Individual Book Report

Chongdan Pan

SUMMARIZATION

Chapter 1: Introduction

We can classify algorithms by how they organize orders. Single-order algorithms work on each order independently, while list-based algorithms will trade each order based on other orders. We can classify by function, too. Typically, algorithmic strategy will work on how to split an order while smart routers strategy aims at finding a venue for the best price. By working approach, we can also classify algorithms into three categories. Scheduling algorithms such as TWAP, VWAP or IS use schedules to place orders. Dynamic algorithms such as POV or VP update schedules based on market conditions. Opportunistic algorithms which trade when opportunities arise, are suited for large orders. Their orders will still follow the schedule.

Algorithms also have other components, which will firstly decide how to split the parent orders into child orders, or set what orders to be hidden or reserved. Then they will set the price and volume of each order before smart routers select venue for orders

Chapter 2: Trading Terminology

Orders with ticker, side, price, quantity are sent to a trading venue. Quotes are posted orders, showing traders' attitudes to the price level. Bid-ask spread and market depth are defined from the distance between quotes. Trade occurs when there's a match within quotes. Tick and lot size are increments of order's price or size. However, tick size is usually the minimum increment of price while lot size can be larger than the minimum increment of quantity. Besides normal trading, auctions where all orders are executed at the same price, are usually used to determine the open and close price.

Trading venues can be lit venues like Electronic Communication Networks (ECNs) or dark pools.At ECN, we can get information about some orders from the orderbook while in dark pools, we can't view any information about orders. Dark pools are under controversy due to lack of regulation, but they provide diversity for traders. Trading venues will charge fees for participants. In the typical "maker-taker" model, market makers will earn a fee for providing liquidity. "Taker-maker" model is being applied to discourage traders to place limit-orders and prevent the queues from being too long.

Price-time rules are most frequently used, but there are other variants. Price-size priority incentivizes traders to submit orders with larger size, leading to a deeper market, but traders may inflate quotes artificially to jump the queue. Price-broker-time gives a broker's clients' priority, incentivizing brokers to route all orders to exchanges instead of making trade in place. We may also determine priority by the publicity.

Chapter 3: Trading Costs, Alpha, and Risk

Trading costs, alpha and risks are three primary considerations for trading strategies. There are trade-offs in the considerations and traders need to strike the optimal balance.

Trading costs can be implicit and explicit. Bid-ask spread is the most obvious cost due to liquidity. It's measured by the difference between bid and ask quotes, which is the cost for buying something and selling it immediately. Due to hidden orders and retail brokers, traders may receive price improvement. The effective spread defined as twice the difference between trade price and midpoint of quoted spread, can take the price improvement into consideration. Market behavior and information leakage may lead to extra cost because people's trade behavior will cause price movement. Opportunity costs are usually caused by missing a trading opportunity, and they're usually overlooked. If we miss a good trading opportunity as scheduled, we're losing the profit due to mistakes. Adverse selection costs usually occur when we can't respond to the market in time, and our previous orders are hit by other fast traders.

Alpha represents the expected price movement over the execution. It only considers the price movement in short-term or micro factors. Latency Arbitrage essentially exploits the alpha from fast feeds of market data. Faster traders identify latent orders and make profits by hitting it, while the other sides' costs are increased due to adverse selection.

The primary risk of an asset can be decomposed into factors, such as market risk and stock-specific risk. PM can decrease the market risk by hedging or diversification. Increasing the trading rate can decrease the stock-specific risk. Based on different risk aversion, traders will have different trading rates to reduce the risk. However, high trading rates can lead to higher trading costs such as fees and the spread.

Chapter 4: Schedule-Based Single-Order Algorithms

TWAP trades at a rate proportional to time. To reduce the trading cost, traders will put some limit orders. Limit orders may increase the tracking error, so we need a band to bound the target trading quantity. When the current quantity is close to lower bound, traders will send a marketable order, and vice versa. The width of the band defines the maximum trading quantity, which is important when considering market impact.

VWAP's trade rate is proportional to historical trading volume. It segments the trading hours into time intervals with same lengths, and the target quantity in each interval is proportional to the relative historical trading volume. In each time interval, VWAP works just like TWAP, but it'll have a larger curvature for a whole day. We'll fit a continuous function for a stock's volume, and even group similar assets to smooth the function.

IS dominates other algorithms since it follows the "buy low, sell high" intuition. We usually compare it to a hypothetical portfolio performance based on arrival price. IS aims at minimizing an objective function composed of execution cost, risk penalty and alpha loss. The three parts are hard to estimate and depend on various factors like asset characteristics, market conditions and traders' risk aversion. The risk aversion can be a parameter measuring how much costs we want to take to reduce risk by one unit. Different risk aversion may lead to different optimal trading horizons.

