

# Simulation Methods for Finance

## Barrier and Look-back Options

Espel, Kulak, MacIver, Qiu, Zhang

Imperial College London

March 10, 2018

# Outline

- 1 Random Variable Generation
- 2 European Call Option
- 3 Barrier Option
- 4 Barrier Option
- 5 Look-back Option
- 6 Using our Code

# Random Variable Generation - Methods

## Random

### **Generate Uniform Distribution**

System built-in Rand

Linear Congruential Generator

### **Convert U to SDN**

Central Limit Theory

Box-Muller

Marsaglia Polar

# Random Variable Generation - Analysis

- Rand returns a random number in  $(1, 2^{15} - 1)$ , may show correlation between numbers.
- CLT requires a sufficient large number of uniform distributions, the computational time is long.
- Box-Muller involves calculation of Sine Cosine and log. Computational time can be long.

# Random Variable Generation - Results

Generating 100 million random numbers

| Method                        | Mean      | Variance | Time  |
|-------------------------------|-----------|----------|-------|
| <b>rand + CLT</b>             | 1.77 e-5  | 0.999909 | 90.34 |
| <b>rand + Box-Muller</b>      | -6.80 e-5 | 1.000703 | 8.64  |
| <b>rand + Marsaglia Polar</b> | 1.70 e-5  | 1.000472 | 6.40  |
| <b>LCG + CLT</b>              | 1.52 e-5  | 0.999973 | 236.3 |
| <b>LCG + Box-Muller</b>       | -1.27 e-4 | 0.999797 | 11.59 |
| <b>LCG + Marsaglia</b>        | 7.36 e-5  | 0.999979 | 10.52 |
| <b>Random</b>                 | 3.58 e-4  | 1.000210 | 1.498 |

# European Call Option - Task

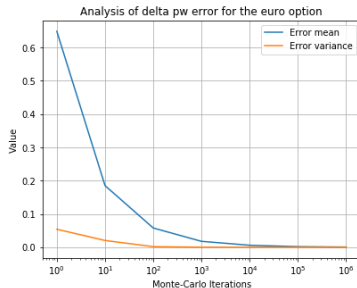
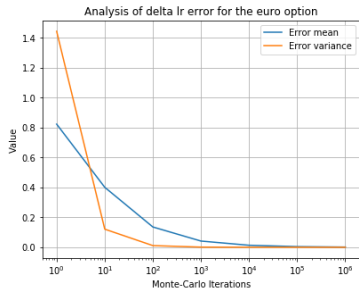
With the usual conventions, recall our model.

$$dS_t = rS_t dt + \sigma S_t dW_t, \quad 0 \leq t \leq T$$

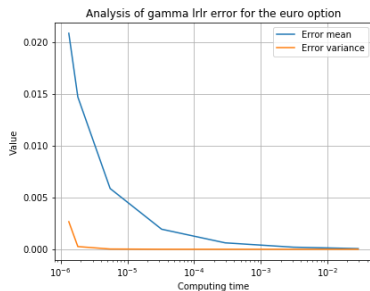
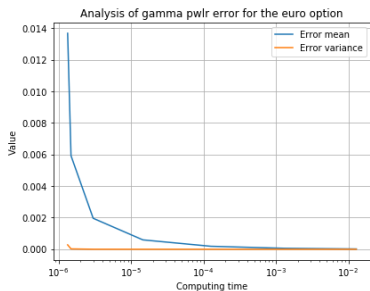
$$C_t = E[e^{-r(T-t)}(S_T - K)^+ | F_t]$$

- Create a module to simulate the results for the European option.
- Compute the Greeks (delta, gamma, vega) using different methods.
- Analyse the error compared to the theoretical result.

# European Call Option - Analysis



# European Call Option - Analysis





# European Call Option - Results

- There is a significant accuracy gap beyond 1,000 simulations.
- We based our conclusions on the industry standard: 100,000 simulations.
- There is a trade-off computation time/accuracy.

|                   | Error Mean | Error Variance | Time  |
|-------------------|------------|----------------|-------|
| <b>Delta LR</b>   | Worst      | Worst          | Best  |
| <b>Delta PW</b>   | Best       | Best           | Worst |
| <b>Gamma PWLR</b> | Worst      | Best           | Best  |
| <b>Gamma LRPW</b> | -          | Worst          | Best  |
| <b>Gamma LRLR</b> | -          | -              | Worst |
| <b>Vega LR</b>    | Worst      | -              | Worst |
| <b>Vega PW</b>    | Best       | -              | Best  |

# Barrier Option - Task

$$m_0^T = \min_{0 \leq t \leq T} S_t, \quad M_0^T = \max_{0 \leq t \leq T} S_t$$

For an up-and-out barrier call option,  $B > S_0$

$A_T = (S_T - K)^+ 1_{M_0^T < B}$ , where  $B$  is a barrier level and  $1_S$  is an indicator function.

