Algorithmic Trading

Technology continues to have an increasingly significant impact on how securities are traded in today's markets. Many trading floors have been replaced by electronic trading platforms and more than a third of the trading volume in the United States can be attributed to algorithmic trading. Every large broker–dealer provides algorithmic trading services to their institutional clients in order to assist their trading. The algorithms used by institutional investors, hedge funds, and many other market participants are used to make trading decisions about the timing, price, and size of trades, with the objective of reducing risk-adjusted costs.

In a broad sense, the term *algorithmic trading* is used to describe trading in an automated manner according to a set of rules. It is often used interchangeably with statistical trading or statistical arbitrage, which may or may not be automated, but is based on signals derived from statistical analyses or models. Smart order routing, program trading, and rules-based trading are some of the other terms associated with algorithmic trading. More recently, the range of functions and activities associated with algorithmic trading has grown to include market impact modeling, execution risk analytics, cost aware portfolio construction, and the use of market microstructure effects.

In this article, we first explain the basic ideas of market impact and optimal execution from both the sell- and buy-side perspectives. We then provide an overview of the most popular algorithmic trading strategies.

Market Impact and the Order Book

The limit order book contains resting limit orders. These orders rest in the book and provide liquidity as they wait to be matched with nonresting orders, which represent a demand for liquidity. The three most common types of nonresting orders are marketable limit orders, market orders, and fill-or-kill orders.

The bid side of the limit book contains resting bids to buy a certain number of shares of stock at a certain price. The offer side contains resting offers to sell a certain number of shares of stock at a certain price.

A market order is a demand for an immediate execution of a certain number of shares at the best possible price. To get the best possible price, a market order sweeps through one side of the limit order book—starting with the best price—matching against resting orders until the full quantity of the market order is filled or the book is completely depleted.

Unlike a market order, a marketable limit order can be executed only at a specified price or better. For example, a marketable limit order to buy 100 shares at \$90.01 can match with a resting limit order to sell 200 shares at \$90.00. The trade print—the price at which the trade would take place—would be \$90.00.

The following examples illustrate how market orders to sell interact with resting limit orders to buy.

Figure 1 shows the idealized market impact of a 200-share market order to sell. The x- and y-axes display the time and price, respectively.

The bid side of the limit order book contains bids to buy a certain number of shares of stock at a certain price. Resting limit orders—orders that sit in the order book—are said to *provide liquidity* by mitigating the market impact of orders that must be filled immediately. The state of the book establishes a pretrade equilibrium (1), which is disturbed by a market order to sell 200 shares (2). Market orders must be filled immediately, and therefore represent a demand for liquidity.

As the sell order depletes the bid book by matching with limit orders to buy, it obtains an increasingly less favorable (lower) trade price, resulting in the trade print (3). Assuming no other trading activity, over time, liquidity providers replenish the bid book to (4), which is the posttrade equilibrium.

The difference between (4) and (1) is an information-based effect called *permanent market impact*. It is the market's response to information that a market participant has decided not to own 200 shares of this stock. This effect is typically modeled as immediate and linear in the total number of shares executed. Huberman and Stanzl [9] show that, if the effect were not linear and immediate, buying and selling at two different rates could produce an arbitrage profit.

The difference between (4) and (3) is called *temporary market impact*. The trader who initiated the trade is willing to obtain a less favorable fill price (3) to get his/her trade done immediately. This *cost*

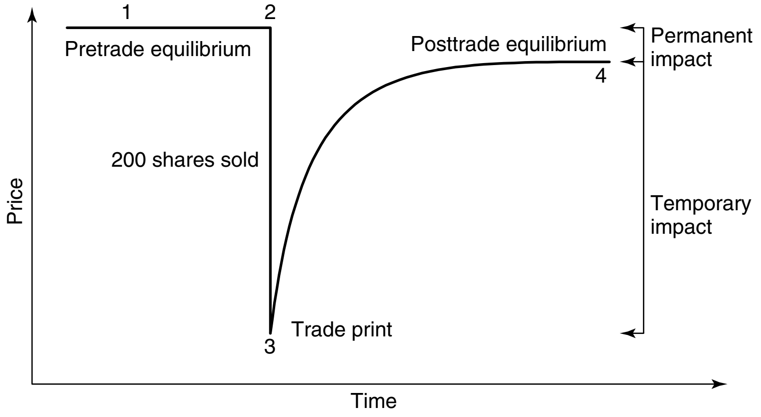

Figure 1 Idealized market impact model showing sell of 200 shares

of immediacy is typically modeled as linear or square root. Under the assumption of square root impact, with all other factors held constant, a trade of 200 shares executed over the same period of time as a trade of 100 shares would have square root of two times more temporary impact per share.

