

**Raport Badawczy**

**RB/39/2001**

**Research Report**

**Metody Monte-Carlo  
w wycenie instrumentów  
pochodnych**

**Piotr Nowak, Piotr Nycz,  
Maciej Romaniuk**

**Instytut Badań Systemowych  
Polska Akademia Nauk**

**Systems Research Institute  
Polish Academy of Sciences**



# **POLSKA AKADEMIA NAUK**

## **Instytut Badań Systemowych**

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 8373578

fax: (+48) (22) 8372772

Pracę zgłosił: dr Piotr Nowak, inż. Piotr Nycz  
mgr Maciej Romaniuk

Warszawa 2001

**INSTYTUT BADAŃ SYSTEMOWYCH PAN**

Numer (symbol pracowni/nr kolejny opracowania/rok)  
PWDwWR/...../2001

**Piotr Nowak  
Piotr Nycz  
Maciej Romaniuk**

Tytuł opracowania:     *Metody Monte – Carlo w wycenie instrumentów pochodnych*

Nazwa Pracowni:     *Pracownia Metod Stochastycznych*

Kierownik Pracowni:     *prof. dr hab. Olgierd Hryniewicz*

Nazwa zadania:     *„Ilościowe metody zarządzania ryzykiem finansowym”*

Nazwa ewentualnego podzadania: -

Kierownik zadania:     *doc. dr hab. Dariusz Gątarek*

Wykonawcy (zespół):     *Piotr Nowak, Piotr Nycz, Maciej Romaniuk*

Warszawa, grudzień 2001 r



# Metody Monte-Carlo w wycenie pochodnych instrumentów finansowych

Piotr Nowak

Piotr Nycz

Maciej Romaniuk

Grudzień 2001

## 1 Wstęp

Teoretyczna wycena pochodnych instrumentów finansowych jest często trudna do realizacji. W szczególności dotyczy to opcji typu amerykańskiego oraz opcji egzotycznych. W tej sytuacji z pomocą przychodzi m. in. symulacja Monte-Carlo pozwalająca na uzyskanie przybliżonych cen wymienionych instrumentów przy użyciu generatora liczb pseudolosowych.

Podczas wykonywania symulacji pojawiają się dwa rodzaje błędów: błędy estymacji oraz błędy dyskretyzacji. Jednym z podstawowych zadań związanych z wykorzystywaniem symulacji Monte-Carlo jest próba redukcji powyższych błędów. Jako przykładowa metoda redukująca błędy estymacji omówiona zostanie metoda odbić lustrzanych (antithetic). W ostatnim czasie dużym powodzeniem cieszy się również metoda importance sampling, która będzie

przedmiotem naszych dalszych zainteresowań. Błędy dyskretyzacji omówione w paragrafie 3.2 wiążą się z przejściem od przypadku procesu ceny akcji z czasem ciągłym ((1), paragraf 2.2) do jego dyskretyzacji ((2), paragraf 3.1).

Istotną częścią naszych badań jest program symulacyjny *Option Pricer*, który posługuje się optymalnymi pod względem czasowym algorytmami, korzystając równocześnie z najnowszych osiągnięć światowych z dziedziny generatorów liczb pseudolosowych.

## 2 Wycena pochodnego instrumentu finansowego

Niech  $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t \in [0, T]}, P)$  będzie przestrzenią probabilistyczną z filtracją generowaną przez ruch Browna  $(B_t)_{t \in [0, T]}$  i zbiory miary  $P$  zero. Rozważamy model z czasem ciągłym ze skończonym horyzontem czasowym ( $T < \infty$ ).

Przez  $(S_t)_{t \in [0, T]}$  oznaczamy cenę instrumentu podstawowego. W ogólnym przypadku wymaga się, aby  $(S_t)_{t \in [0, T]}$  był semimartyngałem. Dla potrzeb tego artykułu założymy, że proces ten jest procesem Ito opisanym równaniem stochastycznym

$$dS_t = \mu_t dt + \sigma_t dB_t,$$

gdzie  $\mu_t$  i  $\sigma_t$  są  $\mathcal{F}_t$ -adaptowanymi procesami spełniającymi  $P$ -p.n. warunki:

$$\int_0^t |\mu_s| ds < \infty \quad \text{i} \quad \int_0^t \sigma_s^2 ds < \infty \quad \text{dla każdego } t \in [0, T].$$

Niech zmienna losowa  $X$  oznacza wypłatę końcową pochodnego instrumentu finansowego o terminie wygaśnięcia  $T$ . W bogatej literaturze z zakresu matematyki finansowej (patrz: (1999, Weron, Weron)) opisana jest ogólna

metoda arbitrażowej (zakładającej brak zysku bez ryzyka) wyceny pochodnego instrumentu finansowego. Metoda ta zakłada następujące kroki.

