

FE570 Financial Markets and Trading

Lecture 10. Execution Strategies

Steve Yang

Stevens Institute of Technology

11/20/2018

Outline

- 1 Execution Strategies
- 2 Benchmark-Driven Schedules
- 3 Cost-Driven Schedules / Risk-Neutral Framework
- 4 Cost-Driven Schedules / Risk-Averse Framework

Execution Strategies

- Trading strategies that were discussed in former chapters are derived for producing positive returns in round-trip trades (see Jorion 2010). Sometimes, we refer these strategies as *opportunistic algorithm*.
- Implementation shortfall* (IS) is a measure of the total transaction costs. IS represents the difference between the actual portfolio return and the paper estimate of this return at the beginning of trading.

If trading of an order with size X started at price p_0 (*arrival price*) and ended at price p_N , and the order was split into N child orders of size x_k that were filled at price p_k , then

$$\text{IS} = \sum x_k p_k - p_0 \sum x_k + (p_N - p_0)(X - \sum x_k) + C \quad (1)$$

where C is the fixed cost. The first two terms represent execution cost, and the third tells opportunity cost.

Execution Strategies

- Note that not all child orders may be executed during the trading day. For example, submission of child orders may be conditioned on specific price behavior. The unfilled amount, $X - \sum x_k$, determines an *opportunity cost*.
- *Algorithmic trading* is a new field that focuses on making decisions where and how to trade. The professional trading community attributes algorithmic trading primarily to the execution strategies (Johnson 2010).
- The question of *whether* to trade is beyond the scope of our lecture today. It is assumed that the decision to trade a given amount within a given time horizon has been made and we are concerned only with its implementation.
- The decision *where* to trade is important for institutional trading, and modern trading systems often have *liquidity aggregators* that facilitate connections to various sources.

Execution Strategies

- In general, there are two major families of execution algorithms: **benchmark-driven algorithms** and **cost-driven algorithms**.
- Benchmark algorithms are based on some simple measures of market dynamics rather than on explicit optimization protocols.
- Cost-driven algorithms minimize IS and are often named *implementation shortfall algorithms*.
- * Obviously, any execution algorithm addresses the problem of minimizing execution costs. Market impact due to order execution in conditions of limited liquidity is the main culprit of trading loss. Large orders can move price in the adverse direction, and a general way of reducing trading loss is splitting large orders into smaller child orders and spanning them over a given time interval.

Benchmark-Driven Schedules

- *Time-weighted average price (TWAP)*. In this schedule, child orders are spread uniformly over a given time interval. Such a simple protocol has a risk of exposure of the trader's intentions to other market participants.

For example, some *scalpers* may realize that a large order is being traded and start trading the same instrument in expectation that the large trading volume will inevitably move the price.

To prevent information leak, TWAP schedule may be randomized in terms of size and submission time of child orders. Then, periodic execution benchmarks are implemented for following the average schedule.

For example, if the trading interval is four hours, 25% of the trading volume must be executed each hour, and the child order size may be adjusted deterministically for each hour.

More sophisticated TWAP schedules may use adaptive algorithms based on short-term price forecast.

- **Volume-weighted average price (VWAP).** Markets often have pronounced intraday trading volume patterns. Therefore, the VWAP schedule may be more appropriate than the TWAP schedule.

If an asset during some time interval has N trades with price p_k and volume v_k , its VWAP is



$$\text{VWAP} = \frac{\sum_{k=1}^N v_k p_k}{\sum_{k=1}^N v_k} \quad (2)$$

Practical implementation of the VWAP algorithm involves calculation of the percentage of daily trading volume u_k for each trading period k using historical market data:

$$u_k = v_k / \sum_{i=1}^N v_i, \text{ the size of } k\text{-th child order } x_k = X u_k \quad (3)$$

- Historical estimates of u_k may have significant variation. Therefore, sophisticated VWAP algorithms have adaptive mechanisms accounting for short-term price trend and dynamics of u_k .
- It should be noted that while the VWAP algorithm helps in minimizing the market impact cost, it does not necessarily yield possible price appreciation, which is, in fact, a form of opportunity cost.

Indeed, if price grows (falls) on a high volume during a day, the trader might get more price appreciation if the entire buy (sell) order is placed in the morning rather than spread over the whole day. On average, however, such an opportunity cost is compensated for buy (sell) orders on days when the price falls (grows).
- The VWAP benchmark has become very popular in post-trade analysis. How well an algorithm performs can be checked by comparing the realized trading cost with the true VWAP calculated using available market data.

