

A Guided Tour of Chapter 7: Derivatives Pricing and Hedging

Ashwin Rao

ICME, Stanford University

February 15, 2021

Brief Overview of Derivatives

- Term *Derivative* comes from the word *Derived*
- Financial product whose structure (and hence, *value*) is derived from the *performance* of an *underlying* entity
- Technically a legal contract between buyer and seller that is either:
 - *Lock-type*: Entitles buyer to future contingent-cashflow (*payoff*)
 - *Option-type*: Buyer has future *choices*, leading to contingent-cashflow
- Some common derivatives:
 - Forward - Contract to deliver/receive asset on future date for fixed cash

$$\text{Forward Payoff: } f(X_t) = X_t - K$$

- European Option - *Right* to buy/sell on future data at agreed price

$$\text{Call and Put Option Payoff: } \max(X_t - K, 0) \text{ and } \max(K - X_t, 0)$$

- American Option - Can exercise option on *any day* before expiration
- Why do we need derivatives?
 - To protect against adverse market movements (*risk-management*)
 - To express a market view *cheaply* (leveraged trade)

Derivatives Pricing and Hedging problems as MDPs

- *Pricing*: Determination of fair value of an asset or derivative
- *Hedging*: Protect against market movements with “opposite” trades
- *Replication*: Clone payoff of a derivative with trades in other assets
- We consider two applications of Stochastic Control here:
 - Optimal Exercise of American Options in an idealized setting
 - Optimal Hedging of Derivatives Portfolio in a real-world setting
- Both problems enable us to price the respective derivatives
- Expressing these problems as MDP Control brings ADP/RL into play
- Optimal Exercise of American Options is Optimal Stopping problem
- So we start by learning about Stopping Time and Optimal Stopping

Stopping Time

- Stopping time τ is a “random time” (random variable) interpreted as time at which a given stochastic process exhibits certain behavior
- Stopping time often defined by a “stopping policy” to decide whether to continue/stop a process based on present position and past events
- Random variable τ such that $Pr[\tau \leq t]$ is in σ -algebra \mathcal{F}_t , for all t
- Deciding whether $\tau \leq t$ only depends on information up to time t
- Hitting time of a Borel set A for a process X_t is the first time X_t takes a value within the set A
- Hitting time is an example of stopping time. Formally,

$$T_{X,A} = \min\{t \in \mathbb{R} | X_t \in A\}$$

eg: Hitting time of a process to exceed a certain fixed level

Optimal Stopping Problem

- Optimal Stopping problem for Stochastic Process X_t :

$$W(x) = \max_{\tau} \mathbb{E}[H(X_{\tau}) | X_0 = x]$$

where τ is a set of stopping times of X_t , $W(\cdot)$ is called the Value function, and H is the Reward function.

- Note that sometimes we can have several stopping times that maximize $\mathbb{E}[H(X_{\tau})]$ and we say that the optimal stopping time is the smallest stopping time achieving the maximum value.
- Example of Optimal Stopping: Optimal Exercise of American Options
 - X_t is risk-neutral process for underlying security's price
 - x is underlying security's current price
 - τ is set of exercise times corresponding to various stopping policies
 - $W(\cdot)$ is American option price as function of underlying's current price
 - $H(\cdot)$ is the option payoff function, adjusted for time-discounting

Optimal Stopping Problems as Markov Decision Processes

- We formulate Stopping Time problems as Markov Decision Processes
- *State* is X_t
- *Action* is Boolean: Stop or Continue
- *Reward* always 0, except upon Stopping (when it is $= H(X_\tau)$)
- *State*-transitions governed by the Stochastic Process X_t
- For discrete time steps, the Bellman Optimality Equation is:

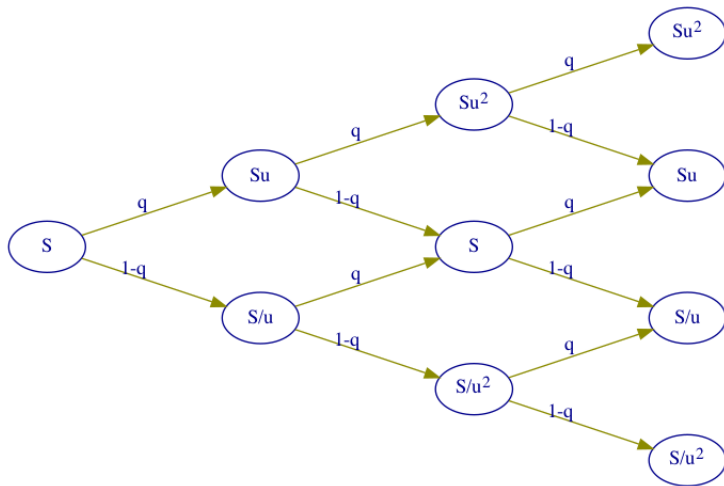
$$V^*(X_t) = \max(H(X_t), \mathbb{E}[V^*(X_{t+1})|X_t])$$

