

# Stochastic Control for Optimal Trade Order Execution

Ashwin Rao

ICME, Stanford University

February 6, 2020

# Overview

- 1 Trading Order Book and Price Impact
- 2 Definition of Optimal Trade Order Execution Problem
- 3 Simple Models, leading to Analytical Solutions
- 4 Real-World Considerations, Extensions, Reinforcement Learning

# Trading Order Book (TOB)



# Basics of Trading Order Book (TOB)

- Buyers/Sellers express their intent to trade by submitting bids/asks
- These are Limit Orders (LO) with a price  $P$  and size  $N$
- Buy LO  $(P, N)$  states willingness to buy  $N$  shares at a price  $\leq P$
- Sell LO  $(P, N)$  states willingness to sell  $N$  shares at a price  $\geq P$
- Trading Order Book aggregates order sizes for each unique price
- So we can represent with two sorted lists of (Price, Size) pairs

Bids:  $[(P_i^{(b)}, N_i^{(b)}) \mid 1 \leq i \leq m], P_i^{(b)} > P_j^{(b)} \text{ for } i < j$

Asks:  $[(P_i^{(a)}, N_i^{(a)}) \mid 1 \leq i \leq n], P_i^{(a)} < P_j^{(a)} \text{ for } i < j$

- We call  $P_1^{(b)}$  as simply *Bid*,  $P_1^{(a)}$  as *Ask*,  $\frac{P_1^{(a)} + P_1^{(b)}}{2}$  as *Mid*
- We call  $P_1^{(a)} - P_1^{(b)}$  as *Spread*,  $P_n^{(a)} - P_m^{(b)}$  as *Market Depth*
- A Market Order (MO) states intent to buy/sell  $N$  shares at the *best possible price(s)* available on the TOB at the time of MO submission

# Trading Order Book (TOB) Activity

- A new Sell LO  $(P, N)$  potentially removes best bid prices on the TOB

$$\text{Removal: } [(P_i^{(b)}, \min(N_i^{(b)}, \max(0, N - \sum_{j=1}^{i-1} N_j^{(b)}))) \mid (i : P_i^{(b)} \geq P)]$$

- After this removal, it adds the following to the asks side of the TOB

$$(P, \max(0, N - \sum_{i: P_i^{(b)} \geq P} N_i^{(b)}))$$

- A new Buy MO operates analogously (on the other side of the TOB)
- A Sell Market Order  $N$  will remove the best bid prices on the TOB

$$\text{Removal: } [(P_i^{(b)}, \min(N_i^{(b)}, \max(0, N - \sum_{j=1}^{i-1} N_j^{(b)}))) \mid 1 \leq i \leq m]$$

- A Buy Market Order  $N$  will remove the best ask prices on the TOB

$$\text{Removal: } [(P_i^{(a)}, \min(N_i^{(a)}, \max(0, N - \sum_{j=1}^{i-1} N_j^{(a)}))) \mid 1 \leq i \leq n]$$

# Price Impact and Trading Order Book Dynamics

- We focus on how a Market order (MO) alters the TOB
- A large-sized MO often results in a big *Spread* which could soon be replenished by new LOs, potentially from either side
- So a large-sized MO moves the Bid/Ask/Mid (*Price Impact* of MO)
- Subsequent Replenishment activity is part of *TOB Dynamics*
- Models for TOB Dynamics can be quite complex
- We will cover a few simple Models in this lecture
- Models based on how a Sell MO will move the TOB *Bid Price*
- Models of Buy MO moving the TOB *Ask Price* are analogous

# Optimal Trade Order Execution Problem

- The task is to sell a large number  $N$  of shares
- We are allowed to trade in  $T$  discrete time steps
- We are only allowed to submit Market Orders
- We consider both *Temporary* and *Permanent* Price Impact
- For simplicity, we consider a model of just the *Bid Price* Dynamics
- Goal is to maximize Expected Total Utility of Sales Proceeds
- By breaking  $N$  into appropriate chunks (timed appropriately)
- If we sell too fast, we are likely to get poor prices
- If we sell too slow, we risk running out of time
- Selling slowly also leads to more uncertain proceeds (lower Utility)
- This is a Dynamic Optimization problem
- We can model this problem as a Markov Decision Process (MDP)