Figure 2 shows what would happen if the same trader were willing to wait some time between trades. The trade print from the previous figure is shown as a reference point (1). As in Figure 1, a pretrade equilibrium (2) is disturbed by a 100-share market order to sell (3). As the market order depletes the bid book by matching with limit orders to buy, it obtains a fill price (4). Over time (5), liquidity providers refill the bid book with limit orders to buy. However, the new posttrade equilibrium (6) is lower than the pretrade equilibrium because it incorporates the information of the executed market order.

Our trader then places another market sell order for 100 shares (6) and obtains a trade print (7). Over time, the temporary impact—(8) minus (7)—decays and results in a new posttrade equilibrium (8). As the permanent impact is assumed to be linear and immediate, the posttrade equilibrium is shown to be the same for one order of 200 shares as it is for two orders of 100 shares each.

Optimal Execution

While our trader waits between trades (5), he/she incurs price risk—the risk that his/her execution will

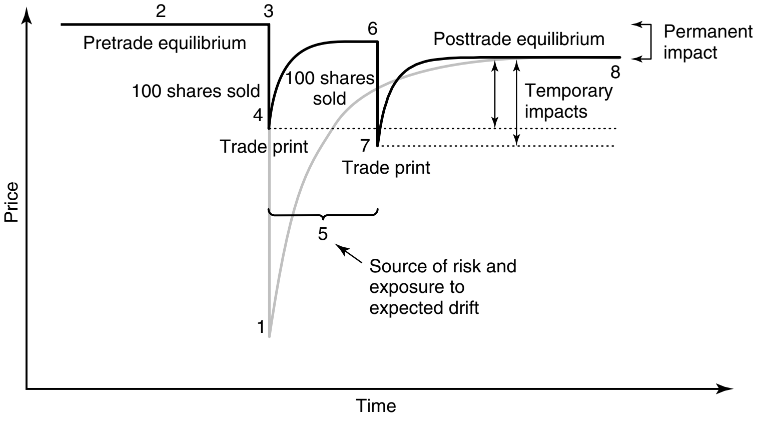

Figure 2 Idealized market impact model showing two sells of 100 shares each

be less favorable due to the random movement of prices. In this context, a *shortfall* is the difference between the effective execution price and the arrival price—the prevailing price at the start of the execution period. If we use the variance of *shortfalls* as a proxy for risk, a trader's aversion to risk establishes a risk/cost trade-off. In the first scenario, he/she pays a higher cost—the difference between (8) and (1)—to eliminate risk. In the second scenario, he/she pays a lower cost—the average of the differences between (8) and (4), and (8) and (7)—but takes on a greater dispersion of shortfalls associated with the waiting time between trades (5). This is the trade-off considered in the seminal paper of Almgren and Chriss [1].

Risk aversion increases a trader's sense of urgency and makes it attractive to pay some premium to reduce risk. The premium the trader pays is in the form of higher temporary market impact. All other factors held constant, a higher expected temporary market impact encourages slower trading, while a higher expected risk or risk aversion encourages faster trading.

Risk aversion embodies the notion that people dislike risk. For a risk-averse agent, the utility of a fair game, $u(G)$, is less than the utility of having the expected value of the game, $E(u(G))$, with certainty. The degree of risk aversion may be captured by the risk aversion parameter λ, which is used to translate risk into a *certain dollar cost equivalent*—the smallest certain dollar amount that would be accepted instead of the uncertain payoff from the fair game. For an agent with quadratic utility, the certain dollar cost equivalent is given by $E(G) - \lambda Var(G)$. Hence, his/her degree of risk aversion is characterized by the family of risk/return pairs with the same constant trade-off between the expected return and risk. An annualized target return and standard deviation imply a risk aversion, and may be translated to a risk aversion parameter of the type used in some optimal execution algorithms.

Another factor that influences the decision to trade more quickly or more slowly is the expectation of price change. For the purpose of execution, a *positive alpha* is an expectation of profits per share per unit time for unexecuted shares. A faster execution captures more of the profits associated with this expectation of price change. A *negative alpha* is the expectation of losses per share per unit time for unexecuted shares. A slower execution incurs less of the losses associated with this expectation of price change.

For example, a trader has positive alpha if he/she expects prices to move lower while he/she is executing his/her sell orders. He/she may choose to front-weight his/her trade schedule—execute more rapidly at the beginning of the execution period—to obtain better execution prices. Similarly, a seller who believes that prices are moving higher may back-weight his/her trade schedule or delay the execution.

