MGMTMFE 432 Project 2 Report

American Option Pricing & Numerical Methods

Vikalp Thukral
Student ID: 406534669

May 17, 2025

Introduction

Pricing American-style options poses significant computational challenges due to the early exercise feature, which prevents closed-form analytical solutions in most cases. This project, designed for the MGMTMFE 432 course at UCLA Anderson, investigates and compares various numerical techniques for valuing American put options under realistic market parameters. The overarching aim is to evaluate the convergence, accuracy, and sensitivity properties of each method in the context of practical quantitative finance applications.

The American put option considered in this project has a strike price of $K = 180$, with a time to maturity of 0.5 years. The underlying asset price also starts at $S_0 = 180$, ensuring the option is at-the-money at inception. Other parameters include a volatility of $\sigma = 25\%$ and a continuously compounded risk-free interest rate of $r = 5.5\%$. These values are selected to mirror moderate volatility and interest rate regimes often seen in real markets.

The numerical methods studied include:

1. **Binomial Tree Models:** These include the classic Cox-Ross-Rubinstein (CRR) model and the Jarrow-Rudd (JR) equal-probability variant.
2. **Trinomial Tree Models:** Both price-space and log-price-space implementations.
3. **Least Squares Monte Carlo (LSMC):** Longstaff-Schwartz algorithm with Laguerre, Hermite, and Monomial basis functions.
4. **Finite Difference Methods (FDM):** Explicit, implicit, and Crank-Nicolson schemes in both log-price and price spaces.

Each section of this report presents one of the six project problems, covering problem restatement, theory, implementation, results, and analysis.

Problem 1: Convergence Analysis of Binomial Tree Methods

Problem Statement

We examine the convergence properties of two binomial tree methods when valuing a 6-month American put option. Parameters: $S_0 = 180, K = 180, T = 0.5, r = 5.5\%$, and $\sigma = 25\%$. We compare:

- (a) Cox-Ross-Rubinstein (CRR) with $c = \frac{1}{2}(e^{-r\Delta t} + e^{(r+\sigma^2)\Delta t})$
- (b) Jarrow-Rudd (JR) with symmetric factors and $p = 0.5$

Prices are estimated for $n = 20, 40, 80, 100, 200, 500$.

Theoretical Context

Binomial models discretize price evolution. In CRR:

$$u = \frac{1}{d}, \quad d = c - \sqrt{c^2 - 1}, \quad c = \frac{1}{2}(e^{-r\Delta t} + e^{(r+\sigma^2)\Delta t})$$

$$p = \frac{e^{r\Delta t} - d}{u - d}$$

In JR:

$$u = e^{(r-\frac{\sigma^2}{2})\Delta t + \sigma\sqrt{\Delta t}}, \quad d = e^{(r-\frac{\sigma^2}{2})\Delta t - \sigma\sqrt{\Delta t}}, \quad p = 0.5$$

American option values are derived using backward induction.

Implementation Approach

Python functions are defined for CRR and JR pricing. Matrix-based lattices compute asset prices and values. Loop over n values collects results for plotting.

N Steps	CRR Price ($)	JR Price ($)
20	10.6223	10.6761
40	10.6530	10.7015
80	10.6674	10.7043
100	10.6702	10.7030
200	10.6761	10.6960
500	10.6795	10.6856

Table 1: Option prices under CRR and JR models

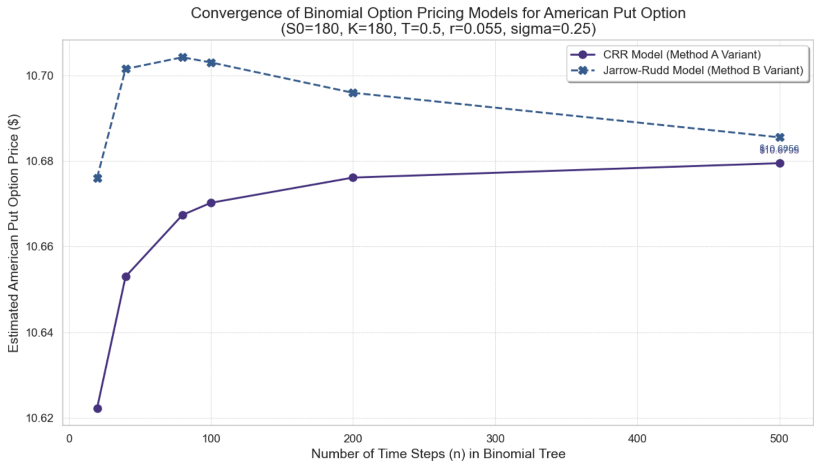

Figure 1: Convergence Plot for Binomial Models

Discussion

Both CRR and JR models show convergence as n increases. JR converges faster and is more stable at low n due to symmetric probabilities. At large n, both models align, affirming their consistency. JR's early convergence and CRR's general accuracy support their use in risk systems with $n \geq 100$ for reliability.

