MFE 405 Computational Methods in Finance Project 1

Vinamra Rai

April 21, 2024

1 Lecture 1

1.1 Problem 1

1.1.1 (a)

The probability of X being at least 35 is 0.021 from using the LGM random number generation. The exact probability from the closed-form solution comes out to be 0.02003. The minor difference in the values could be negated by using a higher number of random numbers for each of Bernoulli's distributions.

1.1.2 (b)

Probability of X $\geq$ 1: 0.5045
Probability of X $\geq$ 4: 0.0674
Mean of the distribution: 1.4957
Standard Deviation of the distribution: 1.5087

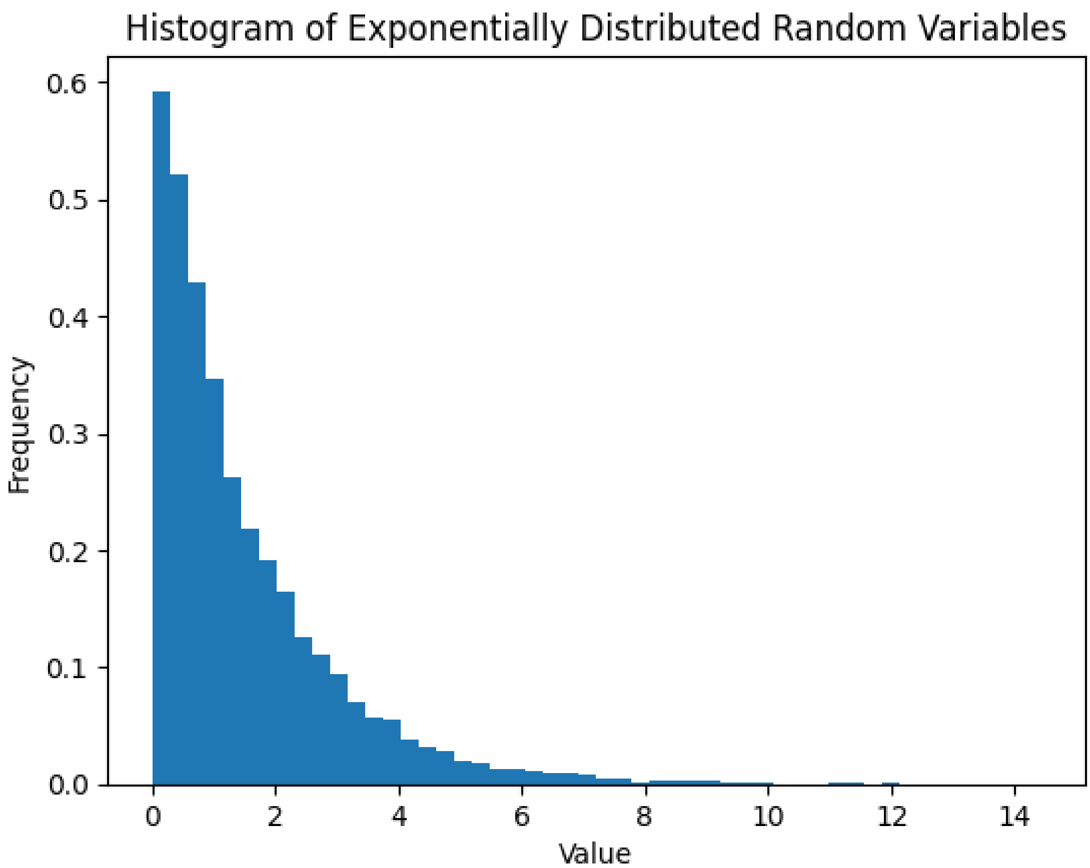

Figure 1: Plot from 1(b)

1.1.3 (e)

To identify a clear difference between the runtimes of the Box-Muller and Polar Marsaglia methods, 200,000 normally distributed random variables were generated. The runtime results are as follows: Even though theoretically, Polar Marsaglia is supposed to be more efficient since the trigonometric functions used in the Box Muller method are computationally expensive, here we see a different result possible for two reasons:

Table 1: Time taken to generate 2000 normally distributed random variables

Method	Time (seconds)
Box-Muller	0.556
Polar-Marsaglia	1.068

In the Polar Marsaglia method, we could generate a lot more variables from the LGM random number generator to satisfy the unit circle criterion. Moreover, in the code, I used loops for Polar Marsaglia, whereas for Box-Muller, all the values were generated in one go using the numpy arrays, which might have further contributed to the discrepancy.