The general form of the optimal execution problem involves finding of the best trade-off between the effects of risk, market impact, and alpha by minimizing risk-adjusted costs relative to a prespecified benchmark. Common benchmarks are volume-weighted average price (VWAP) and arrival price (the price prevailing at the beginning of the execution period).

The first formulations of this problem go back to the seminal papers of Bertsimas and Lo [4] and Almgren and Chriss [1]. Assuming a quadratic utility function, a general formulation of this problem takes the form

$$\min_{x_t}[E(C(x_t)) + \lambda Var(C(x_t))] \tag{1}$$

where $C(x_t)$ is the cost of deviating from the benchmark. The solution is given by the trade schedule x_t that represents the number of shares that remains to buy/sell at time t. The trader's optimal trade schedule is a function of his/her level of risk aversion, $\lambda \geq 0$, which determines his/her urgency to trade, and dictates the preferred trade-off between execution cost and risk.

In the following two subsections, we describe the sell-and buy-side perspectives of the typical arrival price optimal execution models.

The Sell-side Perspective

The typical optimal execution model uses arrival price as a benchmark and balances the trade-off between market impact, price risk, and opportunity cost. Alpha is assumed to be greater than or equal to zero, which means that delaying execution may carry an associated opportunity cost, but does not carry an expectation of profit. The optimal strategy lies somewhere between two extremes: (i) trade everything immediately at a known cost or (ii) reduce market impact by spreading the order into smaller trades over a longer horizon at the expense of increased price risk and opportunity cost.

Bertsimas and Lo [4] proposed an algorithm for the optimal execution problem that finds the minimum expected cost of trading over a fixed period of time for a risk neutral trader, $\lambda = 0$, facing an environment where price movements are assumed to be serially uncorrelated.

Almgren and Chriss [1] extended this concept using quadratic utility to embody the trade-off between the expected cost and price risk. The more aggressive (passive) trade schedules incur higher (lower) market impact costs and lower (higher) price risk. Similar to classical portfolio theory, as λ varies the resulting set of points $(Var(\lambda), E(\lambda))$ traces out the *efficient frontier of optimal trading strategies*. The two extreme cases $\lambda = 0$ and $\lambda \to \infty$ correspond to the minimum impact strategy—trading at a constant rate throughout the execution period—and the minimum variance strategy—a single execution of the entire target quantity at the start of the execution period.

Let us consider selling X shares, that is, we want $x_0 = X$ and $x_T = 0$. Under the assumptions that asset prices follow an arithmetic Brownian motion, permanent impact is immediate and linear in total shares executed, and temporary impact is linear in the rate of trading, the solution of the Almgren and Chriss model is

$$x_t = X \frac{\sinh(\kappa(T-t))}{\sinh(\kappa T)} \tag{2}$$

where $\kappa = \sqrt{\frac{\lambda\sigma^2}{\eta}}$. Here, σ and η represent stock volatility and linear temporary market impact cost.

Note that the solution is effectively a decaying exponential $X\exp(-\kappa t)$ adjusted such that $x_T = 0$. It does not depend on the permanent market impact, consistent with the discussion in the previous section. The urgency of trading is embodied in κ. This parameter determines the speed of liquidation independent of the order size X. For a higher risk aversion parameter or volatility—for example, representing increased perceived risk—the speed of trading increases as well. We also see that for a higher expected temporary market impact cost, the speed of trading decreases.

Impact Models

An impact model is used to predict changes in price due to trading activity. This expectation of price change may be used to inform execution and portfolio construction decisions. Several well-known models have been proposed. For example, see Hasbrouck [8], Lillo *et al.* [12], and Almgren *et al.* [3].

Almgren *et al.* use a proprietary data set obtained from Citigroup's equity trading desk in which a trade's direction (buyer or seller initiated) is known. Note that for most public data sets, trade direction is not available and has to be estimated by a classification algorithm. Classification errors in algorithms such as Lee and Ready [11], and Ellis *et al.* [5] introduce a bias that produces an overestimate of the true trading cost.

In Almgren *et al.*, trades serve as a proxy for trading imbalance. The authors assume that, some time after the complete execution of a parent order, only permanent impact remains. This allows them to separate impact into its temporary and permanent components.

The model parameters can then be calculated from a regression, giving the following results. First, permanent impact cost is linear in trade size and volatility. Second, temporary impact cost is linear in volatility and roughly proportional to the square root—Almgren *et al.* find a power 3/5—of the fraction of volume represented by one's own trading during the period of execution. Hence, for a given rate of trading, a less volatile stock with large average daily volume has the lowest temporary impact costs.