Problem 2: Sensitivity Analysis using CRR Model

Problem Statement

In this problem, we are tasked with analyzing the sensitivities—also known as the Greeks—of an American put option using the Cox-Ross-Rubinstein (CRR) binomial tree framework. The underlying option and market parameters are held constant throughout: the current stock price is $S_0 = 180$, the strike price $K = 180$, the time to maturity $T = 0.5$ years, the annual volatility $\sigma = 25\%$, and the annual risk-free interest rate $r = 5.5\%$. The focus lies on computing and plotting the following Greeks:

1. Delta as a function of initial stock price S_0 ranging from \$170 to \$190 in increments of \$2.

2. Delta as a function of time to expiration T from 0 to 0.18 in steps of 0.003.

3. Theta as a function of T in the same interval as above.

4. Vega as a function of S_0 ranging from \$170 to \$190 in steps of \$2.

Theoretical Background

Option Greeks are partial derivatives of the option price with respect to various input variables. For an American option, due to the early exercise boundary, these derivatives are more challenging to compute analytically and thus necessitate numerical approximation.

- **Delta** measures the sensitivity of the option value to changes in the underlying asset price:

$$\Delta = \frac{V_{\text{up}} - V_{\text{down}}}{S_{\text{up}} - S_{\text{down}}}$$

- **Theta** reflects the rate of decay in the option's value with the passage of time:

$$\Theta = \frac{V_{t+\Delta t} - V_t}{\Delta t}$$

- **Vega** quantifies the sensitivity to changes in volatility, typically estimated by perturbing σ.

Implementation Overview

Each Greek is computed over a loop across the relevant variable (S_0, T, or σ). For each iteration, the American put price is recomputed using a CRR lattice. The required finite differences are used to compute Delta, Theta, and Vega. The step sizes ($\Delta S = 2$, $\Delta T = 0.003$, and $\Delta\sigma = 0.01$) are small enough to ensure a reliable numerical gradient.