- For finite number of time steps, we can do a simple backward induction algorithm from final time step back to time step 0

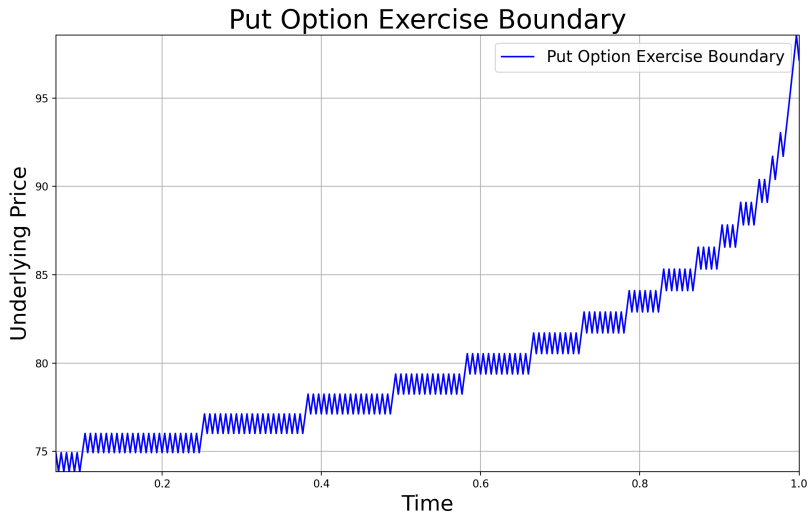
Mainstream approaches to American Option Pricing

- American Option Pricing is Optimal Stopping, and hence an MDP
- So can be tackled with Dynamic Programming or RL algorithms
- But let us first review the mainstream approaches
- For some American options, just price the European, eg: vanilla call
- When payoff is not path-dependent and state dimension is not large, we can do backward induction on a binomial/trinomial tree/grid
- Otherwise, the standard approach is [Longstaff-Schwartz algorithm](#)
- Longstaff-Schwartz algorithm combines 3 ideas:
 - Valuation based on Monte-Carlo simulation
 - Function approximation of continuation value for in-the-money states
 - Backward-recursive determination of early exercise states
- RL is an attractive alternative to Longstaff-Schwartz algorithm

Binomial Tree for Backward Induction



Optimal Exercise Boundary of American Put Option



Classical Pricing and Hedging of Derivatives

- Classical Pricing/Hedging Theory is based on a few core concepts:
 - **Arbitrage-Free Market** - where you cannot make money from nothing
 - **Replication** - when the payoff of a *Derivative* can be constructed by assembling (and rebalancing) a portfolio of the underlying securities
 - **Complete Market** - where payoffs of all derivatives can be replicated
 - **Risk-Neutral Measure** - Altered probability measure for movements of underlying securities for mathematical convenience in pricing
- Assumptions of arbitrage-free and completeness lead to (dynamic, exact, unique) replication of derivatives with the underlying securities
- Assumptions of frictionless trading provide these idealistic conditions
- Frictionless := continuous trading, any volume, no transaction costs
- Replication strategy gives us the pricing and hedging solutions
- This is the foundation of the famous Black-Scholes formulas
- However, the real-world has many frictions \Rightarrow *Incomplete Market*
- ... where derivatives cannot be exactly replicated

Pricing and Hedging in an Incomplete Market

- In an incomplete market, we have multiple risk-neutral measures
- So, multiple derivative prices (each consistent with no-arbitrage)
- The market/trader “chooses” a risk-neutral measure (hence, price)
- This “choice” is typically made in ad-hoc and inconsistent ways
- Alternative approach is for a trader to play *Portfolio Optimization*
- Maximizing “risk-adjusted return” of the derivative plus hedges
- Based on a specified preference for trading risk versus return
- This preference is equivalent to specifying a Utility function
- Reminiscent of the Portfolio Optimization problem we’ve seen before
- Likewise, we can set this up as a stochastic control (MDP) problem
- Where the decision at each time step is: *Trades in the hedges*
- So what’s the best way to solve this MDP?

