Chapter 18 – Executing and Monitoring High-Frequency Trading

18.1 Introduction

18.2 Executing high-frequency trading systems

Overview of Execution Algorithms

- Broker-Dealer는 가격에 영향을 적게 미치도록 오더를 split하는 best execution service를 제공함
- 알고리즘 트레이딩의 등장은 기관투자자들에 미치는 Broker-Dealer의 영향을 최소화함
- 알고리즘 거래는 여러 가지를 최적화 : **order type, size, frequency, liquidity supply, hedge positions …**
- **Trading-aggressiveness selection algorithm** : choose between market and limit order

  Price-scaling strategies : select the best execution price according to benchmark

  Size-optimization algorithms : determine optimal ways to break down large trading lots

Market-Aggressiveness Selection

- HFT상황에서 Aggressive execution의 market impact는 커진다.
- Almgren and Chriss(1999) model

$$\min_{\alpha} \; Cost\,(\alpha) + \lambda\, Risk(\alpha)$$

  (α : percentage of volume(POV), or liquidity, λ : the coefficient of risk aversion of investor)

- Kissell and Malamut(2005) : market aggressiveness(POV or α) is a combination of market and limit order. Market order tend to increase the POV and α, whereas limit orders decrease market aggressiveness.

$$\text{Cost}(\alpha) = E_0[P(\alpha) - P_b], \quad Risk(\alpha) = \sigma\big(\varepsilon(\alpha)\big),$$

$$P(a) = P + f(X, \alpha) + g(X) + \varepsilon(\alpha)$$

(E_0 : ex-ante expectation at the start of the trading period,

P_b : benchmark execution price

$\mathrm{P}(\alpha)$: the realized execution price defined in upper equation

$\varepsilon(\alpha)$: a random deviation ($\mathrm{E}(\varepsilon(\alpha)] = 0,\ Var[\varepsilon(\alpha)] = \sigma^2(\alpha)$)

P : the market price at the time of order entry

$f(X, \alpha)$: a temporary market impact due to the liquidity demand of trading

$\mathrm{g}(\mathrm{X})$: the permanent price impact due to information leakage during order execution)

Price-Scaling Strategies

- Best price에서 거래하기 위한 전략, Best price는 benchmark와 given utility function이랑 관련되어 있음. Ex) average daily price

- **Strike algorithm**

  : designed to capture gains in periods of favorable prices

  ➔ Aggressive (market order) in favorable prices and passive (limit order) in times of unfavorable prices

$$\min_{a_t} E_t\left[P_{t+1}(a_t) - P_{b,t}\right]^2$$

($P_{t+1}(\alpha_t)$: realized price obtained using the trading aggressiveness level α_t decided upon at time t

$P_{b,t}$: the benchmark price at time t)

  ➔ delivers a lower average cost / but ignores participation in favorable price condition

- **Plus algorithm**

  : maximize the probability of outperforming a specified benchmark while minimizing risk

$$\max_{a_t} \frac{E_t\left[P_{t+1}(a_t) - P_{b,t}\right]}{\left(V\left(P_{t+1}(\alpha_t) - P_{b,t}\right)\right)^{1/2}}$$

($P_{t+1}(\alpha_t)$: realized price obtained using the trading aggressiveness level α_t decided upon at time t

$P_{b,t}$: the benchmark price at time t)

  ➔ Deliver a low average cost, but increase the risk of unfavorable price

- **Wealth algorithm**

  : maximize investor wealth in the presence of uncertainty

  ➔ For preserving the investor's wealth in adverse conditions, wealth algorithm is passive in favorable prices but acts aggressively during periods of unfavorable prices

$$\max_{a_t} \log E_t[U(P_{t+1}(a_t))], \quad \mathrm{U(x)} = E[x] - \lambda V[x]$$

(x: the realized payoff, λ: the risk aversion coefficient of the investor, U(-): utility function)

  ➔ Capture a greater proportion of favorable price conditions but at the expense of higher average prices

Slicing Large Orders

- Informed trader는 다른 참여자들에게 order flow를 간파 당하지 않는 것이 중요하다.
- Slicing large order의 목표는 minimize other trading participants' ability to distinguish these orders from other, "noise" trader.
- Chakravarty(2001) : medium-sized orders indeed are followed by disproportionally large price changes, relative to all price and overall proportion of trades and volumes.
- Order fill rate : the proportion of the order that was "hit" or executed.

Order-fill rate	Non-Sliced Order	Sliced Order
Overall order	40%	48%
Limit order	42%	77%

- Sliced orders are executed more quickly.

  ➔ Fully filled sliced order (3m 20sec) < regular order(11m 54sec)

- Execution cost : 1) average transaction costs, 2) risks associated with execution

- Risk embedded in execution

  1) The uncertainty of the price at which market orders are executed

     ■ Insufficient market depth at a reasonable range of prices for market order

  2) The uncertainty in the timing of the execution of limit orders and the associated opportunity cost

     ■ Possible failure to execute a limit order

- 3가지 고려 (for minimizing market impact)

  1. 위험 중립적 포트폴리오가 현재 위험 수준에서 optimal한지
  2. 리스크가 several financial instruments를 통해 다변화될 수 있는가
  3. Execution risk를 헤지할 수 있는가

- Liquidity has been shown to be time varying, yet persistent from one period to the next.

- **Speed of recovery** : a measure of how fast the limit order book absorbs the market impact generated by the previous lot in the execution sequence.

  ➔ Uniformed speed of recovery : spacing of smaller lots

  ➔ Nonuniformed speed of recovery : larger lots should be processed at times with higher speeds of recovery

- Nevmyvaka, Kearns, Papandreou and Sycara(2006)

  : Optimize execution through a dynamic combination of market and limit orders.

  1) Market order of V at the beginning

  ➔ Need to explore the depth of the book at suboptimal prices and wide bid-ask spreads

  2) Market order of V at the end

  ➔ Subject to market volatility risks

  3) Limit order of V at the beginning and market order for the unexecuted shares at the end

: Best strategy is 3), with limit orders(besting the market price by one tick size) placed at the beginning of trading period

18.3 Monitoring High-Frequency Executions

➔ Pre-trade analysis와 run-time performance를 지속적으로 비교하는 과정

Pre-Trade Analysis

- Estimate expected execution costs and risks

  : Desired price, insufficient liquidity and system breakdown….

  ➔ 위의 것을 고려해서 run-time stop-gain and stop-loss parameters를 정의해야 한다.

- Allowable deviation 정의

  1. Allowable deviations in price of the traded instrument
  2. Allowable deviations in market volume or security volume
  3. Maximum allowable trade duration

Monitoring Run-Time Performance

- HFT 특성 상, deviation에 취약하므로 run-time monitoring이 필요함

  1. **Allowable deviation in price** ensures that the execution is suspended whenever the market impact costs become too high
  2. Processing market orders in high-volume conditions limits the market impact of the strategy and increase profitability. (**allowable deviations in security volume**에 대한 고려)
  3. **Maximum allowable duration** reduces the risk of non-execution

  ➔ The longer the limit orders have been outstaSnding, the higher is the probability that the market price has moved away from the limit order prices (increasing the risk of non-execution)