High performance computing for options pricing using Monte Carlo Simulations

Kamesh K - MM16B107

March 19, 2021

Overview

Overview of Monte Carlo Simulations

Monte Carlo Simulations

About Monte Carlo Simulations

- Monte Carlo Simulations are used to model processes in which the outcomes have an associated probability and cannot be predicted due to the intervention of random variables
- Monte Carlo Simulations have different applications in the fields of mathematics, finance, engineering, supply chain and much more.
- Monte Carlo Simulations are computationally intensive and to minimize the error the number of simulations needs to be as high as possible

Example

The value of π can be estimated from a Monte Carlo simulation wherein we choose random points from a 1x1 square and compute the numerical probability that the point lies inside the inscribed circle relating it to actual probability = $\frac{\pi}{4}$

Simulation Results

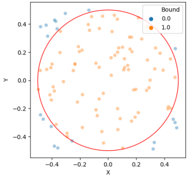

Figure: Simulation with N = 100
Estimated Value = 3.2

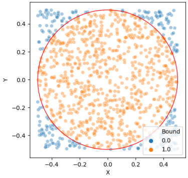

Figure: Simulation with N = 1000
Estimated Value = 3.076

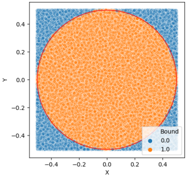

Figure: Simulation with N = 100000
Estimate Value = 3.13088

Error decreases with increase in iterations

Error Analysis

Confidence interval

- The error in case of a Monte Carlo simulation is measured in terms of the width of the confidence interval
- The confidence interval of a given distribution is measured in terms of the mean, variance(standard deviation) and number of samplings of the given distribution
- For a 95% confidence interval, we have the formula

$$CI = \mu \pm (1.96)\frac{\sigma}{\sqrt{N}}$$

- Where μ is the mean of the distribution, σ is the standard deviation and N is the number of samples, which is iterations in case of Monte Carlo simulations

Parallelizability of Monte Carlo Simulation

Need for parallelization

- As mentioned earlier in error calculation, the error or confidence interval is inversely proportional to the number of iterations
- If we are increasing the number of iterations, the system becomes more complex and requires more run time as computational complexity is increased
- To avoid the increase in runtime, we will have to parallelize the code if possible

Virtue of Monte Carlo Simulation

- In case of Monte Carlo simulations, the events are independent of each other
- Each iteration is independent of the other and hence there is a scope of parallelization possible
- Idea is to split the iterations required into different processes and gather them back to reconstruct the Monte Carlo Simulation required

Estimating π parallel results

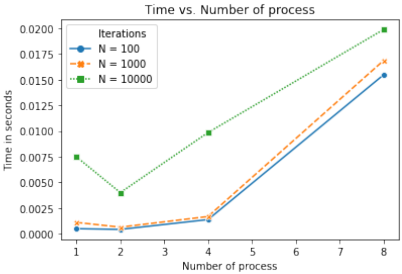

Figure: Time versus Number of processes

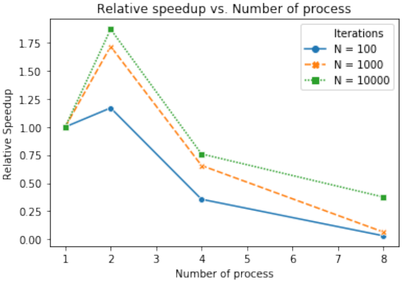

Figure: Relative speedup

The wall time is more for higher number of processes due to the system being less complex that parallelization is not worth it

Monte Carlo simulations for options pricing

Overview of Options

- Options are financial derivatives that give buyers the right, but not the obligation, to buy or sell an underlying asset at an agreed-upon price and date.
- Considering the basic options or vanilla options we have two categories - Call option and Put option
- A European (call/put) option is a contract that gives the holder the right, but not the obligation, to (buy/sell) an agreed quantity of a predetermined underlying asset, at a specific price and can only be executed at time of maturity
- An American (call/put) option is a contract that gives the holder the right, but not the obligation, to (buy/sell) an agreed quantity of a predetermined underlying asset, at a specific price and can be executed anytime before time of maturity