Deep Reinforcement Learning (DRL)

- Dynamic Programming not suitable in practice due to:
 - Curse of Dimensionality
 - Curse of Modeling
- So we solve the MDP with *Deep Reinforcement Learning* (DRL)
- The idea is to use real market data and real market frictions
- Developing realistic simulations to derive the optimal policy
- The optimal policy gives us the (practical) hedging strategy
- The optimal value function gives us the price (valuation)
- Formulation based on [Deep Hedging paper](#) by J.P.Morgan researchers
- More details in the [prior paper](#) by some of the same authors

Problem Setup

- We will simplify the problem setup a bit for ease of exposition
- This model works for more complex, more frictionful markets too
- Assume time is in discrete (finite) steps $t = 0, 1, \dots, T$
- Assume we have a position (portfolio) D in m derivatives
- Assume each of these m derivatives expires in time $\leq T$
- Portfolio-aggregated *Contingent Cashflows* at time t denoted $X_t \in \mathbb{R}$
- Assume we have n underlying market securities as potential hedges
- Hedge positions (units held) at time t denoted $\alpha_t \in \mathbb{R}^n$
- Cashflows per unit of hedges held at time t denoted $Y_t \in \mathbb{R}^n$
- Prices per unit of hedges at time t denoted $P_t \in \mathbb{R}^n$
- PnL position at time t is denoted as $\beta_t \in \mathbb{R}$

States and Actions

- Denote state space at time t as \mathcal{S}_t , state at time t as $s_t \in \mathcal{S}_t$
- Among other things, s_t contains $\alpha_t, \mathbf{P}_t, \beta_t, D$
- s_t will include any market information relevant to trading actions
- For simplicity, we assume s_t is just the tuple $(\alpha_t, \mathbf{P}_t, \beta_t, D)$
- Denote action space at time t as \mathcal{A}_t , action at time t as $\mathbf{a}_t \in \mathcal{A}_t$
- \mathbf{a}_t represents units of hedges traded (positive for buy, negative for sell)
- Trading restrictions (eg: no short-selling) define \mathcal{A}_t as a function of s_t
- State transitions $\mathbf{P}_{t+1} | \mathbf{P}_t$ available from a *simulator*, whose internals are estimated from real market data and realistic assumptions

Sequence of events at each time step $t = 0, \dots, T$

- 1 Observe state $s_t = (\alpha_t, \mathbf{P}_t, \beta_t, D)$
- 2 Perform action (trades) \mathbf{a}_t to produce trading PnL $= -\mathbf{a}_t^T \cdot \mathbf{P}_t$
- 3 Trading transaction costs, eg. $= -\gamma \cdot \text{abs}(\mathbf{a}_t^T) \cdot \mathbf{P}_t$ for some $\gamma > 0$
- 4 Update α_t as: $\alpha_{t+1} = \alpha_t + \mathbf{a}_t$ (force-liquidation at $T \Rightarrow \mathbf{a}_T = -\alpha_T$)
- 5 Realize cashflows (from updated positions) $= X_{t+1} + \alpha_{t+1}^T \cdot \mathbf{Y}_{t+1}$
- 6 Update PnL β_t as:

$$\beta_{t+1} = \beta_t - \mathbf{a}_t^T \cdot \mathbf{P}_t - \gamma \cdot \text{abs}(\mathbf{a}_t^T) \cdot \mathbf{P}_t + X_{t+1} + \alpha_{t+1}^T \cdot \mathbf{Y}_{t+1}$$

- 7 Reward $r_t = 0$ for all $t = 0, \dots, T-1$ and $r_T = U(\beta_{T+1})$ for an appropriate concave Utility function U (based on risk-aversion)
- 8 Simulator evolves hedge prices from \mathbf{P}_t to \mathbf{P}_{t+1}

Pricing and Hedging

- Assume we now want to enter into an incremental position (portfolio) D' in m' derivatives (denote the combined position as $D \cup D'$)
- We want to determine the *Price* of the incremental position D' , as well as the hedging strategy for D'
- Denote the Optimal Value Function at time t as $V_t^* : \mathcal{S}_t \rightarrow \mathbb{R}$
- Pricing of D' is based on the principle that introducing the incremental position of D' together with a calibrated cashflow (Price) at $t = 0$ should leave the Optimal Value (at $t = 0$) unchanged
- Precisely, Price of D' is the value x such that

$$V_0^*((\alpha_0, \mathbf{P}_0, \beta_0 - x, D \cup D')) = V_0^*((\alpha_0, \mathbf{P}_0, \beta_0, D))$$

- This Pricing principle is known as the principle of *Indifference Pricing*
- The hedging strategy at time t for all $0 \leq t < T$ is given by the Optimal Policy $\pi_t^* : \mathcal{S}_t \rightarrow \mathcal{A}_t$

DRL Approach a Breakthrough for Practical Trading?

- The industry practice/tradition has been to start with *Complete Market* assumption, and then layer ad-hoc/unsatisfactory adjustments
- There is some past work on pricing/hedging in incomplete markets
- But it's theoretical and not usable in real trading (eg: Superhedging)
- My view: This DRL approach is a breakthrough for practical trading
- Key advantages of this DRL approach:
 - Algorithm for pricing/hedging independent of market dynamics
 - Computational cost scales efficiently with size m of derivatives portfolio
 - Enables one to faithfully capture practical trading situations/constraints
 - Deep Neural Networks provide great function approximation for RL