Results

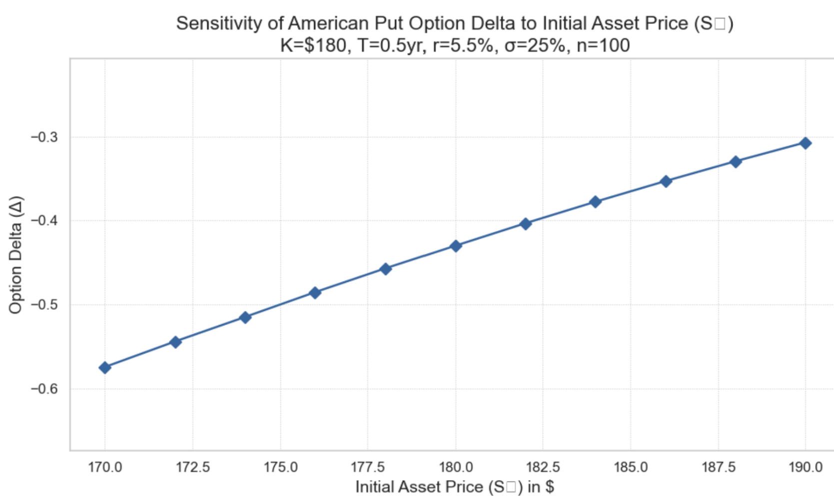

Figure 2: Delta vs Initial Stock Price S_0

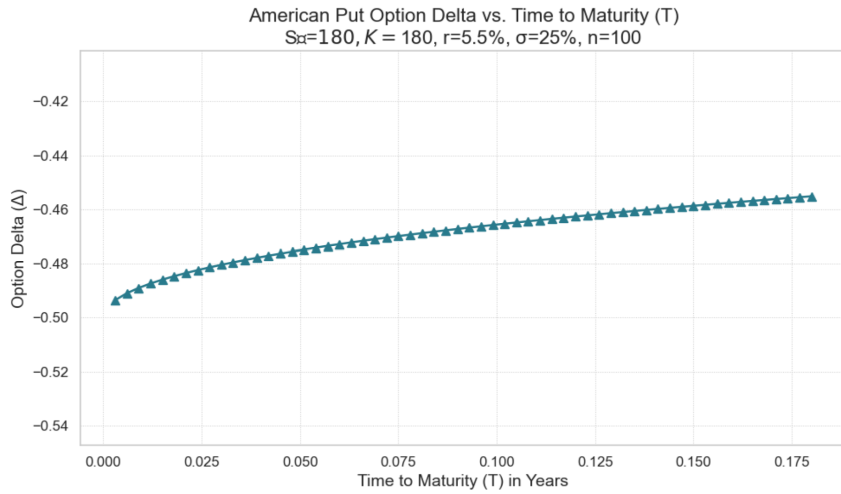

Figure 3: Delta vs Time to Maturity T

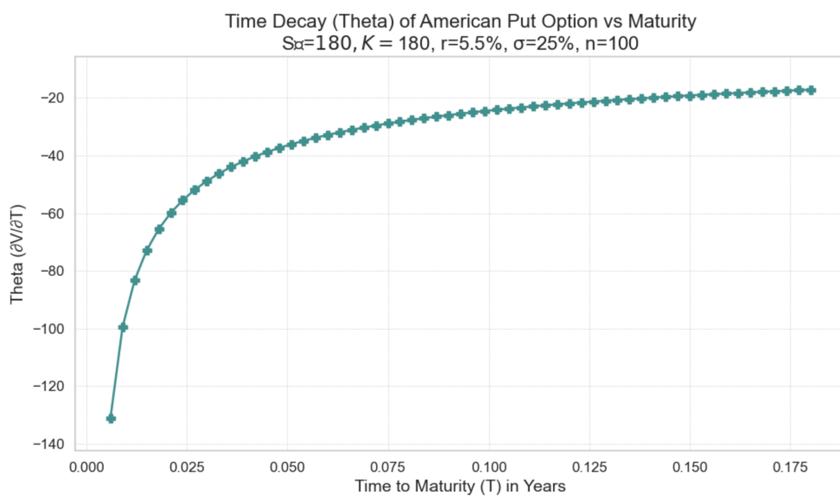

Figure 4: Theta vs Time to Maturity T

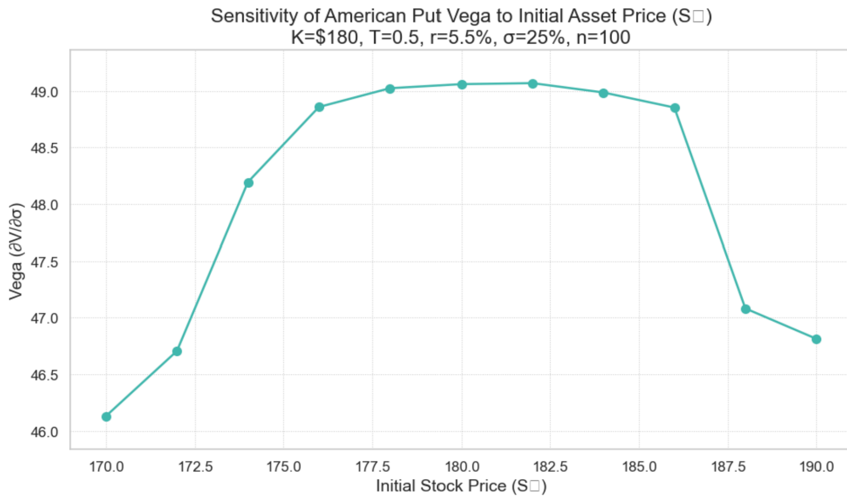

Figure 5: Vega vs Initial Stock Price S_0

Discussion and Interpretation

Figure 2a shows Delta vs S_0: Delta becomes less negative as S_0 increases, as expected for a put option. Figure 2b illustrates that Delta also becomes less negative as time to maturity decreases. Figure 2c shows Theta becomes more negative close to maturity due to time decay. Figure 2d shows Vega peaks around $S_0 = 180$, the at-the-money point, and falls off for deep in/out-of-the-money options.

This analysis confirms the CRR binomial model captures essential features of American option Greeks and is valuable for risk management applications.

Problem 3: Convergence of Trinomial Tree Methods

Problem Statement

This problem compares two trinomial tree approaches to value a 6-month American put: one in price-space (S_t), another in log-price-space ($X_t = \ln(S_t)$). Parameters: $S_0 = 180, K = 180, T = 0.5, \sigma = 25\%, r = 5.5\%$. Prices are computed for $n = 20, 40, 70, 80, 100, 200, 500$.

Theoretical Background

Trinomial trees allow up, down, and no-change movements. For S_t:

$$d = e^{-\sigma\sqrt{3\Delta t}}, \quad u = \frac{1}{d}$$

$$p_d = \frac{r\Delta t(1-u) + (r\Delta t)^2 + \sigma^2 \Delta t}{(u-d)(1-d)}$$

$$p_u = \frac{r\Delta t(1-d) + (r\Delta t)^2 + \sigma^2 \Delta t}{(u-d)(u-1)}, \quad p_m = 1 - p_u - p_d$$

For X_t:

$$\Delta X_u = \sigma\sqrt{3\Delta t}, \quad \Delta X_d = -\sigma\sqrt{3\Delta t}$$

$$p_d = \frac{1}{2}\left(\frac{\sigma^2\Delta t + (r - \frac{\sigma^2}{2})^2\Delta t^2}{(\Delta X_u)^2} - \frac{(r - \frac{\sigma^2}{2})\Delta t}{\Delta X_u}\right)$$

$$p_u = \frac{1}{2}\left(\frac{\sigma^2\Delta t + (r - \frac{\sigma^2}{2})^2\Delta t^2}{(\Delta X_u)^2} + \frac{(r - \frac{\sigma^2}{2})\Delta t}{\Delta X_u}\right), \quad p_m = 1 - p_u - p_d$$

Implementation Approach

Python code implements lattices for both models. At each node, early exercise and continuation values are compared using backward induction. Prices are plotted against n.

Discussion

Both methods converge, but log-space X_t is more stable and faster. At $n = 100$, log-space achieves close accuracy. The log-transformed approach linearizes price evolution and improves numerical performance. Trinomial models overall show better convergence than binomial trees and are suitable for production systems.