Payoff of options

- The specific price at which (buy/sell) happens is called the strike price or exercise price and the current value of underlying asset is stock price
- The price of the option is given by the payoff it can produce at the time of maturity
- For a simple call option and simple put option the payoff is given by the following expressions:

$$call = max(stock\ price\ -\ strike\ price, 0)$$
$$put = max(strike\ price\ -\ stock\ price, 0)$$

Monte Carlo Simulation for option pricing

- Given that the price of the option is driven by a random variable, it can be modelled by a Monte Carlo Simulation
- The need for Monte Carlo Simulation is due to the complexity of the closed form expressions when the option is not an european option
- For complex options such as path dependent options, there is no closed form expression available and hence we will have to simulate the complete scenario to obtain the price of the option
- By using Monte Carlo Simulation, the changes required in an option can be modified in the fundamental equations and the complete path can be simulated, thereby the price of the option can be computed

Framework for parallelization of Monte Carlo Simulations

Framework for computing the price

- For all the options irrespective of path dependent or independent, we will be simulating the trajectories of the underlying asset using the governing equations
- The number of paths generated is equal to the number of iterations for which the Monte Carlo simulation is carried out
- Each of the path generated is independent of the other paths, but the values in a path are dependent on each other

Framework for parallelization

- Given that each of the paths are independent of each other, we can parallelize this on a task based parallelization model
- Assuming the total number of iterations to be N and number of processes to be nProcs, we can divide the task to be performed to N/nProcs blocks for each of the process and they compute the same
- After the simulation in each of the process, the data is retrived in root and the simulation is consolidated for the required iterations

Implementation of Monte Carlo Simulation

Mathematical Modelling

Governing Equation

- The stock price of the underlying asset is modelled with the help of the the current stock price, time step considered, volatility of underlying stock, drift rate of stock and a random variable

$$dS = \mu S ds + \sigma S d$$

- Where S is the Stock price, μ is the risk free rate, σ is the volatility and d is the drift rate of stock
- The time discretized form of the wiener process called as ito's process is:

$$\Delta S = \mu S \Delta t + \sigma \epsilon S \Delta t$$

Mathematical Modelling

Governing Equation

- A more useful form of the previous expression for Monte Carlo Simulation is:

$$S + \Delta S = Se^{\left[\left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma\epsilon\sqrt{\Delta t}\right]}$$

- This expression can be rewritten in a time marching form as the following:

$$S_{t+1} = S_t e^{\left[\left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma\epsilon\sqrt{\Delta t}\right]}$$

- Where ϵ is the random variable from a normal distribution. The above expression will be modified and used for generating the paths in case of a Monte Carlo Simulation.

Path Independent Option Pricing

- For path independent options, the Δt can be directly set to the time to expiry as the price is path independent
- The expression narrows down to:

$$S_T = S_t e^{\left[\left(\mu - \frac{\sigma^2}{2}\right)(T-t) + \sigma \epsilon \sqrt{T-t}\right]}$$