The Buy-side Perspective

Optimal execution algorithms have less value to a typical portfolio manager if analyzed separately from the corresponding returns earned by his/her trading strategy. In fact, high transaction costs are not bad *per se*—they could simply prove to be necessary for generating superior returns. At present, the typical sell-side perspective of algorithmic trading does not take expectation of profits or the client's portfolio objectives into account. Needless to say, this is an important component of execution.

The decisions of the trader and the portfolio manager are based on different objectives. The trader decides on the timing of the execution, breaking large parent orders into a series of child orders that, when executed over time, represent the correct trade-off between opportunity cost, market impact, and risk. The trader sees only the trading assets, whereas the portfolio manager sees the entire portfolio, which includes both, the trading assets and the static—nontrading—positions.

The portfolio manager's task is to construct a portfolio by optimizing the trade-off between the opportunity cost, market impact, and risk for the full set of trading and nontrading assets. In general, the optimal execution framework described by Almgren and Chriss is not appropriate for the portfolio manager.

Engle and Ferstenberg [6] proposed a framework that unites these objectives by combining optimal execution and classical mean–variance optimization models. In their model, trading takes place at discrete time intervals as the portfolio manager rebalances his/her portfolio holdings w_t at times $t = 0, 1, \ldots, T$ subject to changing expected returns, μ_t, and risk (as measured by the covariance matrix of returns), Ω_t, until he/she reaches the portfolio that reflects his/her final view $w_T = \frac{1}{2\lambda}\Omega_T^{-1}\mu_T$. The joint dynamic optimization problem has the form

$$\begin{aligned}\max_{\{w_t\}} \Bigg\{ & \sum_{t=1}^{T} \left(w_T^T \mu_T - \lambda w_T^T \Omega_T w_T \right) \\ & - \sum_{t=1}^{T} \left\{ \Delta w_t^T \tau_t + (w_T - w_{t-1})\mu_t \right. \\ & \left. + \lambda (w_T - w_{t-1}) \Omega_t (w_T - w_{t-1}) \right\} \\ & + 2\lambda \sum_{t=1}^{T} (w_T - w_{t-1}) \Omega_t w_T \Bigg\} \end{aligned} \qquad (3)$$

where $\tau_t = \tau_t(\Delta w_t)$ is the temporary market impact function (for simplicity of exposition, we ignore permanent impacts). This is a dynamic programming problem that has to be solved by numerical techniques.

Each one of the three terms in the objective function above has an intuitive interpretation. The first term represents the standard mean–variance optimization problem. The second term corresponds to the optimal execution problem. The third term is the covariance between the remaining shares to be traded and the final position. In the single asset case, the third term is positive (negative) for buying (selling) orders, which implies that risk is reduced (increased). If this term is ignored, which occurs when portfolio allocation and optimal execution are performed separately, then the measurement of total risk is biased.

Popular Algorithmic Trading Strategies

A small number of execution strategies have become *de facto* standards and are offered by most technology providers, banks, and institutional broker/dealers. However, even among these standards, the large number of input parameters makes it difficult to compare execution strategies directly.

Typically, a strategy is motivated by a *theme* or *style* of trading. The objective is to minimize either absolute or risk-adjusted costs relative to a *benchmark*. For strategies with mathematically defined objectives, an *optimization* is performed to determine how to best use the strategy to maximize a trader's or portfolio manager's utility. A *trade schedule*—or *trajectory*—is planned for strategies with a target quantity of shares to execute. The *order placement* engine—sometimes called the *microtrader*—translates from a strategy's broad objectives to individual orders. User-defined *input parameters* control the trade schedule and order placement strategy.

In this section, we review some of the most common algorithmic trading strategies.

Volume-weighted Average Price

Six or seven years ago, the VWAP execution strategy represented the bulk of algorithmic trading activity. Currently, it is second in popularity only to arrival price. The appeal of benchmarking to VWAP is that the benchmark is easy to compute and intuitively accessible.

The typical parameters of a VWAP execution are the start time, the end time, and the number of shares to execute. Additionally, optimized forms of this strategy require a choice of risk aversion.

The most basic form of VWAP trading uses a model of the fractional daily volume pattern over the execution period. A trade schedule is calculated to match this volume pattern. For example, if the execution period is one day, and 20% of a day's volume is expected to be transacted in the first hour, a trader using this basic strategy would trade 20% of his/her target accumulation or liquidation in the first hour of the day. Since the daily volume pattern has a U shape—with more trading in the morning and afternoon and less in the middle of the day—the volume distribution of shares executed in a VWAP pattern would also have this U shape.

VWAP is an ideal strategy for a trader who meets all of the following criteria:

- his/her trading has little or no alpha during the execution period;
- he/she is benchmarked against the VWAP;
- he/she believes that market impact is minimized when his/her own rate of trading represents the smallest possible fraction of all trading activity; and
- he/she has a set number of shares to buy or sell.

