MONTE CARLO method for option pricing
Introduction

Problem Setting and Description

We explore the problem of pricing European vanilla options.

The problem of option pricing is one of the most famous and most well-studied problems in financial mathematics.

- Known Solutions
- Monte Carlo Approach
The structure of the Project Paper

Introduction of option and its types, talk about the reasons and why options are important

- Different scenarios through an example
- Monte Carlo Approach, Parallels with Black Scholes method
- Monte Carlo Numerical Solution
Theoretical Background

- Options and Option Pricing
- How options make money?
- Monte Carlo Method
- Black Scholes Method
In finance, an option is a financial contract, a security that gives right to buy/sell a specified financial asset.

Options are characterized by strike price and maturity.

**Call**
- Right to buy
- Profit: \[ \max(S-K); 0 \]

**Put**
- Right to sell
- Profit: \[ \max(K-S); 0 \]
How Options Make Money?

Yahoo stock is currently trading at 500\$

instructor purchases a call option 10\$ per share

let us discuss the following scenarions

- Yahoo stock is valued at 525\$ on the expiration date.
- Yahoo stock is valued at 505\$ on the expiration date.
- Yahoo stock is valued at 450\$ on the expiration date.
How Options Make Money?

(a) Payoff from buying a call option
How Options Make Money?

(b) Payoff from buying a put option
Why are options important?

security of limited risk and the advantage of leverage

Investors use options for two primary reasons:

speculation of the risk

there is no "sure thing," as every investment incurs at least some risk

hedging

protection against unforeseen events, insurance investors hope never to use

+ insurance = safety
Monte Carlo Methods are a broad class of computational algorithms that rely on repeated random sampling to obtain numerical results.

- to solve problems that might be deterministic in principle
- used in optimization, numerical integration, and generating draws from a probability distribution
Consider the probability of a particular sum of the throw of 2 dice

total number of 36 states

Hence, it can be manually computed the P of a particular outcome
The accuracy of MC is a function of the number of realizations.
Monte Carlo Simulation and Sampling

The terms simulation and sampling are used interchangeably.

The term sampling could be referred to those cases in which there is no dynamics to organize a simulation over time, whereas simulation results the generation of sample paths.

Simulation is used when the process consists of multiple steps.

- **American options**: intermediate states are necessary
- **European options**: intermediate states are not necessary
Assumptions of The Black–Scholes Model

- There is a riskless way of earning money

Interest rate of these riskless bonds is called the risk–free rate

- risk–free rate is constant

- price of a particular stock over a time period is a random walk with a constant drift

- stock does not pay any dividends

Some of these assumptions, for example constant risk–free rate and constant volatility, were later dropped

- volatility is the degree of variation of a trading price series over time as measured by the standard deviation of returns
value of a call option for a non-dividend-paying underlying stock

\[ C(t, S_t) = N(d_1)S_t - N(d_2)Ke^{-r(T-t)}, \]

\[ d_1 = \frac{1}{\sigma \sqrt{T-t}} [\ln(\frac{S_t}{K}) + (r + \frac{\sigma^2}{2})(T - t)], \quad d_2 = d_1 - \sigma \sqrt{T-t} \]

It can be used as a benchmark for our MC simulation because we will use the same assumptions as Black–Scholes model so.
NUMERICAL SOLUTION

− The Mechanics of Monte Carlo
− Monte Carlo Simulation
− Option Pricing Example using MC
The Mechanics of MC Simulation

- A dynamic evolution of a system over time can be described as a class of discrete-time, continuous-time, discrete-event.

- We actually simulate a discrete-time approximation, but given the issues involved in sample path generation.

Monte Carlo methods come into play when randomness is introduced into the differential equation, so it yields to SDE.

- A well known model is the Geometric Brownian Motion
In mathematical finance it is used to model stock prices in the MC model, Black–Scholes model etc.
Solving the SDE

If $W(t)$ is Weiner Process, then

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

$$dS_t = \mu(t, S_t) dt + \sigma(t, S_t) dW_t$$

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

As $\ln(S(t))$ follows a normal distribution, the stock price $S(t)$ follows a lognormal distribution with expectation value

$$E(S_t) = S_0 e^T$$
Monte Carlo Simulation

3.2.1 MC Process

We can assume that stock price follows GBM

\[ S_t = S_{t-1}(1 + \mu \Delta t) + \sigma \epsilon \sqrt{\Delta t} \]

Each step consists of a shift by the expected return and an upward or downward movement based on the vol. scaled by r.n..

- calculate the stock price at maturity
- calculate the payoff of the option based on the stock price
- discount the payoff at the riskfree rate to today’s price

The simulated results are compared and normalized to the values obtained Black Scholes Option Pricing method results.
Monte Carlo Simulation

stock price for day = 0...419

stock price
day

0 100 200 300 400 500
600 700 800 900 1000 1100
Let’s take a real-life European option and try to price it.

As of November 25 of 2016, Google’s stock is valued at $S = 761.68$. We want to buy a call option with strike price $K = 680$ which has an expiration date 19 January, 2018 (419 days).

It’s implied volatility, according to Yahoo Finance, is equal to 28.05%

$S=761.68, \; K=680, \; T=1,1479, \; vol=0,2805, \; r=0$
The result at the end:

| Monte Carlo simulation: 132.820882 |
| Black–Scholes formula: 132.949203 |

The Monte Carlo simulation was done with one million trials. It yields a value very close to the "right" one calculated by BS.

We obtained an option price close to its real life value
Option Pricing Example using MC

C++ and MATLAB Code Implementation

```matlab
stock_plot.m

S = 761.68;
vol = 0.3232;
T = 1.14794520548;
r = 0.04;
step = 419;
dt = T / step;
sqdt = sqrt(dt);
rr = stdnormal_rnd(1, step);

St = S;
stockPrices = [];
ts = [];
stockPrices(1) = S;
ts(1) = 0;

for st = 1:step
    St = St*(1 + r*dt + vol*sqdt*rr(1, st));
    stockPrices(st + 1) = St;
ts(st + 1) = st;
endfor

plot(ts, stockPrices)
title("stock price for day = 0...419");
xlabel("day");
ylabel("stock price");
```
Thanks for attention