1. Znalezienie miary  $Q$  równoważnej mierze  $P$ , względem której zdyskontowany proces ceny instrumentu podstawowego  $Z_t = \Lambda_t^{-1} S_t$  jest martyngałem. ( $\Lambda_t = \exp(rt)$ ).
2. Obliczenie wartości  $(V_t)_{t \in [0, T]}$  instrumentu pochodnego zgodnie ze wzorem

$$V_t = \Lambda_t E^Q \left( \Lambda_T^{-1} X | \mathcal{F}_t \right).$$

## 2.1 Model Blacka-Scholesa

Zakładamy, że cena akcji opisana jest równaniem stochastycznym postaci

$$dS_t = \left( \mu + \frac{1}{2} \sigma^2 \right) S_t dt + \sigma S_t dB_t,$$

Wówczas proces ceny akcji

$$S_t = S_0 \exp(\sigma B_t + \mu t),$$

będący rozwiązaniem powyższego równania nazywany jest geometrycznym ruchem Browna. Zakładamy także, że wolna od ryzyka stopa procentowa  $r$  jest stała na odcinku czasowym  $[0, T]$ , akcje są podzielne i nie przynoszą dywidend. Nie uwzględniamy także kosztów transakcji i podatków.

Istnienie równoważnej miary martyngałowej wynika z następującego twierdzenia Girsanowa.

**Twierdzenie 1.** *Jeżeli  $(B_t)_{t \in [0, T]}$  jest ruchem Browna względem miary  $P$ , a  $(\gamma_t)_{t \in [0, T]}$  jest procesem  $\mathcal{F}_t$  - prognozowalnym, spełniającym warunek*

$$E^P \left( \frac{1}{2} \int_0^T \gamma_t^2 dt \right) < \infty,$$

to istnieje miara  $Q$  taka, że

- $Q$  jest równoważne mierze  $P$ ,
- $\frac{dQ}{dP} = \exp\left(-\int_0^T \gamma_t dB_t - \frac{1}{2} \int_0^T \gamma_t^2 dt\right)$ ,
- $\tilde{B}_t = B_t + \int_0^T \gamma_s dB_s$  jest ruchem Browna względem  $Q$ .

## 2.2 Wycena europejskiej opcji kupna

W przypadku europejskiej opcji kupna  $X = f_T = (S_T - K)^+$ .

Stąd

$$V_t = e^{-r(T-t)} E^Q \left( (S_T - K)^+ | \mathcal{F}_t \right).$$

Z twierdzenia Girsanowa wynika, że

$$S_t = S_0 \exp \left( \sigma \tilde{B}_t + \left( r - \frac{1}{2} \sigma^2 \right) t \right). \quad (1)$$

Po przekształceniach otrzymujemy wzór na cenę opcji w chwili  $t$

$$V_t = S_t \Phi \left( \frac{\ln \left( \frac{S_t}{K} \right) + \left( r + \frac{\sigma^2}{2} \right) (T-t)}{\sigma \sqrt{T-t}} \right) - K e^{-r(T-t)} \Phi \left( \frac{\ln \left( \frac{S_t}{K} \right) + \left( r - \frac{\sigma^2}{2} \right) (T-t)}{\sigma \sqrt{T-t}} \right).$$

## 3 Wyznaczanie ceny opcji metodą Monte-Carlo

Standartową metodą wyceny opcji, oprócz wykorzystania wzoru Blacka-Scholesa i jego pochodnych, jest wykorzystywanie symulacji zwanej metodą Monte-Carlo (w skrócie określanej dalej jako metoda MC). Metoda ta ma



szczególne szerokie zastosowanie w znajdowaniu wartości dla opcji egzotycznych. W przypadku naszego programu metodę tą wykorzystujemy jak na razie tylko do wyceny opcji europejskich, ale przewidujemy rozszerzenie jego możliwości na wykonywanie symulacji metodą MC dla innych typów opcji.