Close algorithm is using the closing price instead of the arrival price in IS, since we have zero cost and risk in closing the auction. It will send the entire order to the closing auction though some executed orders will be back loaded. Since the order may be large, we need to consider its influence on the closing price and performance. The algorithm may be particularly useful to maintain the balance between cash and assets for funds.

Chapter 5: Dynamic Single-Order Algorithms

POV algorithm trades by following the realized volume of the current market. Traders need to set a participation rate, which is the rate between the volume of executed trades and all volume in the market. To increase the tracking accuracy, the schedule is updated on a tick-by-tick basis and uses market orders. POV also has a band for tolerance, and it needs to take its own impact on the market into consideration. The algorithm can naturally reduce risk because it trades more aggressively when volume is high,but it essentially responds to the market instead of participating in it.

Forecasted POV follows forecasted volume instead of realized volume to keep the participation rate. It will update the forecasted market trade volume by the current market behavior and calculate the trading end time based on it. Since the algorithm sets the end time and volume ahead of the current market, its behavior is not lagged.

Optimal POV algorithms identify the optimal trading rate based on the current market. The target participation rate is still required, but the algorithm allows for wider deviations. What's more, the algorithms have a "must complete" option, allowing urgent manual order. In this case, it'll switch to TWAP or VWAP to ensure the completion.

Chapter 6: Opportunistic Single-Order Algorithms

Opportunistic algorithms are most ill-defined because it's quite subjective to say what an opportunity is. However, they share the same objective of buying low and selling high.

“Hide and Take” algorithms hold some limit or hidden order in reverse when there is no good opportunity to hide interest. The algorithm will pop up and throw market orders to make a trade when there’s an opportunity. The algorithm has better performance on illiquid assets than scheduled algorithms since it has a lower risk. When the opportunity is rare, traders may still force the algorithm to trade at a minimum participation rate.

Adaptive Arrival Price algorithm identifies opportunities based on the short-term price movement of the asset. It may mimic the behavior of a trader who is selling high and buying low, but it may also be consistent with a momentum algorithm, which buys when the pricing is rising and sells when falling. The relative difference between the current price and target price can also be used to identify the opportunity. Besides the price, volume and turnover rate can be important and useful for opportunity identification, too.

Chapter 7: Practical Complications in Single-Order Algorithm Development

Edge cases are what every trader should be aware of. Different user inputs, parameters and market behavior may lead to ill-defined or unwanted behavior of the algorithm. Therefore, the developer needs to carefully explore all possible scenarios to identify edge cases. Designing a rigorously testing algorithm can be helpful.

Auction is a special part of trading, and it’s hard to estimate the trading volume or price before the completion of it. For VWAP, we typically trade by estimating the volume in auction, while for TWAP, we usually manually set a target volume. POV won’t participate because we can’t get the volume.

The contradiction between price limit and target volume can be an edge case. Due to the price limit forced by traders, it may be hard for the algorithm to stay in the volume band. We need to set a priority for the algorithm to make a choice. When the limit is canceled, the algorithm may behave extremely aggressively and impact the market. Hence, we need to smooth trading volume to reduce it.

Setting volume constraints can be useful to prevent huge market impact. With volume constraints, we can switch between POV and VWAP based on the settings. The volume constraints may be triggered by the volume of the market, but for an illiquid market, the algorithm may behave badly since the calculation is lagged.

Clean up is important and hard to deal with due to limited time. Latency and volatility may prevent the algorithm from completing orders with an ideal price. The auction also makes the market unpredictable. The algorithm should start cleaning up before the final auction. We usually place a marketable limit order to control the trading cost.

Chapter 8: Multi-Order Algorithms

Pair Trading is one of the simplest multi-order algorithms because it only trades two assets. We prefine the ideal ratio of the prices of two assets, and a deviation of the realized ratio triggers us to enter into a position, Our trigger should have a buffer on either side due to trading costs. As the prices revert back, we need to close out our position, which should be closer to the ideal ratio. To prevent the algorithm from trading an unbounded position, we also need to define a maximum position. When we're triggered to enter the position, we usually first trade the illiquid asset with limit orders to limit the cost. The money that we put into both assets should be as equal as possible hence we want to set a small size for each order to reduce the capital difference.