Deviation from these criteria may make VWAP strategies less attractive. For example, market participants who trade over the course of a day and have strong positive alpha may prefer a front-weighted trajectory, such as those that are produced by an arrival price strategy.

The period of a VWAP execution is most typically a day or a large fraction of a day. Basic VWAP models predict the daily volume pattern using a simple historical average of fractional volume. Several weeks to several months of data are commonly used. However, this forecast is noisy. On any given day, the actual volume pattern deviates substantially from its historical average, complicating the strategy's objective of minimizing its risk-adjusted cost relative to the VWAP benchmark. Some models of fractional volume attempt to increase the accuracy of volume pattern prediction by making dynamic adjustments to the prediction based on observed trading results throughout the day.

Several variations of the basic VWAP strategy are common. The ideal VWAP user (as defined earlier) can lower his/her expected costs by increasing his/her exposure to risk relative to the VWAP benchmark. For example, assuming an alpha of zero, placing limit orders throughout the execution period and catching up to a target quantity with a market order at the end of the execution period will lower the expected cost while increasing risk. This is the highest risk strategy. Continuously placing small market orders in the fractional volume pattern is the lowest risk strategy, but has a higher expected cost. For a particular choice of risk aversion, somewhere between the highest and lowest risk strategies, is a compromise optimal strategy that perfectly balances risk and costs.

For example, a risk-averse VWAP strategy might place one market order of 100 shares every 20 s, whereas a less risk-averse strategy might place a limit order of 200 shares, and, 40 s later, place a market order for the difference between the desired fill of 200 and the actual fill (which may have been smaller). The choice of the average time between market orders in a VWAP execution implies a particular risk aversion.

For market participants with a positive alpha, a frequently used rule-of-thumb optimization is compressing trading into a shorter execution period. For example, a market participant may try to capture more profits by doing all of his/her VWAP trading in the first half of the day instead of taking the entire day to execute.

In another variant of VWAP—*guaranteed VWAP*—a broker commits capital to guarantee his/her client the VWAP price in return for a predetermined fee. The broker takes on a risk that the difference between his/her execution and VWAP will be greater than the fee he/she collects. If institutional trading volume and individual stock returns were uncorrelated, the risk of guaranteed VWAP trading could be diversified away across many clients and many stocks. In practice, managing a guaranteed VWAP book requires some complex risk calculations that include modeling the correlations of institutional trading volume.

Time-weighted Average Price

The time-weighted average price (TWAP) execution strategy attempts to minimize market impact costs by maintaining an approximately constant rate of trading over the execution period. With only a few parameters—start time, end time, and target quantity—TWAP has the advantage of being the simplest execution strategy to implement. As with VWAP, optimized forms of TWAP may require a choice of risk aversion. Typically, the VWAP or arrival price benchmarks are used to gauge the quality of a TWAP execution. TWAP is hardly ever used as its own benchmark.

The most basic form of TWAP breaks a parent order into small child orders and executes these child orders at a constant rate. For example, a parent order of 300 shares with an execution period of 10 min could be divided into three child orders of 100 shares each. The child orders would be executed at the 3:20, 6:40, and 10:00 min marks. Between market orders, the strategy may place limit orders in an attempt to improve execution quality.

An ideal TWAP user has almost the same characteristics as an ideal VWAP user, except that he/she believes that the lowest *trading rate*—not the lowest *participation rate*—incurs the lowest market impact costs.

TWAP users can benefit from the same type of optimization as VWAP users by placing market orders less frequently, and using resting limit orders to attempt to improve execution quality.

Participation

The participation strategy attempts to maintain a constant fractional trading rate. That is, its own trading rate as a fraction of the market's total trading rate should be constant throughout the execution period. If the fractional trading rate is maintained exactly, participation strategies cannot guarantee a target fill quantity.

The parameters of a participation strategy are the start time, end time, fraction of market volume the strategy should represent, and max number of shares to execute. If the max number of shares is specified, the strategy may complete execution before the end time. Along with VWAP and TWAP, participation is a popular form of nonoptimized strategies, though some improvements are possible with optimization.

VWAP and arrival price benchmarks are often used to gauge the quality of a participation strategy execution. The VWAP benchmark is particularly appropriate because the volume pattern of a perfectly executed participation strategy is the market's volume pattern during the period of execution. An ideal user of participation strategies has all of the same characteristics as an ideal user of VWAP strategies, except that he/she is willing to forego certain execution to maintain the lowest possible fractional participation rate.