### 3.1 Symulacja trajektorii akcji metodą MC

W tym paragrafie przedstawimy pokrótce sposób znajdowania ceny opcji metodą MC oraz związek pomiędzy tą metodą a wzorem Blacka-Scholesa (w skrócie zwanego dalej wzorem BS).

Najważniejszym elementem w użyciu metody MC do wyznaczania ceny dowolnej opcji jest przeprowadzenie symulacji trajektorii zachowania się ceny akcji w czasie aż do momentu zapadania opcji (oznaczanego wcześniej jako  $T$ ). Ponieważ symulacja ta wykonywana jest na komputerze (o skończonej przestrzeni pamięci), wymagana jest dyskretyzacja badanego odcinka czasu  $[0, T]$  w zbiór punktów czasowych  $\{t_0 = 0, t_1, t_2, \dots, t_n = T\}$ . Liczba punktów czasowych  $n$  będzie dalej określana jako ilość kroków symulacji.

Przybliżenie geometrycznego ruchu Browna  $\mathcal{B}_t$  uzyskujemy wykorzystując tzw. *schemat Eulera*:

$$S_k = S_{k-1} \exp \left[ \left( r - \frac{\sigma^2}{2} \right) (t_k - t_{k-1}) + \epsilon_k \sigma \sqrt{t_k - t_{k-1}} \right] \quad (2)$$

gdzie  $S_i$  jest wartością przyjmowaną przez trajektorię ceny akcji w momencie  $t_i$  (przy czym wartość  $S_0$  jest zadana przez warunki brzegowe zadania – patrz paragraf 2), a  $\epsilon_k$  jest niezależną zmienną losową z rozkładu  $N(0,1)$  (tzn.  $\epsilon_1, \epsilon_2, \dots, \epsilon_n \stackrel{\text{iid}}{\sim} N(0,1)$ ). Pozostałe symbole są analogiczne do używanych w paragrafie 2.1.

Po uzyskaniu w przedstawiony powyżej sposób zbioru wartości trajektorii  $\{S_0, \dots, S_n\}$  obliczamy dla przeprowadzonej symulacji wartość funkcji wypłaty (przy czym postać tej funkcji zależy od typu rozważanej opcji). Funkcję tą będziemy oznaczać jako  $f(S_0, \dots, S_n)$  (pomijając indeks zależności od typu opcji).

Wykorzystując powyższą metodę  $N$  razy (liczbę  $N$  nazywać będziemy dalej ilością trajektorii) otrzymujemy  $N$  wartości funkcji  $f^i(S_0^i, \dots, S_n^i)$ , gdzie indeks górny  $i$  oznacza dane uzyskane z symulowania  $i$ -tej trajektorii (gdzie  $i = 1, \dots, N$ ). Wyliczona z nich średnia jest estymatorem poszukiwanej ceny opcji:

$$\tilde{C}^* = \frac{1}{N} \sum_{i=1}^N f_{(K,T)}^i(S_0^i, \dots, S_n^i) \quad (3)$$

$$\tilde{C} = e^{-rT} \tilde{C}^* \quad (4)$$

gdzie  $\tilde{C}$  oznacza poszukiwany (zdyskontowany) estymator.

Chcielibyśmy w tym miejscu zwrócić uwagę na teoretyczne podstawy umożliwiające wyprowadzenie wzoru (3), które, jak się nam wydaje, nie są dostatecznie jasno wyłożone w dostępnej literaturze. Przede wszystkim należy podkreślić znaczenie schematu Eulera (wzór (2)) w procesie przejścia do równoważnej miary martyngałowej i możliwości stosowania metody MC – cała istotna matematycznie informacja na ten temat zawarta jest właśnie w tym wzorze, a w szczególności założenie o braku arbitrażu w każdym momencie czasowym. W jasny sposób wynika stąd, że wykorzystanie metody MC do innych typów opcji (również egzotycznych), wymaga jedynie odmiennej funkcji wypłaty. Jednocześnie próba zasymulowania innego procesu stochastycznego obrazującego ruch cen akcji niż geometryczny ruch Browna (a więc wykorzystanie do tego celu innego martyngału) wymagać będzie stworzenia

odpowiednika wzoru (2), co może stanowić istotny problem matematyczny i symulacyjny.

