<>0)
       then Form1.Chart1.Series[0].AddXY(x,K(x,CC,GG,MM,YY));
   end;


end;

procedure TForm1.RadioButton13Click(Sender: TObject);
```

```
begin
  LabeledEdit20.Enabled:=true;
  LabeledEdit25.Enabled:=true;
end;

procedure TForm1.Button2Click(Sender: TObject);
var Epsilon,CC,GG,MM,YY,x,dx:real;
begin

end;

procedure TForm1.Button3Click(Sender: TObject);
var
  N,M,i:integer;E,Bl,Br,Abig,Sigma2,dx2,CC,GG,MM,YY,Epsilon,
    T,S0,r,Lambda:real;
  u:Arr;
begin

  N :=StrToInt(LabeledEdit12.Text);
  M :=StrToInt(LabeledEdit2.Text);
  E :=StrToFloat(LabeledEdit9.Text);
  Bl:=StrToFloat(LabeledEdit3.Text);
  Br:=StrToFloat(LabeledEdit4.Text);
  Abig:=StrToFloat(LabeledEdit1.Text);
  Sigma2:=StrToFloat(LabeledEdit13.Text)*
    StrToFloat(LabeledEdit13.Text);
  dx2:=StrToFloat(LabeledEdit14.Text);
  CC:=StrToFloat(LabeledEdit5.Text);
  GG:=StrToFloat(LabeledEdit6.Text);
  MM:=StrToFloat(LabeledEdit7.Text);
  YY:=StrToFloat(LabeledEdit8.Text);
  Epsilon:=StrToFloat(LabeledEdit11.Text);
  T:=StrToFloat(LabeledEdit10.Text);
  S0:=StrToFloat(LabeledEdit15.Text);
  r:=StrToFloat(LabeledEdit16.Text);

  if not CheckBox4.Checked then
    Lambda:=LambdaEps(Epsilon,dx2*10,CC,GG,MM,YY,Bl,Br)
                           else
    Lambda:=LambdaEps_Main(Epsilon,Abig,N,2*Abig/N,CC,GG,
      MM,YY,Bl,Br);
```

```
  LabeledEdit17.Text:=FloatToStr(Lambda);
  if not CheckBox4.Checked then
    LabeledEdit18.Text:=FloatToStr(AlfaEps(Epsilon,dx2*10,
      CC,GG,MM,YY,Bl,Br))
                            else
    LabeledEdit18.Text:=FloatToStr(AlfaEps_Main(Epsilon,
      Abig,N,2*Abig/N,CC,GG,MM,YY,Bl,Br));
    LabeledEdit19.Text:=FloatToStr(SigmaEpsSquare(Epsilon,
      dx2,CC,GG,MM,YY))

end;

procedure TForm1.FormShow(Sender: TObject);
begin
  Form1.CheckBox2Click(Sender);

  Form1.LabeledEdit21Change(Sender);
  Form1.LabeledEdit22Change(Sender);

end;

procedure TForm1.Button4Click(Sender: TObject);
var a,b,c,d,x:ArrMax; i,N:integer; aa,bb,cc:real;
begin

end;

procedure TForm1.CheckBox1Click(Sender: TObject);
var dx2,CC,GG,MM,YY,Epsilon,Sigma2,SigmaE2:real;
begin
  dx2:=StrToFloat(LabeledEdit14.Text);
  CC:=StrToFloat(LabeledEdit5.Text);
  GG:=StrToFloat(LabeledEdit6.Text);
  MM:=StrToFloat(LabeledEdit7.Text);
  YY:=StrToFloat(LabeledEdit8.Text);
  Epsilon:=StrToFloat(LabeledEdit11.Text);
  Sigma2 :=StrToFloat(LabeledEdit13.Text)*
    StrToFloat(LabeledEdit13.Text);
  SigmaE2:=SigmaEpsSquare(Epsilon,dx2,CC,GG,MM,YY);
  LabeledEdit1.Text:=FloatToStr(5*sqrt(Sigma2+SigmaE2));
end;
```