Portfolio trading algorithms are generally multi-order extensions of scheduled-based single-order SI algorithms and share the same goal, which is to balance the cost and portfolio risk of each order, and find the optimal schedule to minimize the total trading penalty.The only difference in trading penalty is trading risk, because we need to add correlation terms within any two assets to the risk. The sign of the correlation depends on the orders' direction or the assets, hence we can hedge through negative correlation. We can also utilize cross-order price impacts to gain extra alpha. However, it doesn't necessarily mean the portfolio algorithm will have a lower risk, since it focuses on the total penalty including trading cost. Therefore, if we're changing the trading cost or volatility of one asset, the strategy will change for all assets. Compared to the IS algorithm, the algorithm will trade more slowly to reduce costs for negatively correlated orders, but it'll trade liquid and highly volatile assets aggressively to reduce cost and risk.

We can implement the algorithm through VWAP with an optimal volume profile, which is calculated during each horizon. The impact between orders is important, hence we tend to send a sequence of shorter-lived orders instead of a single larger order. The pros lie in various trading styles for each asset, more control on overall constraints and flexible utilization of correlated assets. We can also design the algorithm by heuristics to reduce computational complexity. The algorithm can be used to rebalance the portfolio or find out the optimal quantity within constraints. However, these simpler versions may behave bad for atypical assets or miss liquidity due to traditional heuristic assumptions.

Portfolio algorithms require extra practical considerations. With a larger portfolio, we have a higher computation complexity, hence we need simplifications. When there's a conflict between a human's order, constraints or the algorithm's strategy, the optimization granularity or priority should be adjusted. The intraday in cost, volume, and risk may be random, hence we need to take it into consideration by adding buffers or parameters.We may also shorten the end time to reduce risk and cost.

Chapter 9: Child Order Pricing and Sizing

In general, we need to consider the pricing, sizing and routing of the child orders for submission jointly, but approaching pricing, sizing and routing sequentially can still effectively approximate an optimal solution.

There are multiple factors affecting the limit prices, such as whether it needs to be filled immediately. Essentially we're balancing the low cost provided by limit order with liquidity or the risk that the price goes in an unfavorable direction. Fair value is the true economic value of an asset, which can be estimated from analysis of the asset. Edge is the premium we get from trading. The higher the edge is, the harder it is to find a counterparty. By pricing the limit order, we're maximizing expected gain, which is the sum of edge and the benefit of not filling. When trading, we compute the expected gain for each edge level and choose the optimal one. The benefit of not filling includes trading fee and opportunistic cost due to price movement.

Adverse selection can significantly affect the edge we choose, the order we submit and the trades we achieve may both show some information about the asset, leading to a different the fair price. In addition, the price may move naturally, therefore it's critical to adjust orders in time based on the updated fair price. The extreme requirement for speed makes retail investors and human traders inferior.

In practice, we're using the mid price or weighted average prices to estimate the fair value. Weighted average prices may have intuitive behavior or high volatility due to the randomness of size. For edge determination, we can feed models historical orderbook information to calculate the probability. We may also peg our order on the best quotes, but extra protective logic is required since pegged order may be manipulated by other traders due to its stickiness and blindness. Another way is to use average/time-weighted or exponentially weighted historical spread to estimate the edge.

There are other techniques or considerations that we need to be aware of to improve the algorithm, such as how to set the price aggressively when the algorithm falls behind the schedule, or when the algorithm is triggered to evaluate the fair price. We may also incorporate short-term alphas into our estimation for extra profits. We can also exploit special order types or send fleeting orders to achieve specific goals during trading.

For sizing, the algorithm mainly decides the quantity of each order to be displayed at different price levels. Some algorithms may send a lot of orders at multiple passive price levels to get a premium from who's seeking liquidity. These prices also implicitly define the priority of the orders. These orders may effectively attract counterparties but leak the intention of the traders. Hidden orders have lower priority, and trades only happen with them when there is a sweeping or oversize order. Reserve orders are combinations of light and hidden orders, aiming to exploit the advantages of both.

Chapter 10: Smart Routers

Smart Market Order Router is designed to choose the best venues for marketable orders. We need to pay attention to synchronization and coordination since our order at one exchange may cause change in other exchanges, and leak our intentions. Since the quote may change very quickly, the routers will use marketable IOC limit order, which will be canceled back if the price is unfavorable. Before the trade's confirmation, the router will keep finding the best revenues in case the orders are returned. When there are hidden orders, IOC orders can be used to "ping" these venues, and we can get the information if the order is partially filled. To prevent information leakage, we use minimum acceptable quantity (MAQ), so that the order can only be filled when the quantity is large enough.