# Problem Notation

- Time steps indexed by  $t = 1, \dots, T$
- $P_t$  denotes Bid Price at start of time step  $t$
- $N_t$  denotes number of shares sold in time step  $t$
- $R_t = N - \sum_{i=1}^{t-1} N_i$  = shares remaining to be sold at start of time step  $t$
- Note that  $R_1 = N, N_T = R_T$
- Price Dynamics given by:

$$P_{t+1} = f_t(P_t, N_t, \epsilon_t)$$

where  $f_t(\cdot)$  is an arbitrary function incorporating:

- Permanent Price Impact of selling  $N_t$  shares
  - Impact-independent market-movement of Bid Price over time step  $t$
  - $\epsilon_t$  denotes source of randomness in Bid Price market-movement
- Sales Proceeds in time step  $t$  defined as:

$$N_t \cdot Q_t = N_t \cdot (P_t - g_t(P_t, N_t))$$

where  $g_t(\cdot)$  is an arbitrary func representing Temporary Price Impact

- Utility of Sales Proceeds function denoted as  $U(\cdot)$



# Markov Decision Process (MDP) Formulation

- Order of MDP activity in each time step  $1 \leq t \leq T$ :
  - Observe *State*  $:= (t, P_t, R_t)$
  - Perform *Action*  $:= N_t$
  - Receive *Reward*  $:= U(N_t \cdot Q_t) = U(N_t \cdot (P_t - g_t(P_t, N_t)))$
  - Experience Price Dynamics  $P_{t+1} = f_t(P_t, N_t, \epsilon_t)$
- Goal is to find a Policy  $\pi^*(t, P_t, R_t) = N_t$  that maximizes:

$$\mathbb{E}\left[\sum_{t=1}^T \gamma^t \cdot U(N_t \cdot Q_t)\right] \text{ where } \gamma \text{ is MDP discount factor}$$

# A Simple Linear Impact Model with No Risk-Aversion

- We consider a simple model with Linear Price Impact
- $N, N_t, P_t$  are all continuous-valued ( $\in \mathbb{R}$ )
- Price Dynamics:  $P_{t+1} = P_t - \alpha N_t + \epsilon_t$  where  $\alpha \in \mathbb{R}_{\geq 0}$
- $\epsilon_t$  is i.i.d. with  $\mathbb{E}[\epsilon_t | N_t, P_t] = 0$
- So, Permanent Price Impact is  $\alpha N_t$
- Temporary Price Impact given by  $\beta N_t$ , so  $Q_t = P_t - \beta N_t$  ( $\beta \in \mathbb{R}_{\geq 0}$ )
- Utility function  $U(\cdot)$  is the identity function, i.e., no Risk-Aversion
- MDP Discount factor  $\gamma = 1$
- This is an unrealistic model, but solving this gives plenty of intuition
- Approach: Define Optimal Value Function & invoke Bellman Equation

# Optimal Value Function and Bellman Equation

- Denote Value Function for policy  $\pi$  as:

$$V^\pi(t, P_t, R_t) = \mathbb{E}_\pi \left[ \sum_{i=t}^T N_i (P_i - \beta N_i) \mid (t, P_t, R_t) \right]$$

- Denote Optimal Value Function as  $V^*(t, P_t, R_t) = \max_\pi V^\pi(t, P_t, R_t)$
- Optimal Value Function satisfies the Bellman Equation ( $\forall 1 \leq t < T$ ):

$$V^*(t, P_t, R_t) = \max_{N_t} (N_t (P_t - \beta N_t) + \mathbb{E}[V^*(t+1, P_{t+1}, R_{t+1})])$$

$$\text{Note: } V^*(T, P_T, R_T) = R_T (P_T - \beta R_T)$$

- From the above, we can infer  $V^*(T-1, P_{T-1}, R_{T-1})$  as:

$$\max_{N_{T-1}} \{ N_{T-1} (P_{T-1} - \beta N_{T-1}) + \mathbb{E}[R_T (P_T - \beta R_T)] \}$$

$$= \max_{N_{T-1}} \{ N_{T-1} (P_{T-1} - \beta N_{T-1}) + \mathbb{E}[(R_{T-1} - N_{T-1})(P_T - \beta(R_{T-1} - N_{T-1}))] \}$$

$$= \max_{N_{T-1}} \{ N_{T-1} (P_{T-1} - \beta N_{T-1}) + (R_{T-1} - N_{T-1})(P_{T-1} - \alpha N_{T-1} - \beta(R_{T-1} - N_{T-1})) \}$$