Participation strategies do not use a trade schedule. The strategy's objective is to participate in volume as it arises. Without a trade schedule, a participation strategy cannot guarantee a target fill quantity. The most basic form of participation strategies waits for trading volume to show up on the tape, and follows this volume with market orders. For example, if the target fractional participation rate is 10%, and an execution of 10 000 shares is shown to have been transacted by other market participants, a participation strategy would execute 1000 shares in response.

Unlike a VWAP trading strategy, which for a given execution may experience large deviations from an execution period's actual volume pattern, participation strategies can closely track the actual—as opposed to the predicted—volume pattern. However, close tracking has a price. In the earlier example, placing a market order of 1000 shares has a larger expected market impact than slowly following the market's trading volume with smaller orders. An optimized form of the participation strategy amortizes the trading shortfall over some period of time. Specifically, if an execution of 10 000 shares is shown to have been transacted by other market participants, instead of placing 1000 shares all at once, a 10% participation strategy might place 100 share orders over some period of time to amortize the shortfall of 1000 shares. The result is a lower expected shortfall, but a higher dispersion of shortfalls.

Market-on-close

The market-on-close strategy is popular with market participants who either want to minimize risk-adjusted costs relative to the closing price of the day or want to manipulate—*game*—the close to create the perception of a good execution. The ideal market-on-close user is benchmarked to the close of the day and has low or negative alpha. The parameters of a market-on-close execution are the start time, the end time, and the number of shares to execute. Optimized forms of this strategy require a risk-aversion parameter.

When market-on-close is used as an optimized strategy, it is similar in its formulation to an arrival price strategy. However, with market-on-close, a back-weighted trade schedule incurs less risk than a front-weighted one. With arrival price, an infinitely risk averse trader would execute everything in the opening seconds of the execution period. With market-on-close, an infinitely risk averse trader would execute everything at the closing seconds of the day. For typical levels of risk aversion, some trading would take place throughout the execution period. As with arrival price optimization, positive alpha increases urgency to trade and negative alpha encourages delayed execution.

In the past, market-on-close strategies were used to manipulate—or *game*—the close, but this has become less popular as the use of VWAP and arrival price benchmarks have increased. Gaming the close is achieved by executing rapidly near the close of the day. The trade print becomes the closing price or very close to it, and hence shows little or no shortfall from the closing price benchmark. The true cost of the execution is hidden until the next day when temporary impact dissipates and prices return to a new equilibrium.

Arrival Price

The arrival price strategy—also called the *implementation shortfall* strategy—attempts to minimize risk-adjusted costs using the arrival price benchmark. Arrival price optimization is the most sophisticated and popular of the commonly used algorithmic trading strategies.

The ideal user of arrival price strategies has the following characteristics:

- he/she is benchmarked to the arrival price;
- he/she is risk averse and knows his/her risk aversion parameter;
- he/she has high positive or high negative alpha; and
- he/she believes that market impact is minimized by maintaining a constant rate of trading over the maximum execution period while keeping trade size small.

Most implementations are based on some form of the risk-adjusted cost minimization introduced by Almgren and Chriss [1]. In the most general terms, an arrival price strategy evaluates a series of trade schedules to determine which one minimizes risk-adjusted costs relative to the arrival price benchmark. As discussed in the section on optimal execution, under certain assumptions, this problem has a closed form solution.

The parameters in an arrival price optimization are alpha, number of shares to execute, start time, end time, and a risk aversion parameter. For buyers (sellers), positive (negative) alpha encourages faster trading. For both buyers and sellers, risk encourages faster trading, while market impact costs encourage slower trading.

For traders with positive alpha, the feasible region of trade schedules lies between the immediate execution of total target quantity and a constant rate of trading throughout the execution period.

A more general form of arrival price optimization allows for both the buyers and sellers to have either positive or negative alpha. For example, under the assumption of negative alpha, shares held long and scheduled for liquidation are—without considering one's own trading—expected to go up in price over the execution period. This would encourage a trader to delay execution or stretch out trading. Hence, the feasible region of solutions that account for both positive and negative alpha includes back-weighted as well as front-weighted trade schedules.

Other factors that necessitate back-weighted trade schedules in an arrival price optimization are expected changes in liquidity and expected crossing opportunities. For example, an expectation of a cross later in the execution period may provide enough cost savings to warrant taking on some price risk and the possibility of a compressed execution if the cross fails to materialize. Similarly, if market impact costs are expected to be lower later in the execution period, a rational trader may take on some risk to obtain this cost savings.