2 Lecture 2

2.1 Problem 2

2.1.1 (b)

As time t increases, the overall expected value of $B(t)$ being closely associated with the behavior of the cosine function, expected to be $\exp(-t/2)$ seems to be oscillating around the value of 1.

2.1.2 (c)

In our efforts to compute the expected value $B(5)$, we implemented the control variates technique to enhance the efficiency of our estimations. Specifically, we utilized $\cos(W_t)$ as the control variate. This choice was strategic; since the expected value of $\cos(W_t)$ is zero, it doesn't introduce bias. Moreover, $\cos(W_t)$ inherently captures the non-linear nature of $B(5)$, which makes it a strong candidate for reducing variance.

The application of the control variates technique led to significant improvements. The original expected value of $B(5)$ remained unchanged; however, the variance experienced a dramatic decrease. This indicates an enhancement in the precision of our expected value computation for $B(5)$, as shown by the reduction of variance close to zero. The following table encapsulates the comparative results before and after the application of control variates:

Table 2: Effectiveness of Control Variates in Variance Reduction

Metric	Original	Updated
Expected Value	1.051633	1.051633
Variance	72.987757	7.298922e-09
Reduction in Variance (%)	N/A	100.0

The table reveals that the expected value post-adjustment remains the same, substantiating the non-biasing nature of the control variate. The variance, however, is virtually eliminated, indicating a variance reduction of 100%.

2.2 Problem 3

2.2.1 (c)

To estimate the price of the option as in part (a), we employed variance reduction techniques, specifically the use of antithetic variates, while maintaining the same number of simulations. This approach capitalizes on the negatively correlated paths to decrease the variance of our estimator.

When we examine the variance, the effectiveness of the variance reduction technique is apparent. The variance of the estimator using standard Monte Carlo was 1714.4094, which was substantially reduced to 410.8371 with the implementation of Antithetic Variates. This constitutes a variance reduction of approximately 76.04%.

2.3 Problem 4

2.3.1 (a)