# Optimal Policy & Optimal Value Function for case $\alpha \geq 2\beta$

$$= \max_{N_{T-1}} \{ R_{T-1} P_{T-1} - \beta R_{T-1}^2 + (\alpha - 2\beta)(N_{T-1}^2 - N_{T-1} R_{T-1}) \}$$

- For the case  $\alpha \geq 2\beta$ , we have the trivial solution:  $N_{T-1}^* = 0$  or  $R_{T-1}$
- Substitute  $N_{T-1}^*$  in the expression for  $V^*(T-1, P_{T-1}, R_{T-1})$ :

$$V^*(T-1, P_{T-1}, R_{T-1}) = R_{T-1}(P_{T-1} - \beta R_{T-1})$$

- Continuing backwards in time in this manner gives:

$$N_t^* = 0 \text{ or } R_t$$

$$V^*(t, P_t, R_t) = R_t(P_t - \beta R_t)$$

- So the solution for the case  $\alpha \geq 2\beta$  is to sell all  $N$  shares at any one of the time steps  $t = 1, \dots, T$  (and none in the other time steps) and the Optimal Expected Total Sale Proceeds =  $N(P_1 - \beta N)$

# Optimal Policy & Optimal Value Function for case $\alpha < 2\beta$

- For the case  $\alpha < 2\beta$ , differentiating w.r.t.  $N_{T-1}$  and setting to 0 gives:

$$(\alpha - 2\beta)(2N_{T-1}^* - R_{T-1}) = 0 \Rightarrow N_{T-1}^* = \frac{R_{T-1}}{2}$$

- Substitute  $N_{T-1}^*$  in the expression for  $V^*(T-1, P_{T-1}, R_{T-1})$ :

$$V^*(T-1, P_{T-1}, R_{T-1}) = R_{T-1}P_{T-1} - R_{T-1}^2\left(\frac{\alpha + 2\beta}{4}\right)$$

- Continuing backwards in time in this manner gives:

$$N_t^* = \frac{R_t}{T - t + 1}$$

$$V^*(t, P_t, R_t) = R_t P_t - \frac{R_t^2}{2} \left( \frac{2\beta + (T - t)\alpha}{T - t + 1} \right)$$

## Interpreting the solution for the case $\alpha < 2\beta$

- Rolling forward in time, we see that  $N_t^* = \frac{N}{T}$ , i.e., uniformly split
- Hence, Optimal Policy is a constant (independent of *State*)
- Uniform split makes intuitive sense because Price Impact and Market Movement are both linear and additive, and don't interact
- Essentially equivalent to minimizing  $\sum_{t=1}^T N_t^2$  with  $\sum_{t=1}^T N_t = N$
- Optimal Expected Total Sale Proceeds =  $NP_1 - \frac{N^2}{2}(\alpha + \frac{2\beta - \alpha}{T})$
- So, *Implementation Shortfall* from Price Impact is  $\frac{N^2}{2}(\alpha + \frac{2\beta - \alpha}{T})$
- Note that Implementation Shortfall is non-zero even if one had infinite time available ( $T \rightarrow \infty$ ) for the case of  $\alpha > 0$
- If Price Impact were purely temporary ( $\alpha = 0$ , i.e., Price fully snapped back), Implementation Shortfall is zero with infinite time available