Z kolei wzór (3) wynika z podstawowej teorii związanej z metodami MC i jest dość prostym przełożeniem na ten język MPWL. Tak więc próba zmiany procesu stochastycznego nie będzie mieć żadnego wpływu na stosowalność tego wzoru.

### 3.2 Miary błędu metody MC

Najpowszechniej w literaturze dotyczącej użycia metody MC dla celów wyceny opcji stosowane są dwie miary błędu estymacji (dla uproszczenia zwane dalej w tym paragrafie po prostu błędami).

Pierwszą z nich jest *błąd względny* pomiędzy estymatorem uzyskanym metodą MC (oznaczanym w (4) jako  $\tilde{C}$ ) a  $C$  – ceną wyliczoną przy wykorzystaniu teoretycznego wzoru (np. przy pomocy wzoru BS dla opcji europejskich – patrz ods!):

$$\text{Err}_R = \left| \frac{C - \tilde{C}}{C} \right| \quad (5)$$

Powyższy wzór wykorzystywany jest szczególnie często do porównywania jakości symulacji różnych implementacji metod MC (patrz np. (1997, Galanti, Jung)) i z tego też powodu został zaimplementowany w naszym programie. Niestety, nie jest do końca jasnym dlaczego w literaturze owa wartość jest podawana jako pewna stała deterministyczna (1997, Galanti, Jung) – wielkość  $\text{Err}_R$  jest przecież zmienną losową i zmienia się pomiędzy symulacjami. Brak jest nam na razie przesłanek na temat rozwiązania tej sprzeczności (dokładne teoretyczne obliczenie wartości  $\text{Err}_R$  jest z oczywistych powodów zbyt skomplikowane, aby było ono stosowane).

Drugą miarą błędu jest błąd średniokwadratowy, a ściślej rzecz biorąc – wariancja. Należy w tym miejscu podkreślić (co znowu nie jest wyraźnie zaznaczone w dostępnej nam literaturze), iż *możliwe* jest obliczanie dwóch wariancji dla metody MC zastosowanej do obliczania ceny opcji. Po pierwsze wielkość  $\text{Err}_R$ , jak wspomniano powyżej, jest zmienną losową. Stąd możliwe jest rozważanie wielkości  $\text{Var Err}_R$  – wariancji błędu względnego. Ze względu na problemy z teoretycznym szacowaniem tej wielkości, można wprowadzić związany z nią estymator – dla określonej liczby  $m$  powtórzeń całych „pakietów” symulacji (tzn. po przeprowadzeniu  $m$  razy po  $N$  symulacji w sposób przedstawiony w paragrafie 3.1) możemy obliczyć wyrażenie:

$$\text{Var}_{Ez}^{\sim} = \frac{1}{m-1} \sum_{i=1}^m (\tilde{C}_i - C)^2 \quad (6)$$

gdzie  $\tilde{C}_i$  jest estymatorem ceny opcji uzyskany w  $i$ -tym powtórzeniu pakietu  $N$  symulacji. Wartość  $\text{Var}_{Ez}^{\sim}$  pozwoliliśmy sobie określić jako estymator *wariancji zewnętrznej*, tj. wariancji pomiędzy wartością teoretyczną ceny opcji a jej estymatorem uzyskany metodą MC.

Drugim współczynnikiem wariancji jest wielkość błędu średniokwadratowego pomiędzy funkcją wypłaty dla pojedynczej trajektorii a poszukiwanym estymatorem (niezdyskontowanym):

$$E(f(S_0, \dots, S_n) - \tilde{C}^*)^2 \quad (7)$$

Ponieważ  $E(f(S_0, \dots, S_n)) = \tilde{C}^*$ , stąd wzór (7) określiliśmy jako *wariancję wewnętrzną*. Podobnie jak poprzednio, teoretyczne obliczenie tej wielkości jest zbyt skomplikowane dla praktycznych zastosowań. Stąd potrzeba wprowadzenia jej estymatora, który można łatwo zapisać przy pomocy wzoru:

$$\text{Var}_{In}^{\sim} = \frac{1}{N-1} \sum_{i=1}^N (f^i(S_0^i, \dots, S_n^i) - \tilde{C}^*)^2 \quad (8)$$

Jak się wydaje, w dostępnej literaturze pod uwagę brana jest właśnie ta wariancja (patrz np. (1999, Weron, Weron). Dlatego też wartość wzoru (8) jest obliczana w naszym programie.