Example

TWAP vs. VWAP During a slow trading day, the TWAP may be very similar to the VWAP, even to the penny at times. However, in a volatile session, or when volume is higher than usual, the two indicators may start to diverge.



- **Percent of volume (POV)**. In this schedule, the trader submits child orders with sizes equal to a certain percentage of the total trading volume, γ .

This implies that child orders have acceptable market impact (if any), and execution time is not strictly defined.

In estimating the size of child order x_k , one should take into account that the child order must be included in the total trading volume X_k at time period k :

$$\gamma = x_k / (X_k + x_k)$$

As a result, $x_k = \gamma X_k / (1 - \gamma)$

考

- **Participation weighted price (PWP)**. This benchmark is a combination of VWAP and POV. Namely, if the desirable participation rate is γ and the order volume is N , PWP for this order is VWAP cauculated over N/γ shares traded after the order was submitted.

Cost-Driven Schedules While executing a large order, a risk-averse trader faces a dilemma: Fast execution implies larger child orders and hence higher market impact and higher IS. On the other hand, submitting smaller child orders consumes more time and exposes traders to the price volatility risk (market risk).

- Cost-driven schedules can be partitioned into **risk-neutral algorithms** and **risk-averse algorithms**. In the former case, the schedule is derived by minimizing market impact. In the later case, the schedule is derived by minimizing utility function that has two components: market impact and volatility risk.
- **Risk-Neutral Framework**
Bertsimas & Lo (1998) introduced the following model for optimal execution. The objective is to minimize the execution cost:

- Risk-Neutral Framework

$$\begin{aligned} \min_{x_k} E\left\{\sum_{k=1}^N x_k p_k\right\} \\ \text{s.t. } \sum_{k=1}^N x_k = X \end{aligned}$$

It is assumed that price follows the arithmetic random walk in the absence of market impact, and market impact is permanent and linear upon volume:

$$p_k = p_{k-1} + \theta x_k + \epsilon_k$$

where $\theta > 0$ and ϵ_k is an IID process that is uncorrelated with trading and has zero mean. Then, the volume remaining to be bought, w_k can be determined as a dynamic programming problem.

- **Risk-Neutral Framework**

$$w_k = w_{k-1} - x_k, w_1 = X, w_{N+1} = 0$$

The dynamic programming optimization is based on the solution optimal for the entire sequence $\{x_1^*, \dots, x_N^*\}$ must be optimal for the subset $\{x_k^*, \dots, x_N^*\}$, $k > 1$. This property is expressed in the Bellman equation in recursive format:

$$V_k(p_{k-1}, w_k) = \min E\{p_k x_k + V_{k+1}(p_k, w_{k+1})\}, \text{ and } \{x_k\}$$

It follows from the boundary condition $w_{N+1} = 0$ that $x_T^* = w_T$. Then, the Bellman equation can be solved recursively: first by going backward and retrieving the relationship between x_k^* and w_k , and then by going forward, beginning with the initial condition $w_1 = X$.

It turns out a simple and rather trivial solution: $x_1^* = \dots = x_N^*$.

● Risk-Neutral Framework

Discussions:

- This result is determined by the model assumption that the permanent impact does not depend on either price or the size of the unexecuted order.
- More complicated models generally do not have an analytical solution. Yet, they can be analyzed using numerical implementation of the dynamic programming technique.
- Obizhaeva & Wang (2005) expanded this approach to account for exponential decay of market impact.
- Gatheral (2009) described the relationship between the shape of the market impact function and the decay of market impact. In particular, Gatheral has shown that the exponential decay of market impact is compatible only with linear market impact.

Risk-Averse Framework:

- The *risk-averse framework* for optimal execution was introduced by Grinold & Kahn (2000). Almgren & Chriss (2000) expanded this approach by constructing *the efficient trading frontier* in the space of possible execution strategies.
 - Let's apply the Almgren-Chriss model to the selling process (the buying process is assumed to be symmetrical). Our goal is to sell X units within the time interval T .
Let's divide T into N periods with length $\tau = T/N$ and define discrete times $t_k = k \cdot \tau$ where $k = 0, 1, \dots, N$. Furthermore, let's introduce a list $n = \{n_0, \dots, n_N\}$, where n_i is the number of units sold during the interval $t_{i-1} < t \leq t_i$.
 - Another list will also be used: $x = \{x_0, \dots, x_N\}$, where x_k is the remaining number of units at time t_k to be sold; $x_0 = X$; $x_N = n_0 = 0$

$$x_k = X - \sum_{i=1}^{i=k} n_i = \sum_{i=k+1}^{i=N} n_i$$

Risk-Averse Framework:

- It is assumed that price S follows the arithmetic random walk with no drift. Another assumption is that market impact can be partitioned into the permanent part that lasts the entire trading period T , and the temporary part that affects price only during one time interval τ . Then,

$$S_k = S_{k-1} + \sigma\tau^{1/2}d\xi_1 - \tau g(n_k/\tau)$$

where the function $g(n_k/\tau)$ describes the permanent market impact. The temporary market impact contributes only to the sale price of the order k

$$\hat{S}_k = S_{k-1} + \sigma\tau^{1/2}d\xi_1 - \tau h(n_k/\tau)$$

but does not affect S_k . And the total trading cost equals:

$$IS = XS_0 - \sum_{k=1}^N n_k \hat{S}_k$$

Risk-Averse Framework:

Again the total trading cost equals:

$$\begin{aligned}
 IS &= XS_0 - \sum_{k=1}^N n_k \hat{S}_k \\
 &= - \sum_{k=1}^N x_k (\sigma \tau^{1/2} d\xi_k - \tau g(n_k/\tau)) + \sum_{k=1}^N n_k h(n_k/\tau)
 \end{aligned}$$

Winthin these assumptions, the expected IS, $E(x)$ and its variance, $V(x)$, equal

$$E(x) = \sum_{k=1}^N \tau x_k g(n_k/\tau) + \sum_{k=1}^N n_k h(n_k/\tau)$$

$$V(x) = \sigma^2 \tau \sum_{k=1}^N x_k^2$$

The Almgren-Chriss framework minimizes: $U = E(x) + \lambda V(x)$

Risk-Averse Framework

- Both permanent and temporary market impacts are assumed to be linear on order size:

$$\begin{aligned}g(n_k/\tau) &= \gamma n_k/\tau \\ h(n_k/\tau) &= \epsilon \text{sign}(n_k) + \eta n_k/\tau\end{aligned}$$

Here, γ and η are constant coefficients, ϵ is fixed cost (fees, etc.), and sign is the sign function. Then,

$$E(x) = \frac{1}{2}\gamma X^2 + \epsilon X + \frac{\tilde{\eta}}{\tau} \sum_{k=1}^N n_k^2, \tilde{\eta} = \eta - \gamma\tau/2$$

- Minimization of the utility function is then reduced to equating zero to $\delta U/\delta x_k$, which yields

$$\begin{aligned}x_{k-1} - 2x_k + x_{k+1} &= \tilde{\kappa}^2 \tau^2 x_k \\ \text{with } \tilde{\kappa}^2 &= \lambda \sigma^2 / \tilde{\eta}\end{aligned}$$

Risk-Averse Framework

- The solution to the above formulation is

$$x_k = X \frac{\sinh(\kappa(T - t_k))}{\sinh(\kappa T)} , k = 0, 1, \dots, N$$

Then, it follows from the definition $n_k = x_k - x_{k-1}$ that

$$n_k = 2X \frac{\sinh(\kappa\tau/2)}{\cosh(\kappa T)} \cosh(\kappa(T - t_{k-1/2})) , k = 1, \dots, N$$

where $t_{k-1/2} = (k - 1/2)/\tau$ and κ satisfies the following relation

$$2(\cosh(\kappa\tau) - 1) = \tilde{\kappa}^2 \tau^2$$

When τ approaches zero, $\tilde{\eta} \rightarrow \eta$ and $\tilde{\kappa}^2 \rightarrow \kappa^2$. Note that κ is independent of T and characterizes exponential decay of the size of sequential child orders. Obviously, the higher is risk aversion λ , the shorter is the order's half-life.

● Risk-Averse Framework

Discussions:

- Almgren & Chriss (2000) define the efficient trading frontier as the family of strategies that have minimal trading cost for a given cost variance, that is , a curve in the space E-V.
- Recent extension of the Almgren-Chriss framework by Huberman & Stahl (2005), Almgren & Lorenz (2007), Jondeau et al.(2008), and Shied & Schöneborn (2009) have led to models that account for time-dependent volatility and liquidity, sometimes within the continuum-time framework.
- All these extended models generally share the assumption that market impact can be represented as a combination of the permanent and short-lived transitory components.
- Bouchaud et al. (2004) and Schmidt (2010) exhibit the power-law decay of market impact.