A variant of the basic arrival price strategy is *adaptive arrival price*. A favorable execution may result in a windfall in which an accumulation of a large number of shares takes place at a price significantly below the arrival price. This can happen by random chance alone. Almgren and Lorenz [2] demonstrated that a risk-averse trader should use some of this windfall to reduce the risk of the remaining shares. He/she does this by trading faster and thus incurring a higher market impact. Hence, the strategy is adaptive in that it changes its behavior based on how well it is performing.

Crossing

Though crossing networks have been around for some time, their use in algorithmic trading strategies is a relatively recent development. The idea behind crossing networks is that large limit orders—the kind of orders that may be placed by large institutional traders—are not adequately protected in a public exchange. Simply displaying large limit orders in the open book of an electronic exchange may leak too much information about institutional traders' intentions. This information is used by prospective counterparties to trade more passively in the expectation that time constraints will force traders to replace some or all of the large limit orders with market orders. In other words, information leakage encourages *gaming* of large limit orders. Crossing networks are designed to limit information leakage by making their limit books opaque to both their clients and the general public.

A popular form of cross is the *midquote cross*, in which two counterparties obtain a midquote fill price. The midquote is obtained from a reference exchange, such as the NYSE or other public exchange. Regulations require that the trade is then printed to a public exchange to alert other market participants that it has taken place. The cross has no market impact but both the counterparties pay a fee to the crossing network. These fees are typically higher than the fees for other types of algorithmic trading because the market impact savings are significant while the fee is contingent on a successful cross.

More recently, crossing networks have offered their clients the ability to place limit orders in the crossing networks' dark books. Placing a limit order in a crossing network allows a cross to occur only at a certain price. This makes crossing networks much more like traditional exchanges, with the important difference that their books are opaque to market participants.

To protect their clients from price manipulation, crossing networks implement antigaming logic. As previously explained, opaqueness is itself a form of antigaming, but there are other strategies. For example, some crossing networks require orders to be above a minimum size or to remain in the network longer than a prespecified minimum time. Other networks will cross only orders of similar size. This prevents traders from pinging—sending small orders to the network to determine which side of the network's book has an order imbalance.

Another approach to antigaming prevents crosses from taking place during periods of unusual market activity. The assumption is that some of this unusual activity is caused by traders trying to manipulate the spread in the open markets to get a better fill in a crossing network.

Some networks also attempt to limit participation by active traders, monitoring their clients' activities to see if their behavior is more consistent with normal trading than with gaming.

There are several different kinds of crossing networks. A *continuous crossing network* constantly sweeps through its book in an attempt to match buy orders with sell orders. A *discrete crossing network* specifies points in time when a cross will take place, say every half hour. This allows market participants to queue up in the crossing network just prior to a cross instead of committing resting orders to the network for extended periods of time. Some crossing networks allow scraping—a one time sweep to see if a single order can find a counterparty in the crossing network's book—while others allow only resting orders.

In *automated* crossing networks, resting orders are matched according to a set of rules, without direct interaction between the counterparties. In *negotiated* crossing networks, the counterparties first exchange indications of interest, then negotiate price and size *via* tools provided by the system.

Some traditional exchanges now allow the use of *invisible orders*, resting orders that sit in their order books but are not visible to market participants. These orders are also referred to as *dark liquidity*. The difference between these orders and those placed in a crossing network is that traditional exchanges offer no special antigaming protection.

Private dark pools are collections of orders that are not directly available to the public. For example, a bank or pension manager might have enough order flow to maintain an internal order book that, under special circumstances, is exposed to external scraping by a crossing network or *crossing aggregator*.

A *crossing aggregator* charges a fee for managing a single large order across multiple crossing networks. Order placement and antigaming rules differ across networks, making this task fairly complex. A crossing aggregator may also use information about historical and real-time fills to direct orders. For example, failure to fill a small resting buy order in a crossing network may betray information of a much larger imbalance in the network's book. This makes the network a more attractive destination for future sell orders. In general, the management of information across crossing networks should give crossing aggregators higher fill rates than exposure to any individual network.

Crossing lends itself to several optimization strategies. Longer exposure to a crossing network not only increases the chances of an impact-free fill, but also increases the risk of a large and compressed execution if an order fails to obtain a fill. Finding an optimal exposure time is one type of crossing optimization. A more sophisticated version of this approach is solving for a *trade-out*, a schedule for trading shares out of the crossing network into the open markets. As time passes and a cross is not obtained, the strategy mitigates the risk of a large, compressed execution by slowly trading parts of the order into the open markets.

Other Algorithms

Two other algorithms are typically included in the mix of standard algorithmic trading offerings. The first is *liquidity seeking* where the objective is to soak up available liquidity. As the order book is depleted, trading slows down. As the order book is replenished, trading speeds up.