# Models in Bertsimas-Lo paper

- [Bertsimas-Lo](#) was the first paper on Optimal Trade Order Execution
- They assumed no risk-aversion, i.e. identity Utility function
- The first model in their paper is a special case of our simple Linear Impact model, with fully Permanent Impact (i.e.,  $\alpha = \beta$ )
- Next, Bertsimas-Lo extended the Linear Permanent Impact model
- To include dependence on Serially-Correlated Variable  $X_t$

$$P_{t+1} = P_t - (\alpha N_t + \theta X_t) + \epsilon_t, X_{t+1} = \rho X_t + \eta_t, Q_t = P_t - (\alpha N_t + \theta X_t)$$

- $\epsilon_t$  and  $\eta_t$  are i.i.d. (and mutually independent) with mean zero
- $X_t$  can be thought of as market factor affecting  $P_t$  linearly
- Bellman Equation on Optimal VF and same approach as before yields:

$$N_t^* = \frac{R_t}{T - t + 1} + h(t, \alpha, \theta, \rho) X_t$$

$$V^*(t, P_t, R_t, X_t) = R_t P_t - (\text{quadratic in } (R_t, X_t) + \text{constant})$$

- Serial-correlation predictability ( $\rho \neq 0$ ) alters uniform-split strategy

# A more Realistic Model: LPT Price Impact

- Next, Bertsimas-Lo present a more realistic model called “LPT”
- *Linear-Percentage Temporary Price Impact* model features:
  - Geometric random walk: consistent with real data, & avoids prices  $\leq 0$
  - % Price Impact  $\frac{g_t(P_t, N_t)}{P_t}$  doesn't depend on  $P_t$  (validated by real data)
  - Purely Temporary Price Impact

$$P_{t+1} = P_t e^{Z_t}, X_{t+1} = \rho X_t + \eta_t, Q_t = P_t(1 - \alpha N_t - \theta X_t)$$

- $Z_t$  is a random variable with mean  $\mu_Z$  and variance  $\sigma_Z^2$
- With the same derivation as before, we get the solution:

$$N_t^* = c_t^{(1)} + c_t^{(2)} R_t + c_t^{(3)} X_t$$

$$V^*(t, P_t, R_t, X_t) = e^{\mu_Z + \frac{\sigma_Z^2}{2}} \cdot P_t \cdot (c_t^{(4)} + c_t^{(5)} R_t + c_t^{(6)} X_t + c_t^{(7)} R_t^2 + c_t^{(8)} X_t^2 + c_t^{(9)} R_t X_t)$$



# Incorporating Risk-Aversion/Utility of Proceeds

- For analytical tractability, Bertsimas-Lo ignored Risk-Aversion
- But one is typically wary of *Risk of Uncertain Proceeds*
- We'd trade some (Expected) Proceeds for lower Variance of Proceeds
- [Almgren-Chriss](#) work in this Risk-Aversion framework
- They consider our simple linear model maximizing  $E[Y] - \lambda \text{Var}[Y]$
- Where  $Y$  is the total (uncertain) proceeds  $\sum_{i=1}^T N_i Q_i$
- $\lambda$  controls the degree of risk-aversion and hence, the trajectory of  $N_t^*$
- $\lambda = 0$  leads to uniform split strategy  $N_t^* = \frac{N}{T}$
- The other extreme is to minimize  $\text{Var}[Y]$  which yields  $N_1^* = N$
- Almgren-Chriss derive *Efficient Frontier* and solutions for specific  $U(\cdot)$
- Much like classical Portfolio Optimization problems

# Real-world Optimal Trade Order Execution (& Extensions)

- Arbitrary Price Dynamics  $f_t(\cdot)$  and Temporary Price Impact  $g_t(\cdot)$
- Non-stationarity/non-linear dynamics/impact require (Numerical) DP
- Frictions: Discrete Prices/Sizes, Constraints on Prices/Sizes, Fees
- Incorporating various markets factors in the State bloats State Space
- We could also represent the entire TOB within the State
- Practical route is to develop a simulator capturing all of the above
- Simulator is a *Market-Data-learned Sampling Model* of TOB Dynamics
- In practice, we'd need to also capture *Cross-Asset Market Impact*
- Using this simulator and neural-networks func approx, we can do RL
- References: [Nevmyvaka, Feng, Kearns; 2006](#) and [Vyetrenko, Xu; 2019](#)
- Exciting area for Future Research as well as Engineering Design