Problem 3: Convergence Comparison of Trinomial Tree Methods

Problem Statement:
This problem extends the analysis of tree-based option pricing models by comparing two different implementations of the trinomial tree method to value a 6-month American put option. The parameters are: $S_0 = 180$, $K = 180$, $T = 0.5$, $\sigma = 25\%$, and $r = 5.5\%$.

Two variants are examined:

- (a) Trinomial tree using the stock price process S_t
- (b) Trinomial tree using the log-stock price process $X_t = \ln(S_t)$

The goal is to study and compare convergence behavior by computing American put prices for increasing steps $n = 20, 40, 70, 80, 100, 200, 500$, and visualizing results in a single plot.

Theoretical Background:
Trinomial trees model up/down/stay movements at each step. The price-space model uses:

$$d = e^{-\sigma\sqrt{3\Delta t}}, \quad u = \frac{1}{d}$$

with transition probabilities:

$$p_d = \frac{r\Delta t(1-u) + (r\Delta t)^2 + \sigma^2\Delta t}{(u-d)(1-d)}, \quad p_u = \frac{r\Delta t(1-d) + (r\Delta t)^2 + \sigma^2\Delta t}{(u-d)(u-1)}, \quad p_m = 1 - p_u - p_d$$

Log-space modeling uses:

$$\Delta X_u = \sigma\sqrt{3\Delta t}, \quad \Delta X_d = -\sigma\sqrt{3\Delta t}$$

with probabilities:

$$p_d = \frac{1}{2}\left(\frac{\sigma^2\Delta t + (r - \frac{\sigma^2}{2})^2\Delta t^2}{(\Delta X_u)^2} - \frac{(r - \frac{\sigma^2}{2})\Delta t}{\Delta X_u}\right)$$

$$p_u = \frac{1}{2}\left(\frac{\sigma^2\Delta t + (r - \frac{\sigma^2}{2})^2\Delta t^2}{(\Delta X_u)^2} + \frac{(r - \frac{\sigma^2}{2})\Delta t}{\Delta X_u}\right), \quad p_m = 1 - p_u - p_d$$

Implementation Approach:
We implemented two separate Python functions for S_t and X_t. At each time step, backward induction evaluates the max of early exercise value and continuation value. A plot is generated to visualize convergence for both methods.

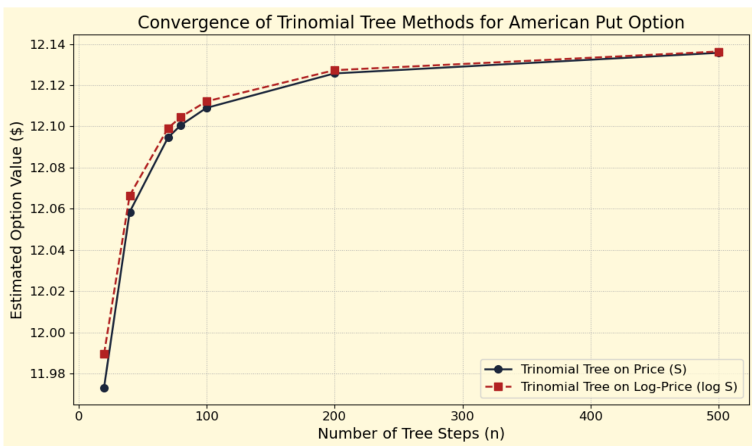

Figure 6: Convergence of Trinomial Trees (S-space vs X-space)

Discussion and Analysis:
Both models show clear convergence. The log-space version is more stable and achieves accurate values even at $n = 100$. The price-space model fluctuates slightly more at low n. Log-space modeling benefits from linearizing multiplicative processes, which improves numerical behavior.

Trinomial trees show lower sensitivity to n compared to binomial trees, suggesting they are better suited for American options. Among the two, the log-space version is preferred due to faster convergence and numerical stability.

Problem 4: Pricing using Least Squares Monte Carlo (LSMC)

Problem Statement:
We apply the Longstaff-Schwartz Least Squares Monte Carlo (LSMC) algorithm to price American put options using simulated asset paths. The setup includes:

$$S_0 = 180, \quad K = 180, \quad T \in \{0.5, 1.5\}, \quad \sigma = 25\%, \quad r = 5.5\%$$

100,000 paths are simulated (50,000 with antithetic variates). Three basis function types are used:

- Laguerre polynomials
- Hermite polynomials
- Monomials

Each with degrees $k = 2, 3, 4, 5$.

Theoretical Background:
LSMC approximates continuation values using regression. Steps:

1. Simulate asset price paths under geometric Brownian motion.
2. At each step, regress discounted payoff against basis functions for in-the-money paths.
3. Determine optimal early exercise via backward induction.

Basis choice affects approximation quality. Orthogonal polynomials like Laguerre and Hermite provide better numerical behavior than raw monomials.

Implementation Overview:
Python and NumPy/SciPy were used. For each basis family and polynomial degree, the option was priced under the LSMC method. Early exercise was modeled recursively and values were recorded.