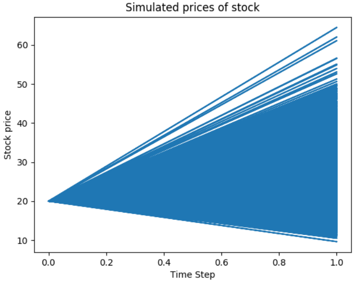

Figure: Simulation with N = 10000

Distribution of predicted stock prices

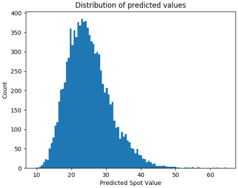

Figure: Simulation with N = 10000
Price = 5.8497±0.2233$

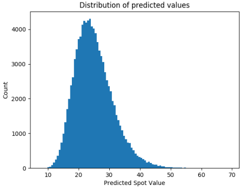

Figure: Simulation with N = 100000
Price = 5.9658±0.0775$

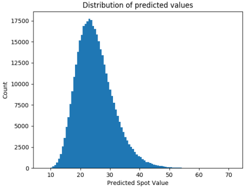

Figure: Simulation with N = 400000
Price = 5.978±0.0353$

Literature value of option price: 5.9842$

Path Independent Option Pricing Results

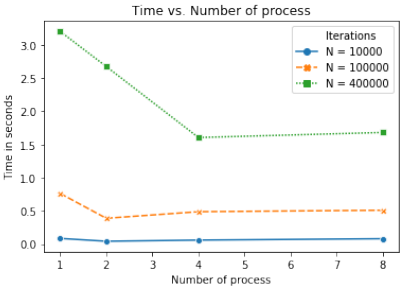

Figure: Time versus Number of processes

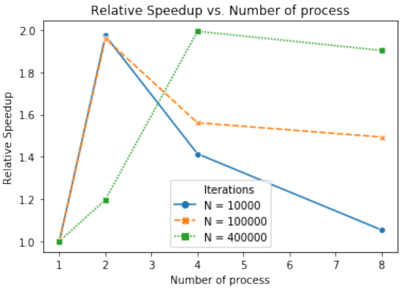

Figure: Relative speedup

Path Dependent Option Pricing

- For path dependent options, the entire path needs to calculated and hence the expression can't be reduced
- The expression for computation is:

$$S_{t+1} = S_t e^{\left[\left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma \epsilon \sqrt{\Delta t}\right]}$$

- The expression is evaluated in time steps of Δt from S_t to S_T

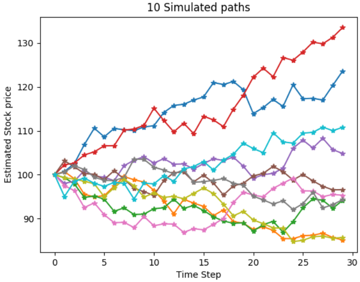

Figure: Sample of 10 Simulated paths

Distribution of predicted stock prices

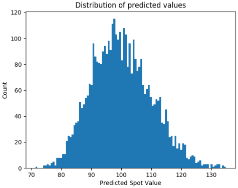

Figure: Simulation with N = 4000
Price = 5.7953±0.3062$

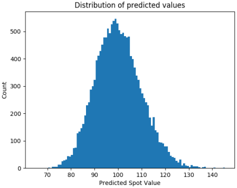

Figure: Simulation with N = 16000
Price = 5.7295±0.1509$

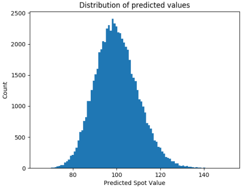

Figure: Simulation with N = 64000
Price = 5.7041±0.075$

Literature value of option price: 5.730$

Path Dependent Option Pricing Results

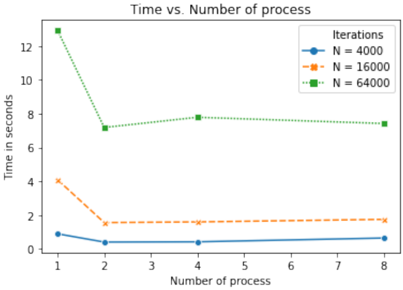

Figure: Time versus Number of processes

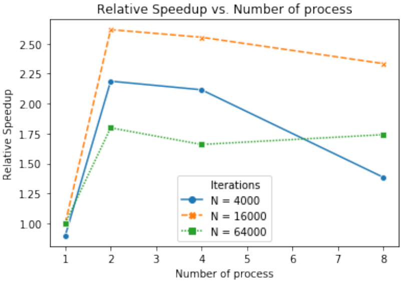

Figure: Relative speedup

Thank You!