```
procedure TForm1.CheckBox2Click(Sender: TObject);
var dx:real;
begin
  dx:=StrToFloat(LabeledEdit21.Text);
  LabeledEdit11.Text:=FloatToStr(4*dx);
end;

procedure TForm1.LabeledEdit1Change(Sender: TObject);
var N:integer; Abig:real;
begin
  if CheckBox2.Checked then CheckBox2Click(Sender);
end;

procedure TForm1.LabeledEdit12Change(Sender: TObject);
var N:integer; Abig:real;
begin
  N :=StrToInt(LabeledEdit12.Text);
  Abig:=StrToFloat(LabeledEdit1.Text);
end;

procedure TForm1.LabeledEdit2Change(Sender: TObject);
var M:integer; T:real;
begin
  M :=StrToInt(LabeledEdit2.Text);
  T:=StrToFloat(LabeledEdit10.Text);
  LabeledEdit22.Text:=FloatToStr(T/M);
end;

procedure TForm1.LabeledEdit10Change(Sender: TObject);
var M:integer; T:real;
begin
  M :=StrToInt(LabeledEdit2.Text);
  T:=StrToFloat(LabeledEdit10.Text);
  LabeledEdit22.Text:=FloatToStr(T/M);
end;

procedure TForm1.LabeledEdit21Change(Sender: TObject);
var
  Abig,Sigma,SigmaEps2,dx,dx2,CC,GG,MM,YY,Epsilon:real;
  u:Arr;
```

```
begin
  Form1.CheckBox2Click(Sender);

  Form1.CheckBox1Click(Sender);

  dx:=StrToFloat(LabeledEdit21.Text);
  Abig:=StrToFloat(LabeledEdit1.Text);
  LabeledEdit12.Text:=IntToStr(round(2*Abig/dx));
end;

procedure TForm1.LabeledEdit22Change(Sender: TObject);
var T,dt:real;
begin
  T:=StrToFloat(LabeledEdit10.Text);
  dt:=StrToFloat(LabeledEdit22.Text);
  LabeledEdit2.Text:=IntToStr(round(T/dt));
end;

procedure TForm1.LabeledEdit24Change(Sender: TObject);
begin
  Form1.Button1Click(Sender);
end;

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
  LabeledEdit5.Text:=FloatToStr(6.25);
  LabeledEdit6.Text:=FloatToStr(14.4);
  LabeledEdit7.Text:=FloatToStr(60.2);
  LabeledEdit8.Text:=FloatToStr(0);
end;

procedure TForm1.RadioButton8Click(Sender: TObject);
begin
  LabeledEdit5.Text:=FloatToStr(0.0502);
  LabeledEdit6.Text:=FloatToStr(0.0029);
  LabeledEdit7.Text:=FloatToStr(0.0031);
  LabeledEdit8.Text:=FloatToStr(0.1079);
end;

procedure TForm1.RadioButton1Click(Sender: TObject);
begin
```

```
  LabeledEdit5.Text:=FloatToStr(0.043);
  LabeledEdit6.Text:=FloatToStr(0.003);
  LabeledEdit7.Text:=FloatToStr(0.0031);
  LabeledEdit8.Text:=FloatToStr(0.0912);
end;

procedure TForm1.RadioButton7Click(Sender: TObject);
begin
  LabeledEdit5.Text:=FloatToStr(0.0422);
  LabeledEdit6.Text:=FloatToStr(0.0128);
  LabeledEdit7.Text:=FloatToStr(0.0131);
  LabeledEdit8.Text:=FloatToStr(0.0895);
end;

procedure TForm1.RadioButton3Click(Sender: TObject);
begin
  LabeledEdit20.Enabled:=false;
  LabeledEdit25.Enabled:=false;
end;

procedure TForm1.RadioButton4Click(Sender: TObject);
begin
  LabeledEdit25.Enabled:=false;
end;

procedure TForm1.RadioButton11Click(Sender: TObject);
begin
  LabeledEdit26.Enabled:=true;
  LabeledEdit27.Enabled:=true;
end;

procedure TForm1.RadioButton12Click(Sender: TObject);
begin
  LabeledEdit26.Enabled:=true;
  LabeledEdit27.Enabled:=false;
end;

procedure TForm1.RadioButton9Click(Sender: TObject);
begin
  LabeledEdit26.Enabled:=false;
  LabeledEdit27.Enabled:=false;
```

```
end;

procedure TForm1.RadioButton10Click(Sender: TObject);
begin
  LabeledEdit26.Enabled:=false;
  LabeledEdit27.Enabled:=false;
end;


end.
```