Smart Limit Order Routers are responsible for routing limit orders to venues which can execute most quickly. Therefore, we need to estimate the queue and trading rate at exchange. For small orders, we prefer venues with short queues, while for large orders, we prefer venues with high execution rates. However, it may be hard to find the optimal value since the queue and trading rate may depend on each other. For estimation, we may use statistics models such as logistic regression or complex machine learning models. In practice, people prefer the empirical model based on machine learning. "Maker-taker" and "taker-maker" exchanges have different mechanisms for trading. The later one is becoming popular due to shorter queues.

Dark Aggregator seeks to find the revenue where order can be executed at the midpoint of the spread. Since we don't know the queue, we can only rely on historical data for estimation. To find the best venue, the dark aggregator will ping all dark pools by sending IOC midpoint orders and see if they're filled. Then, we can estimate the probability of a fill at the venues and post hidden orders to those with the highest probability. If the order is totally filled, we know that there should be an additional quantity left at the price. Due to the limitation of the data we get, reinforcement learning is preferred for routing. Information is valuable in dark pools, hence predator traders may use IOC to uncover larger orders and make a front-run while others may use MAQ for protection. We can also limit information leakage by pricing orders passively, but it also reduces the probability of execution. Dark pools also have methods to prevent adverse selection and information leakage such as trader grouping, trajectory crossing and behavior monitoring. Condition orders can be a useful tool for traders. Since another firm order is required for a matched trade, it enables traders to send orders with larger size to all exchanges. Condition orders may add risks of adverse selection and information leakage which can't be dealt with by MAQ because people are allowed to post orders for probation and ignore the match.

Chapter 11: Performance Measurement

The total performance of the trading can be decomposed into realized performance and unrealized performance depending on a benchmark price of the asset that we're trading. For buyers, the performance is benchmark price minus volume-weighted execution price, and vice versa for sellers. When calculating the execution price, we also need to take trading cost into consideration.The unrealized performance for buyers is benchmark price minus volume-weighted mark-to-market price, which can be close price or the prevailing price at the end of execution. Practitioners will regard unrealized performance as opportunity cost. We can call the performance "cost" by flipping the sign. For multiday orders' performance, we need to evaluate the performance everyday and sum them together since the benchmark price may be different everyday.

Usually we're using multiple benchmark prices to get a better sense. When determining the benchmark, we need to consider the trader's own actions, alpha and the less volatile performance metric caused by the benchmark price. Arrival price is the most commonly benchmark price, and the corresponding cost is called execution shortfall. The execution shortfall may be noisy so we need to control exogenous factors and filter out overnight orders during comparison. VWAP are widely used to evaluate VWAP algorithm's performance or relative performance under certain scenarios. VWAP is calculated from order's submission to execution, hence it can filter out unrelated noise. Compared to arrival prices, VWAP is less noisy but biased because the performance doesn't consider the market impact of our trades. PWP calculates the VWAP based on a hypothetical participation rate. It can be helpful to determine the optimal trading horizon, but it suffers a similar problem as VWAP. Post-Trade price can be the close price or the price X minutes after the trade. Like VWAP, it may suffer from trade's own market impact, hence we need to select the X carefully to balance the bias and variability.

Except for price-based metrics, we can also use participation rate to measure how aggressively we're trading, use parent-level fill rates to check if we're setting a proper limit price, and use child-level fill rates to understand the trade-offs among edge capture, fill rates and adverse selection.

Chapter 12: Performance Measurement Technique and Issues

We usually use basic points as the unit duration comparison because it makes orders on different assets comparable. Absolute performance focuses on the performance in isolation, and it's useful for us to understand what factor leads to low performance. With factors on asset, order and strategy level, we focus on estimating the trading cost by regression. The quality of the trading cost model may be affected by historical data, the model structure and the estimation technique that we're using. Relative performance can be estimated by cost model comparison or direction comparison. In the first method, we're comparing the performance by cost model and the algorithm's performance. In direct comparison, we're just running two algorithms in the market at the same time and comparing their performance. A more common and hybrid way is to compare how two algorithms outperform the performance.

Typically we're measuring the performance based on samples, hence analyzing the sample's characteristics can be helpful for us to identify the key factors. A difference in one characteristic can lead to different conclusions even though all other factors are the same, therefore hybrid cross validation can be useful in practice. When averaging performance over orders, we need to consider the goal of evaluation and statistical efficiency because different methods may mask important differences.

Dollar-weighting averaging is good at measuring the investment performance while order-weighted is preferred to measure the performance on a group of similar orders. To get an accurate estimate, we should choose an appropriate sample size and check if our assumptions are reasonable. Outliers and influential observations may skew our results. Outliers are usually in the tails, and we typically delete them or winsorize them with a new value. Influential observations are essentially affecting the sample structure, and they can be identified manually. If we're seeking accurate past performance or robustness, then we need to keep outliers and influential observations, otherwise we can deal with them before getting a qualitative conclusion.