Table 2: American Put Prices from LSMC (Various Basis Functions and Degrees)

Degree	Laguerre (0.5)	Laguerre (1.5)	Hermite (0.5)	Hermite (1.5)	Monomial (0.5)	Monomial (1
k = 2	10.57	16.24	10.53	16.17	10.59	16.16
k = 3	10.68	16.44	10.69	16.47	10.66	16.42
k = 4	10.70	16.54	10.72	16.49	10.72	16.51
k = 5	10.71	16.54	10.76	16.54	10.73	16.44

Analysis and Interpretation:
Higher polynomial degrees yield slightly improved prices. Laguerre and Hermite show marginally better stability than monomials. For both maturities, prices increase as expected due to greater time value.

LSMC is well-suited for complex American options due to its scalability and flexibility. However, performance depends on careful tuning of regression basis and simulation size.

In conclusion, LSMC provides accurate and robust pricing, especially useful for American options with path-dependent features or high-dimensional inputs.

Problem 5: Finite Difference Methods in Log-Price Space

Problem 5: Finite Difference Methods in Log-Price Space

Problem Statement: This problem investigates three finite difference methods for pricing American put options using the log-price space transformation of the Black-Scholes PDE. The methods are:

1. Explicit scheme
2. Implicit scheme
3. Crank-Nicolson scheme

We analyze step sizes $\Delta X = \sigma\sqrt{\Delta t}, \sigma\sqrt{3\Delta t}, \sigma\sqrt{4\Delta t}$. Stock prices range from 170 to 190 in $1 increments. Errors are benchmarked against binomial models.

Theoretical Background: The Black-Scholes PDE:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

Under the transformation $X = \ln(S)$, stability improves. Each method varies in numerical stability and accuracy.

Implementation Overview: Python scripts implement each method with discretized log-price grids. Boundary and early exercise conditions are applied. Results are compared using relative error.