### 3.3 Błędy dyskretyzacji w metodzie Monte-Carlo

Przy wycenie opcji, zadaniem naszym jest wyznaczenie numerycznej wartości  $E(f(X_T))$ , gdzie

$$f : R \rightarrow R,$$

a  $X$  jest rozwiązaniem równania

$$dX_t = a(X_t, t) dt + b(X_t, t) dB_t, \quad X_0 = x.$$

Jedną z możliwych aproksymacji zmiennej  $X_T$  jest aproksymacja jej wartości zmienną losową  $\hat{X}_{T/h}^h$  określoną rekurencyjną formułą Eulera (por. (2))

$$\hat{X}_{k+1}^h - \hat{X}_k^h = a(\hat{X}_k^h, kh)h + b(\hat{X}_k^h, kh)\sqrt{h}\varepsilon_{k+1}, \quad \hat{X}_0^h = x,$$

gdzie  $\varepsilon_1, \varepsilon_2, \dots$  są niezależnymi zmiennymi losowymi o standardowym rozkładzie normalnym, a  $h$  jest przyrostem czasu. Niech

$$f_h = E\hat{X}_{T/h}^h \text{ i } e_h = E(f(X_T)) - f_h,$$

$e_h$  oznacza błąd dyskretyzacji.

Wówczas formuła Eulera daje zbieżność pierwszego stopnia, tzn.  $\frac{e_h}{h}$  jest ograniczoną funkcją zmiennej  $h$ , a przy dodatkowych założeniach o regularności (jeżeli  $a, b, f$  są gładkie i  $f$  ma wzrost wielomianowy) istnieje  $C$  takie, że  $\frac{e_h - Ch}{h^2}$  jest ograniczoną funkcją  $h$  (1995, Duffie, Glynn). W szczególności, dla europejskiej opcji kupna dostajemy zbieżność pierwszego stopnia.

## 4 Poprawianie działania metody MC

Jednym z interesujących nas problemów (i często rozważanym w literaturze) jest poprawa szybkości i jakości estymatorów znajdujących za pomocą metod MC. Uzyskanie tych celów jest możliwe m.in. poprzez metody redukcji wariancji oraz stosowanie odpowiednich algorytmów generowania liczb pseudolosowych.

### 4.1 Redukcja wariancji metodą „antithetic”

Jak już wspomniano, wariancja jest jedną z miar błędu (patrz paragraf 3.2). Jej redukcja wpływa zaś na zmniejszenie błędu względnego – najważniejszej praktycznej miary błędu, co wielokrotnie obserwowaliśmy podczas działania naszego programu.

Zmniejszenie wielkości wariancji możliwe jest m.in. przy pomocy metody „antithetic” (w polskiej terminologii spotyka się określenie „metoda zmiennych antytetycznych”, „odbić lustrzanych” lub ogólniej „zmiennych skorelowanych”). Głównym założeniem metody „antithetic” jest wprowadzenie korelacji pomiędzy dwiema symulacjami metodą MC, tak aby wariancja sumy zmiennych zależnych była mniejsza od sumy wariancji zmiennych niezależnych. Czy będzie to możliwe zależy oczywiście od znaku korelacji pomiędzy symulacjami.

W naszym programie została zaadaptowana metoda polegająca na tym, iż po zasymulowaniu jednej pełnej trajektorii (patrz paragraf 3.1), druga tworzona jest z już wylosowanych wartości z rozkładu  $N(0,1)$ , ale z odwrotnym znakiem (innymi słowy mamy  $\epsilon_k^i = -\epsilon_k^{i+\frac{N}{2}}$ , gdzie indeks górny oznacza numer trajektorii oraz  $k = 1, \dots, n$ ,  $i = 1, \dots, \frac{N}{2}$ ). Oznacza to, że wykonywana jest

tylko połowa losowań z rozkładu normalnego (patrz (1999, Weron, Weron), (1995, Boyle, Broadie, Glasserman)).