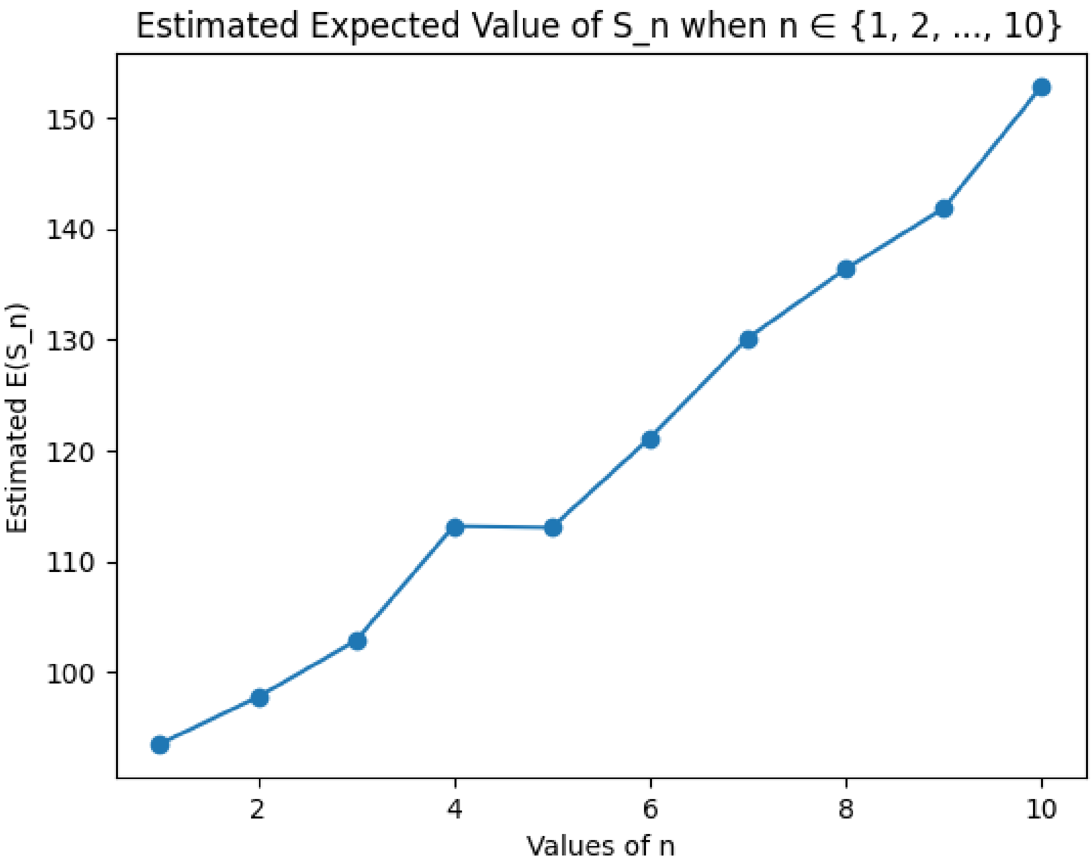

Figure 2: Plot from 4(a)

2.3.2 (b)

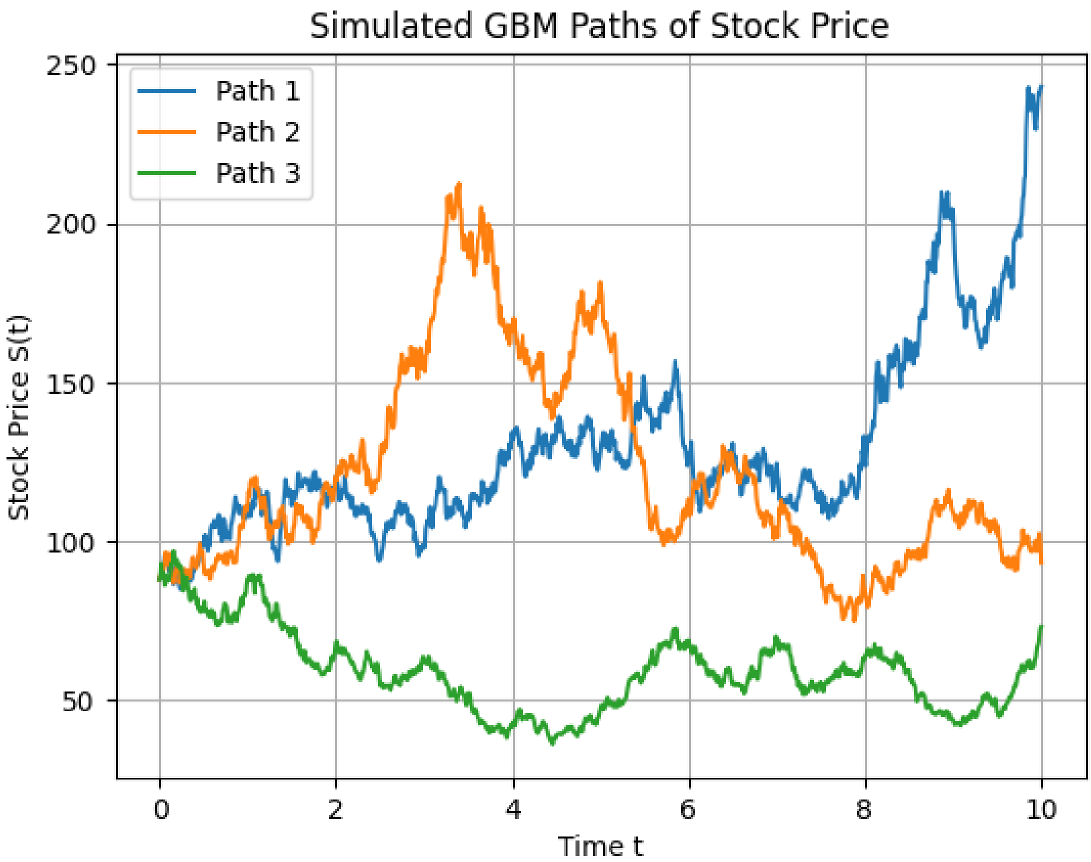

Figure 3: Plot from 4(b)