The second algorithm is *financed trading*. The idea behind this strategy is to use a sale to finance the purchase of a buy with the objective of obtaining some form of hedge. This problem has all of the components of a full optimization. For example, if, after a sell, a buy is executed too quickly, it will obtain a less favorable fill price. On the other hand, executing a buy *leg* too slowly increases the tracking error between the two components of the hedge and increases the dispersion of costs required to complete the hedge.

What is Next?

The average trade size for IBM, as reported in the Trade and Quote (TAQ) database, declined from 650 shares in 2004 to 240 shares in 2007. Falling trade sizes are evidence of the impact of algorithmic trading. Large, infrequent portfolio rebalancing and trading are being replaced by *small delta continuous trading*.

The antithesis of the small delta continuous trading approach is embodied in the idea of *lazy portfolios*, in which portfolios are rebalanced infrequently to reduce market impact costs. The first argument against lazy portfolios is that as time passes, the weights drift further away from optimal target holdings, in both the alpha and risk dimensions. Second, the use of an optimizer after long holding periods tends to produce large deviations from current holdings. When executed—often relatively quickly—these deviations result in significant market impact costs.

Engle and Ferstenberg [6] show that to correctly measure risk, we must take both the existing positions and unexecuted shares into account. This idea unites execution risk with portfolio risk. Portfolio construction and optimal execution are similarly united by incorporating market impact costs directly into the portfolio construction process.

Ideally, the portfolio manager would like to solve a problem similar in nature to the multiperiod consumption-investment problem [13], that, in addition, takes market impact costs and changing probability distributions for a large universe of securities into account. This *dynamic portfolio* or *small delta continuous trading* problem represents the next step in the evolution of institutional money management. However, it presents some mathematical and computational challenges. As has been pointed out by Sneddon [14], dynamic portfolio problem differs in several important ways from the classical multiperiod consumption-investment problem. First, the return probability distributions change throughout time. Second, the objective functions for active portfolio management do not depend on the predicted alpha/risk, but rather on realized return/risk. Finally, the dynamics of the model may be far more complex. Grinold [7] provides an elegant and analytically tractable but greatly simplified model. Kolm and Maclin [10] describe a full-scale simulation-based framework that incorporates realistic constraints and a transaction cost model.

Other efforts of ongoing research in algorithmic trading are extending market microstructure and optimal execution models to futures, options, and fixed income products. These initiatives follow the dominant theme of algorithmic trading, the creation of a unified view, an all-encompassing framework for the entire trading process, including modeling, portfolio construction, risk analytics, and execution across all tradeable asset classes.

References

[1] Almgren, R. & Chriss, N. (2000). Optimal execution of portfolio transactions, *Journal of Risk* **3**(2), 5–39.

[2] Almgren, R. & Lorenz, J. (2007). Adaptive arrival price, in *Algorithmic Trading III: Precision, Control, Execution*, B.R. Bruce, ed, Institutional Investor Journals.

[3] Almgren, R., Thum, C., Hauptmann, E. & Li, H. (2005). Equity market impact, *Risk* **18**(7), 57–62.

[4] Bertsimas, D. & Lo, A.W. (1998). Optimal control of liquidation costs, *Journal of Financial Markets* **1**, 1–50.

[5] Ellis, K., Michaely, R. & O'Hara, M. (2000). The accuracy of trade classification rules: Evidence from Nasdaq, *Journal of Financial and Quantitative Analysis* **35**(4), 529–551.

[6] Engle, R.F. & Ferstenberg, R. (2007). Execution risk, *Journal Portfolio Management* **33**, 34–44.

[7] Grinold, R.C. (2007). Dynamic portfolio analysis, *Journal of Portfolio Management* **34**(1), 16–26.

[8] Hasbrouck, J. (1991). Measuring the information content of stock trades, *Journal of Finance* **46**(1), 179–207.

[9] Huberman, G. & Stanzl, W. (2004). Price manipulation and quasi-arbitrage, *Econometrica* **72**(4), 1247–1275.

[10] Kolm, P. & Maclin, L. (2009). *Small Delta Continuous Trading for Dynamic Portfolios*, Courant Institute, New York University.

[11] Lee, C. & Ready, M. (1991). Inferring trade direction from intraday data, *Journal of Finance* **46**, 733–746.

[12] Lillo, F., Farmer, J.D. & Mantegna, R.N. (2003). Master curve for price-impact function, *Nature* **421**, 129–130.

[13] Merton, R. (1969). Lifetime portfolio selection under uncertainty: The continuous-time case, *Review of Economics and Statistics* **51**, 241–257.

[14] Sneddon, L. (2005). *The Dynamics of Active Portfolios*, Westpeak Global Advisors.

PETTER N. KOLM & LEE MACLIN