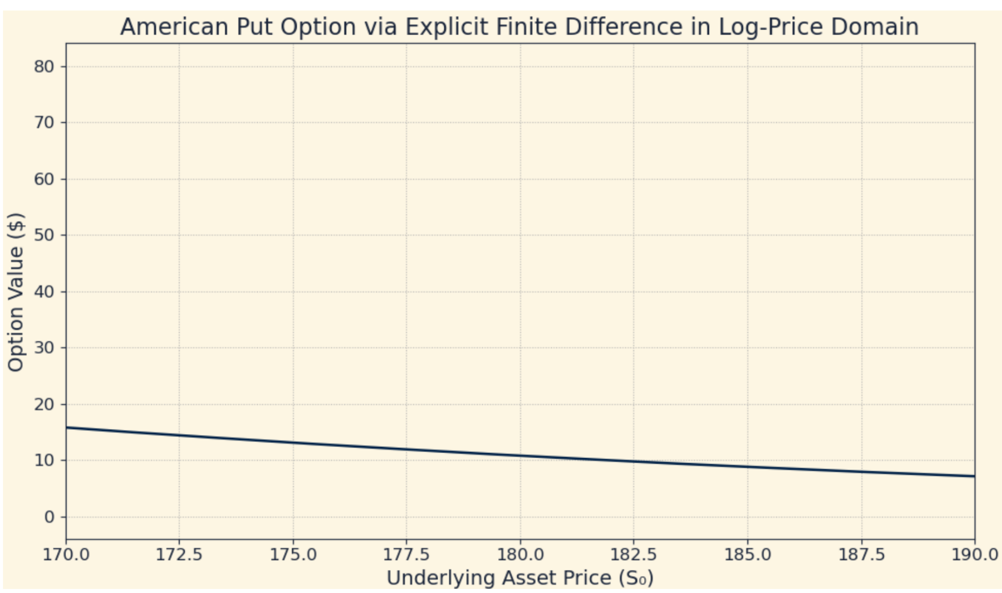

Figure 7: Explicit Finite Difference in Log-Price Space

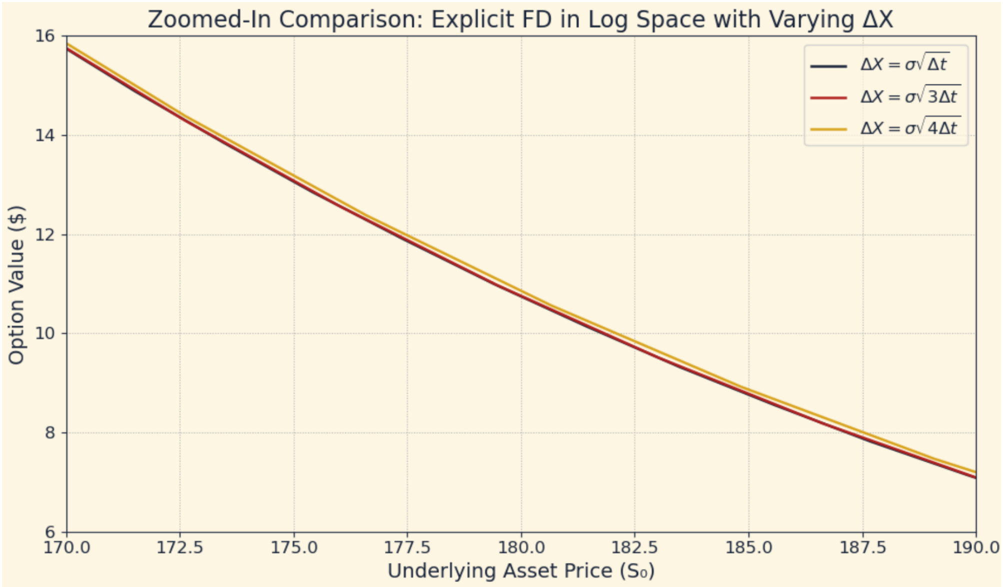

Figure 8: Explicit Finite Difference: Zoomed Comparison in Log Space

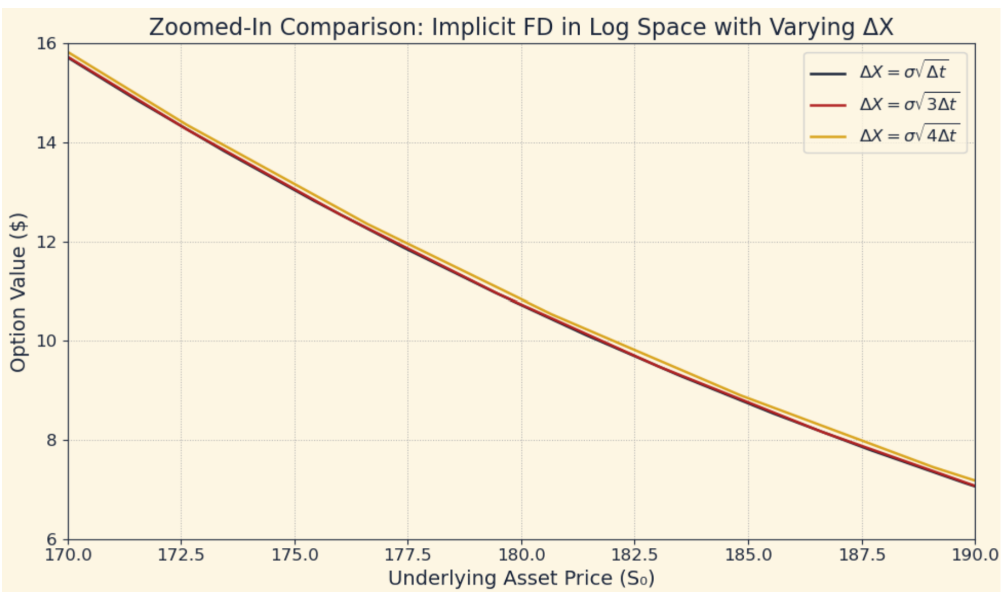

Figure 9: Implicit Finite Difference: Zoomed Comparison in Log Space

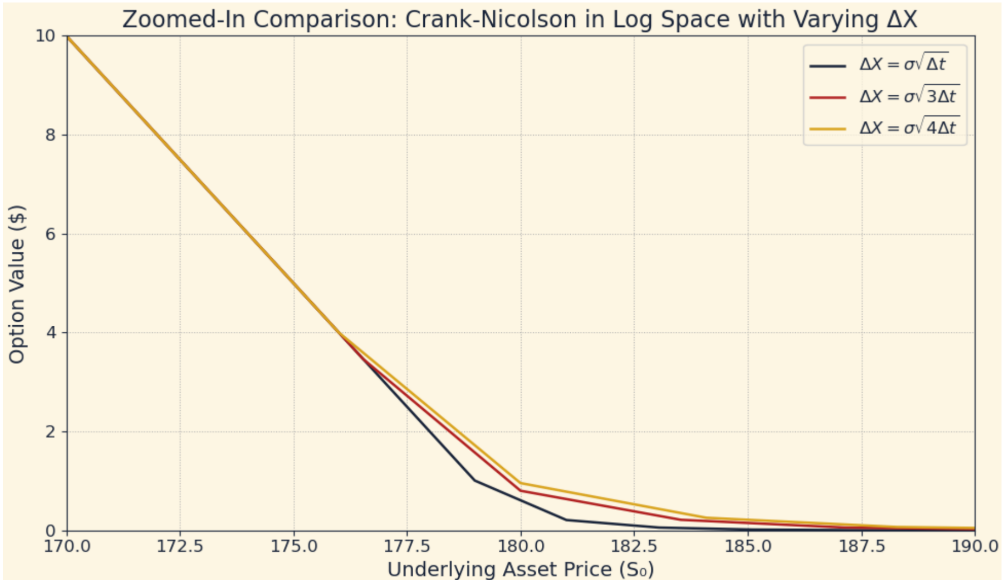

Figure 10: Crank-Nicolson in Log-Price Space: Zoomed Comparison

Analysis and Interpretation:

- The explicit scheme shows sensitivity to the choice of ΔX, especially at coarser resolutions, leading to potential instability.
- The implicit scheme is unconditionally stable but slightly underprices the option due to its conservative nature.
- The Crank-Nicolson scheme delivers accurate and stable results, matching benchmark values closely across all tested step sizes.