# Barrier Option - Two Methods

1. Generate whole paths of  $S_t$  with 1,000 stops each path.  
Time used for \*\*\*simulation : \*\*\*seconds/minutes/hours

# Barrier Option - Two Methods

2. Use Rayleigh distribution generate the  $M_T$  directly.

The maximum of a standard Brownian motion starting at the origin to be at  $b$  at time 1 over period  $[0, T]$  has the Rayleigh distribution

$$F(x) = 1 - e^{-2x(x-b)}, \quad x \geq b.$$

Hence, at time  $T$  with  $S_T$ ,

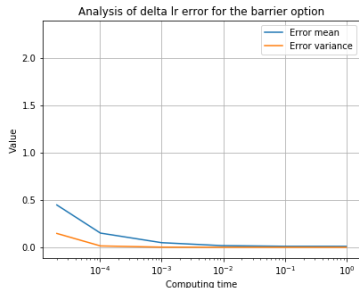
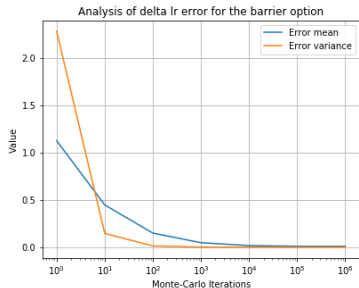
$$M^T = \frac{S_T + \sqrt{S_T^2 - 2T \log U}}{2}$$

Method 2 is 500 times faster!

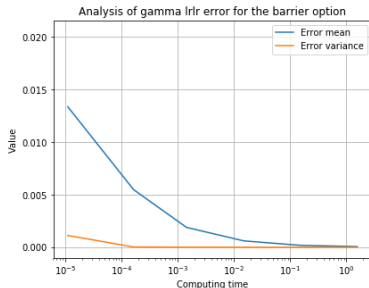
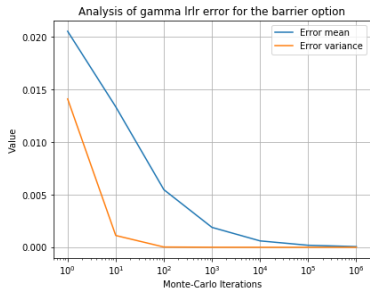
# Barrier Option - Analysis

Simulate Greeks using Likelihood ratio method as Pathwise method is not feasible for barrier option.

# Barrier Option - Analysis with a New Method



# Barrier Option - Analysis with a New Method



# Barrier Option - Results

|                   | Error Mean | Error Variance | Time (s) |
|-------------------|------------|----------------|----------|
| <b>Price</b>      | na         | na             | ..       |
| <b>Delta LR</b>   | 1.04 e-2   | 3.03 e-5       | 9.28 e-2 |
| <b>Gamma LRLR</b> | 1.89 e-4   | 1.89 e-8       | 1.48 e-1 |
| <b>Vega LR</b>    | 7.51 e-1   | 3.22 e-1       | 1.09 e-1 |

**Table:** Sample from our simulation dataset with the new fast method for barrier option simulation. It is clear that the results have significantly improved compared with the previous method. note that the run time depends on the computer used (here a 3.4 GHz Intel Core i7).



# Look-back Option - Task

Lookback call option with fixed strike price  $K$  has payoff  $(M_0^T - K)^+$ . The call option price at time  $t$  is

$$c(S_0, K, t) = e^{-r(T-t)} E[(\max(M_0^t, M_t^T) - K)^+ | \mathcal{F}_t]$$

# Look-back Option - Method

The cdf of the distribution of Maximum  $S_t$  for standard Brownian motion for the period of  $(0, T)$  is

$$F = \Phi\left(\frac{m - aT}{\sqrt{T}}\right) - e^{2am}\Phi\left(\frac{-m - aT}{\sqrt{T}}\right)$$

We used Newton-Raphson method to solve for the above cdf  $F = U$ . Hence, for each random number we generate from  $[0,1]$ , we receive one  $m$  in by solving for  $F$ . And to get  $M^T$  of stock price follows geometric Brownian motion with starting price  $S_0$ , we let  $M^T = S_0 e^{\sigma m}$ .

# Look-back Option - Analysis

# Look-back Option - Results

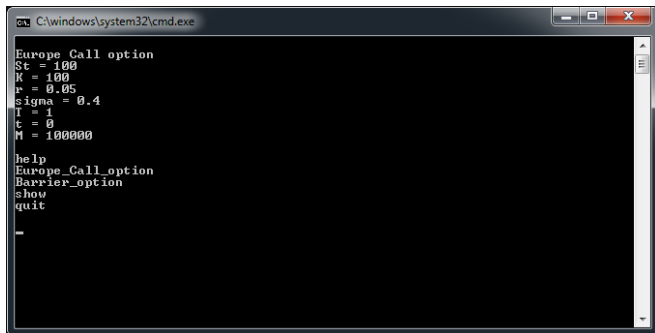
# Using our Code - C++ Package

- We have created a package for C++ users.
- Uses industry standards with dynamic library files.
- Package is very intuitive.

```
1  #include "BarrierLookBackOptions.h"
2
3
4  european_option call(S_t, K, r, s, T, t, iterations);
5
6  double price = call.price();
7  double gamma = call.gamma("lr");
8  double vega = call.vega("lr");
9
```

# Using our Code - "Code Free" terminal

- We have created an intuitive interface.
- This is suitable for users who do not code: you just have to type the command and there is a help mode.
- It supports european call option, barrier out call option and the look-back call option price and greeks.



```
C:\windows\system32\cmd.exe

Europe Call option
St = 100
K = 100
r = 0.05
sigma = 0.4
T = 1
t = 0
M = 100000

help
Europe_Call_option
Barrier_option
show
quit

-
```

# Conclusion

- We have created a module with the core task.
- The module also can also compute elements related to the barrier option and the look-back option.
- Our package is user-friendly and has a "code-free" interface.

# Thank you!

`github.com/tjespel/barrier-and-look-back-options`