REFLECTION and CONNECTION

This book is strongly connected to our trading course and very helpful for me to solidify my understanding of trading. Most important concepts that we learn from our courses almost appear in every chapter in the book, such as order book, quotes, dark pools, quotes, and etc. Although the book is about algorithms, there is only a little mathematics and equations, making it easy to read and understand. I think it's the logic that really matters. On the other hand, since the book is a guide written for practitioners, it uses a lot of examples to illustrate what these concepts' role is in practice and why they're so important. For example, after reading some techniques to capture or prevent information leakage from order books, I come to realize that order books are not only used for determining how and when trades happen, but also quite important for identifying trade opportunities.

As a student with an engineering background, I'm always curious about algorithmic trading and hope to build my own trading strategy by myself. Fortunately, this book provides a very detailed introduction about important components of a trading algorithm step by step. Just as our course, the book focuses more on the execution of trading than how to choose assets for investment. Therefore, we should be able to apply these algorithm designs to different investment strategies. Given the characteristics of assets, venue and strategies, I can have a general idea of what algorithm works best and what's the limitation. In my view, I prefer portfolio trading algorithms. Although it's more complex than others, it exploits as much information from the market as possible. The diversity and hedging also make our algorithm more robust when the market is very volatile.

Compared to the general framework of algorithmic trading, the analysis of execution is even extremely useful during our manual trading. By considering the fair value, edge and opportunistic cost, we can decompose our trading or investment actions into various components with different logic. Such decomposition can greatly help us understand what we're really doing, and what we can really improve. As mentioned many times in the book, the goal of algorithmic trading is to buy low and sell high, I believe it's also applicable to manual trading or long-term investment. In the last two chapters, the book introduces some systematic ways to evaluate the performance of algorithms, which may be used for manual trading as well. Although different evaluation methods have different limitations, they provide a fair platform based on which we can make comparisons and identify the pros and cons of the algorithm.

The smart-routers part is quite novel for me because where I come from, we only have one exchange for one asset, so we never consider how to choose the venues. In some way, multiple venues imply that there is an even fierce competition within the exchanges. Even exchanges are trying to exploit different models to attract market-makers and liquidity-takers. In addition, there are other interesting concepts which broaden my views. For instance, it never occurred to me that we can even use IOC order or MAQ to prevent information leakage. Firm orders also have a wide application. As we learned from the articles during class, there is a lot of controversy around these innovative tools, they essentially aim at providing more choices and flexibility for us to improve the market efficiency.

After reading the book, I think the market is even more unpredictable and complex than I thought, which makes it harder to design a robust algorithm. For example, our algorithm should be able to solve the conflict between preset schedule and manually input. To make the algorithm workable, I must think of the edge case and carefully consider all details. The book not only gives us typical methods and designs used for algorithmic trading, but also many important considerations such as market impact, adverse selection. It's interesting to note that the book uses words such as "cost" and "risk" much more often than "profit". I'm surprised at first glance because everyone is seeking to make profit from trading instead of suffering from risks. However, it turns out that we need to balance cost and risk all the time to achieve optimal performance. I guess it's because the market is always highly competitive and everybody is using all their strengths to gain profit from it. As a beginner, the book teaches me to fully understand the risk and cost before I take any action. These important considerations exploit the experience of previous traders and can effectively prevent me from making stupid mistakes during trading.

Although the book is about algorithmic trading, the majority of its content can be applied to other scenarios as well. Its discussion about the market, decision making strategy, systematic performance evaluation can help us understand how humans and the world are interacting with each other. In essence, I think balancing is the key for trading. For each order, we need to balance cost and risk while for the whole algorithm we need to balance the complexity and robustness. Although subjective factors such as personal risk aversion may be involved in the balancing process, we still need a general and systematic way that can lead us to a reasonable and optimal decision. Compared to humans, algorithms won’t balance their emotional feelings when making decisions. The consistency makes it easier for evaluation and further improvement.

Even though it’s extremely difficult to master trading in the chaotic market, the profit from it is considerable, which keeps encouraging talents to devote themselves into it. The book introduces some advanced topics that can play an important role in trading, such as reinforcement learning. It turns out that people are constantly bringing the state-of-art technology into the market for more profit, and conversely trading and market effectively facilitate the development of advanced technology. Therefore, I conclude that trading is not only a process of gaming or redistribution of wealth, but also creates a highly competitive and tempting warzone which stimulates people’s creativity.