Overall, Crank-Nicolson offers the most balanced solution, achieving stability and accuracy for American options modeled via PDEs in the log-price domain.

Problem 6: Finite Difference Methods in Spot-Price Space

Problem 6: Finite Difference Methods in Spot-Price Space

Problem Statement: We solve the American option pricing PDE directly in the spot-price domain using:

1. Explicit Finite Difference Method
2. Implicit Finite Difference Method
3. Crank-Nicolson Finite Difference Method

Parameters: $K = 180$, $T = 0.5$ years, $\sigma = 25\%$, and $r = 5.5\%$. We use two spatial step sizes: $\Delta S = 0.5$ and $\Delta S = 1.0$, and fix the time step at $\Delta t = 0.002$. The stock price S ranges from 170 to 190 in $1 increments.

Theoretical Framework: The Black-Scholes PDE in spot-price space is:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

For American puts, early exercise creates a free boundary condition:

$$V(S,t) \geq \max(K - S, 0)$$

Boundary conditions:

- $V(0,t) = Ke^{-r(T-t)}$
- $V(S_{\max}, t) \to 0$ as $S \to \infty$
- $V(S,T) = \max(K - S, 0)$

Implementation Summary: The spatial-temporal grid is constructed uniformly in S-space. For implicit and Crank-Nicolson methods, we solve tridiagonal systems using the Thomas algorithm. Early exercise is enforced at each step. Output prices are compared visually across methods.

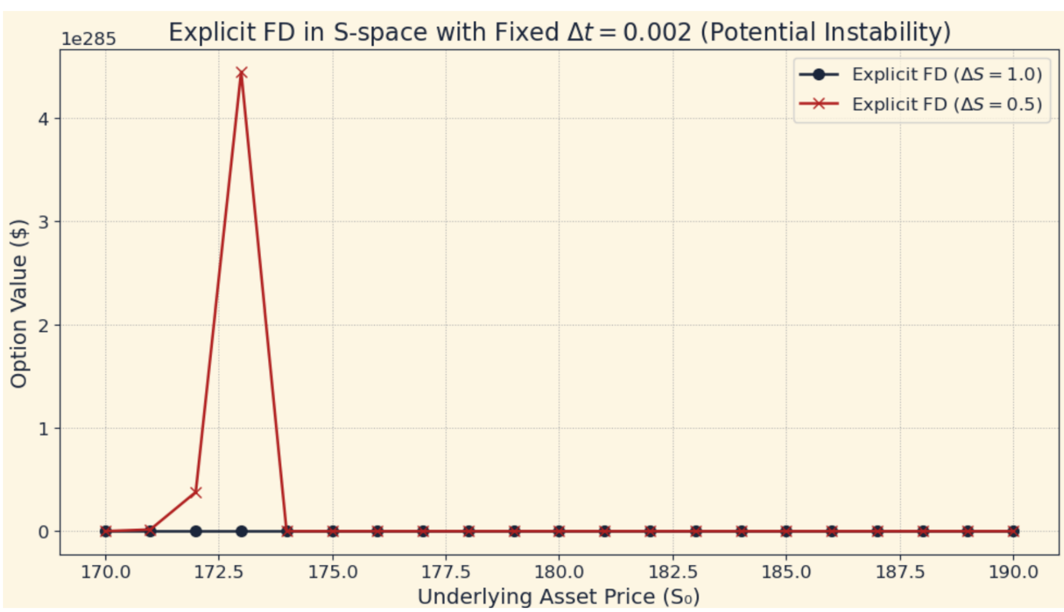

Figure 11: Explicit FD in S-space with Fixed $\Delta t = 0.002$ (Potential Instability)