2.3.3 (c)

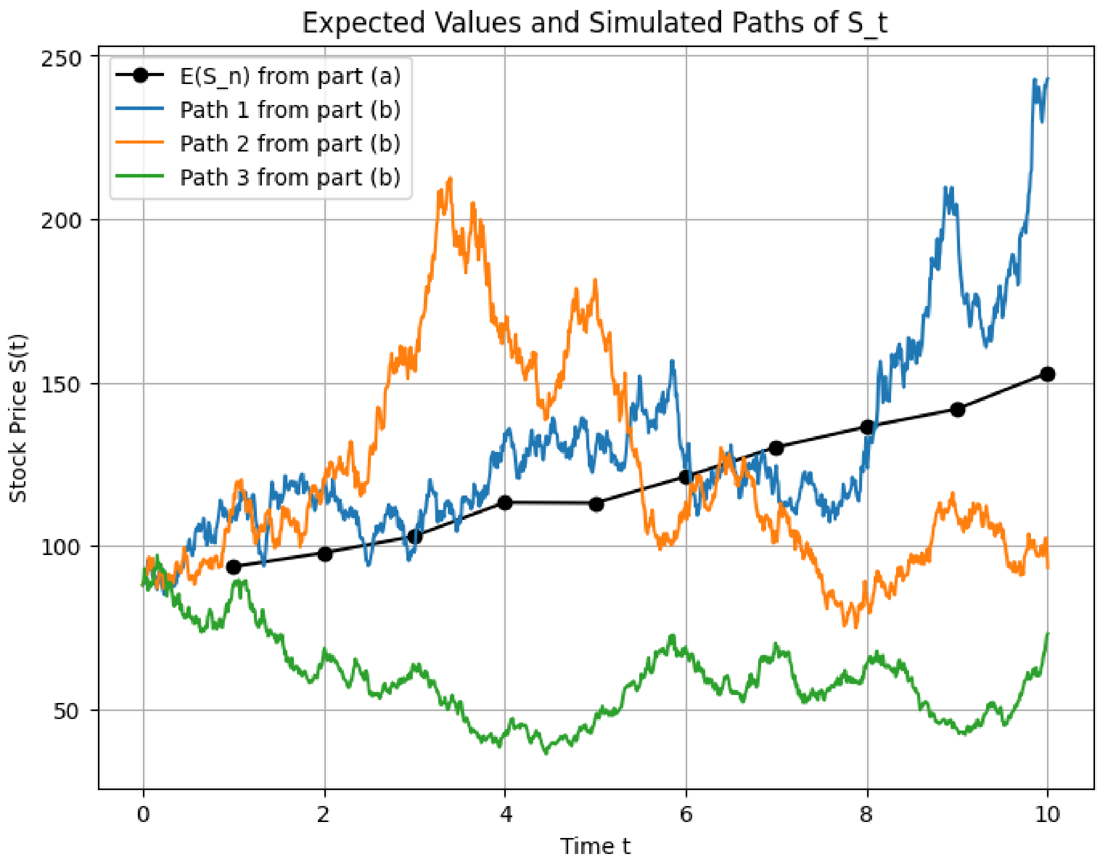

Figure 4: Plot from 4(c)

2.3.4 (d)

Increasing the volatility parameter σ in the Geometric Brownian Motion (GBM) model leads to higher uncertainty in stock price movement. The expected value $E(S_n) = S_0 e^{(r-\frac{\sigma^2}{2})t}$ remains unchanged, as it depends on the constant drift rate r. However, the variance of the stock price at future times increases with σ, resulting in wider confidence intervals around the expected value.

The effects on the plots from parts (a) and (b) when σ increases from 20% to 30% are:

- The graph of $E(S_n)$ should ideally remain the same, as the increase in σ does not directly affect the expected value but influences the dispersion around it. However, to achieve this effect through simulations, we would need to increase the number of simulations to a much higher number for better convergence.
- The individual paths of S_t show greater deviation from the initial stock price S_0 over time due to the increased magnitude of the diffusion term σW_t in the GBM formula, resulting in more pronounced upswings and downswings.