W naszym programie obliczana jest dla metody „antithetic” wariancja (patrz wzór (8)), oczywiście z modyfikacją wynikającą z zależności pomiędzy symulacjami. Ta modyfikacja przyjmuje u nas postać estymatora kowariancji w postaci:

$$\tilde{\text{Cov}}_{In} = \frac{1}{\frac{N}{2} - 1} \sum_{i=1}^{\frac{N}{2}} (X_i - \tilde{C}^*_X)(Y_i - \tilde{C}^*_Y) \quad (9)$$

gdzie  $X$  i  $Y$  oznaczają grupy funkcji wypłat dla symulacji niezależnych (w tej grupie) trajektorii i obliczone na ich podstawie estymatory. Dokładniej rzecz ujmując mamy:

$$X_i = (f^i(\mathcal{S}_0^i, \dots, \mathcal{S}_n^i)) \quad (10)$$

$$Y_i = (f^{i+\frac{N}{2}}(\mathcal{S}_0^{i+\frac{N}{2}}, \dots, \mathcal{S}_n^{i+\frac{N}{2}})) \quad (11)$$

$$\tilde{C}^*_X = \frac{2}{N} \sum_{i=1}^{\frac{N}{2}} X_i \quad \tilde{C}^*_Y = \frac{2}{N} \sum_{i=1}^{\frac{N}{2}} Y_i \quad (12)$$

Jak wykazały przeprowadzone przez nas symulacje, zgodnie z teoretycznymi założeniami, otrzymywaliśmy wartość  $\tilde{\text{Cov}}_{In}$  poniżej zera i dużą redukcję błędu względnego (patrz wzór (5)) w porównaniu do metody MC bez redukcji wariancji (patrz paragraf 3.1).

## 4.2 Algorytmy generowania liczb pseudolosowych

Przyspieszenie i \ lub poprawienie jakości symulacji jest również możliwe przez zmiany w procesie generowania  $\epsilon_k$  (patrz (2)). Można tego dokonać w dwóch kluczowych miejscach: przy generowaniu wartości z rozkładu  $U(0,1)$  oraz przy zmianie tych wartości na zmienne uzyskiwane z rozkładu  $N(0,1)$ .

Optymalizacja procedury dla pierwszego punktu wymaga użycia szybszych i „lepiej rozłożonych” (tzn. o większym okresie i większej losowości) algorytmów generowania dla rozkładu jednostajnego. W przypadku naszego programu użyte zostały następujące algortymy: standardowy (pochodzący z implementacji C++ firmy Borland), algorytm ULTRA (1997, Wieczorkowski, Zieliński, s. 31) napisany w C++ oraz generator japoński (patrz <http://www.math.keio.ac.jp/~matumoto/emt.html>).

Wielość wykorzystanych przez nas algorytmów ma na celu znalezienie najbardziej optymalnego z nich, tzn. porównanie ich właściwości na polu wyliczania cen opcji. Dokładniejsze szczegóły dotyczące implementacji podane zostały w paragrafie 5.2.

Jak się wydaje z dotychczas przeprowadzonych prób najskuteczniejszym generatorem jest generator japoński, jednak dalsze próby będą kontynuowane. Niewykluczamy też zastosowania generatorów nielosowych (tzn. implementację metody quasi – Monte Carlo).

Z kolei do transformacji wartości wylosowanej z rozkładu  $U(0,1)$  (przy pomocy jednego z poprzednich algorytmów) do zmiennej pochodzącej z  $N(0,1)$  użyto algorytmu Marsaglia. W algorytmie tym dwie niezależne zmienne losowe z  $U(0,1)$  przeprowadza się na dwie niezależne zmienne losowe z  $N(0,1)$ . Trwają właśnie prace nad wprowadzeniem drugiej możliwości i użyciem metody odwrotnej dystrybuanty Moro (1995, Moro). Metoda ta uznawana jest w literaturze za znacznie szybszą, jednak mamy zamiar przeprowadzić odpowiednie symulacje. Ponadto algorytm Marsaglia prawdopodobnie zostanie zaadaptowany do losowania z przedziałów ufnosci.



## 5 Program wyceny opcji metodami Monte-Carlo

Program Option Pricer jest przeznaczony do przeprowadzania symulacji komputerowych związanych z wyceną tzw. pochodnych instrumentów wartościowych - opcji. Został zaimplementowany w systemie Borland C++ Builder wersji 5.0 działającym na platformach Microsoftu.

Zapotrzebowania sprzętowe wymagane do pracy programu są niewielkie, jednak należy pamiętać, że to komputer na jakim przeprowadzane są symulacje wyznacza postęp przeprowadzanych badań.