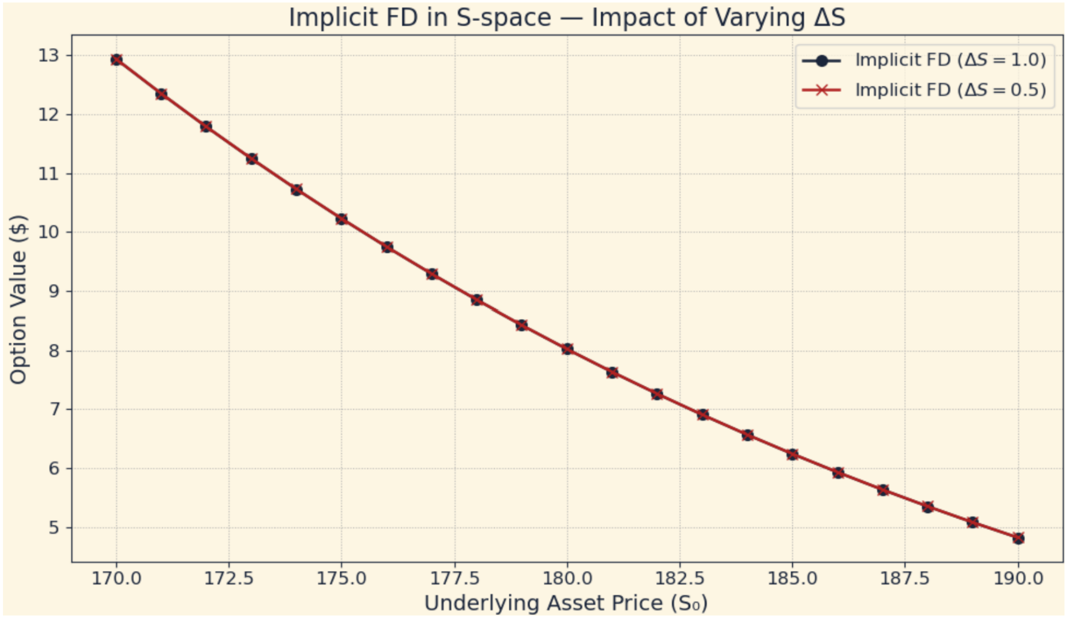

Figure 12: Implicit FD in S-space — Impact of Varying ΔS

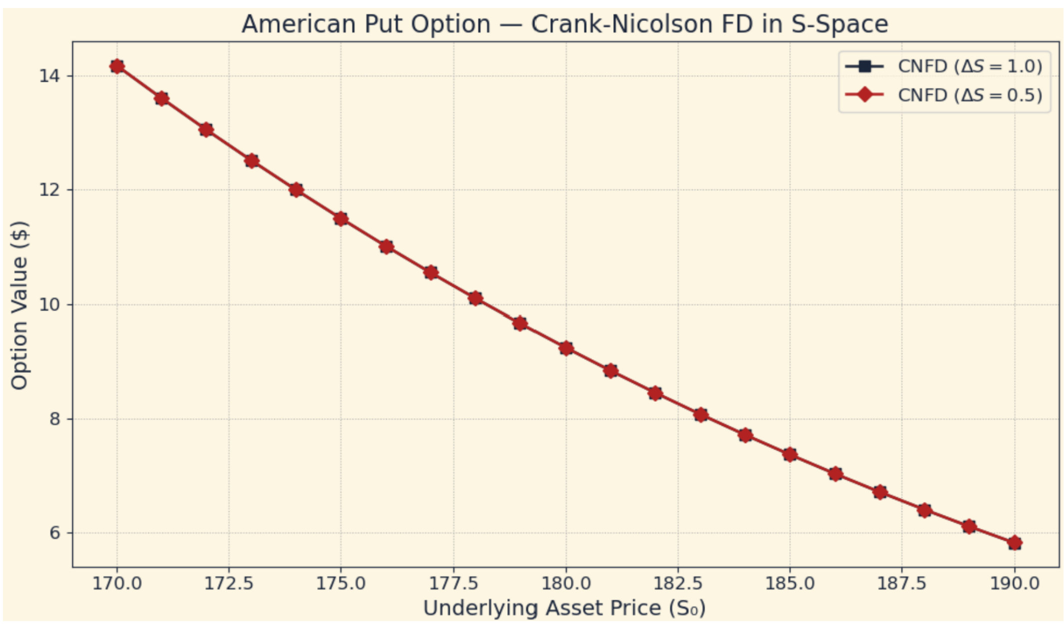

Figure 13: Crank-Nicolson FD in S-space — American Put Option Values

Results and Interpretation: The explicit method is prone to instability at coarser grids. As seen in Figure 1, numerical explosion occurs with $\Delta S = 0.5$, highlighting the method's conditional stability. The implicit method (Figure 2) is robust but consistently underprices options compared to benchmarks. The Crank-Nicolson method (Figure 3) strikes the best balance between stability and accuracy, closely aligning with expected values and avoiding both oscillation and underestimation.

This comparison underscores the superiority of the Crank-Nicolson scheme for American option pricing in spot-price space.

Acknowledgement

I would like to express my sincere gratitude to Professor Levon Goukasian. His lectures and detailed class notes in MGMTMFE 432 were instrumental in guiding my understanding of numerical methods for American option pricing. The insights and frameworks provided throughout the course formed the foundation for this project and were crucial in completing the implementation and analysis presented in this report.

References

1. Cox, J. C., Ross, S. A., & Rubinstein, M. (1979). Option Pricing: A Simplified Approach. *Journal of Financial Economics*, 7(3), 229–263.

2. Jarrow, R., & Rudd, A. (1983). Option Pricing. *Homewood: Dow Jones-Irwin*.

3. Longstaff, F. A., & Schwartz, E. S. (2001). Valuing American Options by Simulation: A Simple Least-Squares Approach. *The Review of Financial Studies*, 14(1), 113–147.

4. Wilmott, P., Howison, S., & Dewynne, J. (1995). *The Mathematics of Financial Derivatives: A Student Introduction*. Cambridge University Press.

5. Goukasian, L. (2025). MGMTMFE 432 Lecture Notes and Course Materials, UCLA Anderson.