3 Lecture 3

3.1 Problem 5

3.1.1 (d)

Table 3: Comparison of Pricing Methods

S_0	Euler Price	Euler Std Error	Milshtein Price	Milshtein Std Error	BSM Price
95	6.093664	0.33916	5.2246	0.322383	5.688951
96	5.906561	0.311848	6.47439	0.337017	6.182333
97	6.289305	0.339605	6.459504	0.339642	6.699223
98	6.860547	0.362899	7.856926	0.37838	7.239329
99	7.681349	0.382469	7.862351	0.380283	7.802283
100	7.733764	0.384682	7.98144	0.390981	8.387628
101	9.571805	0.419875	8.970496	0.42704	8.994862
102	9.398707	0.425544	9.319901	0.418271	9.623445
103	9.996552	0.434561	10.221695	0.45066	10.272787
104	10.703675	0.442653	11.491078	0.49255	10.94226

Here, we observe that the pricing results obtained through Euler's and Milshtein's methods closely align with those from the $N(\cdot)$ approximation discussed in Part (c). Since the second derivative term associated with Milshtein's method is zero in this case, if we increase the number of simulations, the value generated will converge to the ones generated through Euler's process.

3.1.2 (e)

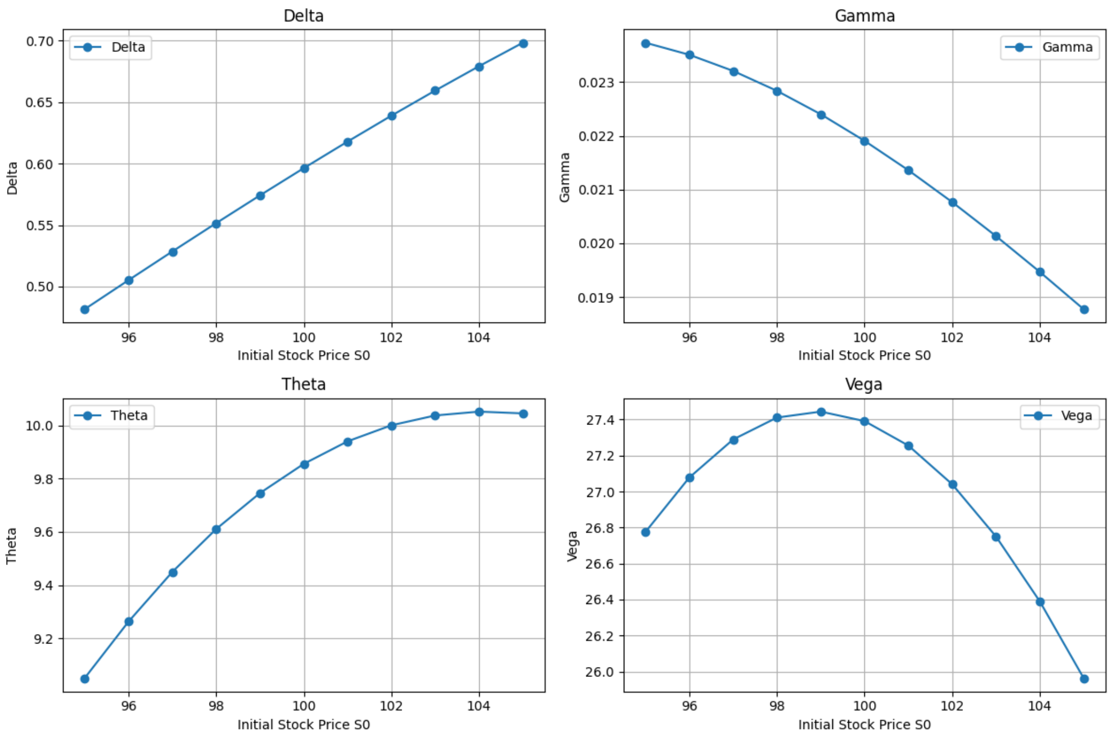

Figure 5: Plot from 5(e)