### 5.1 Funkcje i komendy programu

Program po uruchomieniu tworzy okno zawierające menu służące do sterowania programem.

#### 5.1.1 Metoda symulacji

Obliczenia mogą być przeprowadzane z użyciem dwóch metod:

1. Algorytmu opartego o metody Monte - Carlo. (bez redukcji wariancji)
2. Algorytmu opartego o redukcję wariancji metodą odbić lustrzanych (patrz paragraf 4.1).

Wyboru metody przeprowadzania symulacji dokonuje się przez wydanie odpowiedniej komendy z głównego menu programu – **Options**.

### 5.1.2 Generator liczb pseudolosowych

Jakość wyników otrzymanych z symulacji Monte - Carlo w znacznej mierze zależy od generatora liczb pseudolosowych, i dlatego program umożliwia wybór jednego z trzech generatorów:

1. Standardowy generator dostarczany przez firmę Borland
2. Generator ULTRA.
3. Generator japoński.

### 5.1.3 Ustalenie parametrów początkowych

W celu ustanowienia początkowych parametrów symulacji należy wybrać z menu Option komendę: „Black-Scholes Model Option”. Pojawi się okno dialogowe, które umożliwi wprowadzenie odpowiednich wartości.

### 5.1.4 Uruchomienie symulacji

Gdy wszystkie wartości początkowe zostaną wprowadzone wówczas można przystąpić do symulacji. W celu dokonania jej startu należy z menu Control wybrać menu Control Panel. Po wydaniu tej komendy pojawi się okno dialogowe, które umożliwi za pomocą odpowiednich przycisków rozpocząć lub przerwać symulację oraz zapisać jej wyniki do pliku tekstowego. Opisywane okno dialogowe pokazano poniżej. Wyniki symulacji oraz informacje o stanie programu prezentowane są w głównym oknie Control, natomiast informacje dotyczące przeprowadzanej symulacji prezentowane są w mniejszym oknie poniżej.

### **5.1.5 Wykonywanie długotrwałych obliczeń**

Podczas wykonywania długotrwałych obliczeń - tj. gdy liczba kroków algorytmu symulacyjnego jest duża np. 1000 istnieje możliwość pracy programu w tle systemu operacyjnego. Efektem jest umożliwienie pracy użytkownikowi z innymi programami w czasie trwania symulacji. Stopień obciążenia komputera przez symulację można ustawiać suwakiem regulującym tzw. priorytet wątku wykonującego symulację. Istnieje 7 priorytetów wątku stopniowo zwiększających wykorzystywanie zasobów komputera przez symulację.

### **5.1.6 Zarządzanie projektami**

Biorąc pod uwagę fakt, że symulacje są czasochłonne i dla celów profesjonalnych są wykonywane dosyć często umożliwiono zapisywanie przeprowadzanych symulacji jako projektów, których realizację w każdej chwili można powtórzyć. Umożliwia to proste zarządzanie przeprowadzanymi symulacjami i pozwala porządkować zgromadzone wyniki. W celu utworzenia nowego projektu należy wybrać opcję File z menu oraz komendę Project Options. Z okna, które się pojawi istnieje możliwość sterowania wszystkimi innymi oknami programu.

## **5.2 Problemy związane z implementacją zagadnienia**

Podczas implementacji zagadnień symulacyjnych istotnym problemem jest wydajność tworzonego systemu. Czas oczekiwania na wyniki musi być możliwie krótki, ponieważ tylko wtedy użytkownik będzie mógł rozpatrywać wiele wariantów zagadnienia i szybko podejmować decyzje na podstawie danych wygenerowanych przez komputer. Z praktyki wynika, że w celu

otrzymania wiarygodnych wyników próby należy wykonywać wielokrotnie, co powoduje że czas wykonania pojedynczej symulacji staje się kluczowym parametrem oceny pracy tego typu systemów. Dlatego też oprogramowanie przeznaczone do symulacji jest poddawane nieustannym próbom szybkości.

Optymalizacji oprogramowania dokonuje się przez:

1. Stosowanie mniej czasochłonnych algorytmów.
2. Optymalizację zajętości pamięci komputera.
3. Umiejętną implementację stosowanego algorytmu

Podjmując problem do rozwiązania (w szczególności związany z symulacjami) algorytm rozwiązania jest niejako wymuszony przez rodzaj zagadnienia jaki wykonujemy i przeważnie wybiera się najlepszy ogólnie dostępny. Warto pokusić się więc o jak najlepsze wykorzystanie wydajności języków programowania. Poniżej opisano kilka szczegółów implementacyjnych związanych z symulacjami Monte-Carlo dotyczących wyceny pochodnych instrumentów wartościowych.

### 5.2.1 Precyzja danych używanych do obliczeń

Do implementacji użyto liczb typu *long double*. Jest to typ danych umożliwiający pracę z liczbami rzeczywistymi z zakresu:

$$3.37 * 10^{-4932} < |X| < 1.18 * 10^{4932} \quad (13)$$

Używając takiej reprezentacji danych uzyskujemy gwarancję bardzo wysokiej precyzji (18 liczb po przecinku) otrzymywanych wyników.

## 6 Cele dalszej działalności

W tym paragrafie w punktach wymienimy najważniejsze cele naszej dalszej planowanej działalności związanej z wykorzystaniem metod MC w wyznaczaniu ceny opcji oraz analizą ich skuteczności:

1. Próba wykorzystania algorytmów genetycznych do wyszukiwania miary minimalizującej wariancję w metodzie „important sampling”.
2. Rozszerzenie zakresu prac i możliwości programu na inne typy opcji.
3. Próba analizy wpływu losowości rozmieszczenia punktów na odcinku czasowym  $[0, T]$  na efektywność metody MC.
4. Próba zastąpienia geometrycznego ruchu Browna innym semimartyn-galem opisującym proces ceny akcji w czasie.

## A Bibliografia

P. Boyle, M. Broadie, P. Glasserman. „*Monte Carlo Methods and Security Pricing*”, Journal of Economics, 21, Nos. 8 – 9, pp. 1267 – 1322, 1997

D. Duffie, P. Glynn. „*Efficient Monte Carlo Simulation of Security Prices*”, Annals of Applied Probability, Vol. 5, No. 4, 1995

Galanti, Jung. „*Low Discrepancy Sequences: Monte Carlo Simulation of Option Prices*”, Journal of Derivatis, Fall 1997

B. Moro. „*The Full Monte*”, Risk, Vol.8, No.2, pp. 57 – 58 (Feb. 1995)

A. N. Shiryaev. „*Essentials of Stochastic Finance*”, WSP 1999

A. Weron, R. Weron. „*Inżynieria finansowa*”, WNT 1999

R. Wieczorkowski, R. Zieliński. „Komputerowe generatory liczb losowych”, WNT 1997

## Spis treści

<b>1</b>	<b>Wstęp</b>	<b>1</b>
<b>2</b>	<b>Wycena pochodnego instrumentu finansowego</b>	<b>2</b>
2.1	Model Blacka-Scholesa . . . . .	3
2.2	Wycena europejskiej opcji kupna . . . . .	4
<b>3</b>	<b>Wyznaczanie ceny opcji metodą Monte-Carlo</b>	<b>4</b>
3.1	Symulacja trajektorii akcji metodą MC . . . . .	5
3.2	Miary błędu metody MC . . . . .	7
3.3	Błędy dyskretyzacji w metodzie Monte-Carlo . . . . .	9
<b>4</b>	<b>Poprawianie działania metody MC</b>	<b>10</b>
4.1	Redukcja wariancji metodą „antithetic” . . . . .	10
4.2	Algorytmy generowania liczb pseudolosowych . . . . .	11
<b>5</b>	<b>Program wyceny opcji metodami Monte-Carlo</b>	<b>13</b>
5.1	Funkcje i komendy programu . . . . .	13
5.1.1	Metoda symulacji . . . . .	13
5.1.2	Generator liczb pseudolosowych . . . . .	14
5.1.3	Ustalenie parametrów początkowych . . . . .	14
5.1.4	Uruchomienie symulacji . . . . .	14
5.1.5	Wykonywanie długotrwałych obliczeń . . . . .	15
5.1.6	Zarządzanie projektami . . . . .	15

5.2	Problemy związane z implementacją zagadnienia . . . . .	15
5.2.1	Precyzja danych używanych do obliczeń . . . . .	16
<b>6</b>	<b>Cele dalszej działalności</b>	<b>17</b>
<b>A</b>	<b>Bibliografia</b>	<b>17</b>











