Stanford University MS&E 444 Investment Practice Course Project

Order Book Simulator and Optimal Liquidation Strategies

Su Chen*, Chen Hu†, Yijia Zhou‡

June 2010

Abstract

Our project analyzes the statistics of an order-driven market for single stocks, builds an exchange system to process orders and maintains four order books for different types of orders. We explore the self/cross exciting attributes of order arrivals by computing the cross correlation coefficients. Then we apply four-variate Hawkes process to simulate arrivals of orders, and measure the market impact caused by different sizes of orders. The bigger the liquidation size, the larger the market impact is. Such a market impact has a long memory. This finding is useful for designing optimal liquidation strategies. Setting the market volume weighted average price (VWAP) as the benchmark, we construct three liquidation strategies: naïve split, valley, and relative volume with the aim to complete a full execution of liquidation within a certain time horizon and meanwhile to outperform the VWAP benchmark as much as possible. All of the three strategies can guarantee the full execution of liquidation trades within a certain time horizon. The relative volume strategy has the smallest market impact, and outperforms the VWAP benchmark.

Keywords: order-driven algorithmic trading, optimal liquidation strategies, multivariate Hawkes process

Structure

---

* Statistics, suchen@stanford.edu
† Statistics, Financial Mathematics, chenhu@stanford.edu
‡ Financial Mathematics, zhouyj@stanford.edu

1. Introduction

Algorithmic Trading is widely used by pension funds, mutual funds, and other buy side institutional traders, to divide large trades into several smaller trades and to decide the right market condition to place those orders, so that traders can better control market impact caused by their trades, and thus better control the price and liquidity risk. Sell side traders, such as market makers and some hedge funds, provide liquidity to the market, generating and executing orders automatically.

Algorithmic trading can be based on a variety of different strategies, but they all use market data as an input. Algorithmic trading has become increasingly common recently, and now accounts more than 60% of total volume[1] in US equities by 2010.

Prices of stocks are heavily affected by market microstructure factors. The competitive market paradigm assumes that the security markets are perfectly elastic and all market orders can be executed instantaneously. Elastic markets mean there is no "quantity effect" on the security price. However, in real market situations, when the market participants detect an unusual high volume of sell order, they would naturally adjust their bids to a lower level. As a result, there is a price discrepancy between the price of security at the time when a large trade order is placed and at the time when the trade order is executed. The difference in these prices is called the liquidation discount. Also, the transaction may take a considerable amount of time to complete so that there are execution time lags in selling a large amount of shares. Assuming that the execution time lags and liquidation discount are deterministic, the optimal liquidation problem is to develop an optimal execution strategy such that a trader can unwind a portfolio position within a fixed time constraint subject to the optimization of certain criteria, like the minimization of the expected shortfall in value.

The objective of order book study is to examine how these market microstructure factors can shape the market price and volume. Results of order book study are especially interesting to algorithmic trading in order-driven markets. Algorithmic trading in order-driven markets uses computer programs to identify market opportunities and decide on aspects of the order such as the timing, price, or quantity of the order. Using the intraday order book data, it is interesting to study the market impact of order intensities, and it is also interesting to study how to design execution strategies.

The market volume weighted average price (VWAP) is regarded as a good benchmark to assess the efficiency of order execution[2]. In the case of liquidation, a good liquidation strategy should achieve a price higher than the market VWAP.

A lot of scholars and financial professionals did research about optimal liquidation strategies. Bertsimas and Lo [3] added the price impact into the cost function for buying stocks, designed an execution strategy with the aim of minimizing expected cost of buying stocks within a fixed finite time horizon. Almgren and Chriss [4] split the total price impact into temporary imbalances in supply and demand caused by a trade order, and the price drop that persists for the whole life of the liquidation period. Almgren [5] designed optimal trading strategies with a power law function to minimize the volatility risk and market impact costs. Subramanian and Jarrow[6] incorporated both the execution lags and liquidation discounts in their liquidation model, and designed optimal execution scheme to maximize the sum of expected utility value gained for each sell order. Longstaff [7] studied optimal self-financing rebalance between risk-free assets and risky assets to maximize the expected utility value of terminal wealth. Duffie and Ziegler [8] designed a strategy to minimize expected transaction cost by selling liquid assets first, and another strategy to sell illiquid asset and keep a cushion of liquid assets, at the expense of increasing the expected transaction costs. Kwok [9] consider a portfolio of stock and cash with the objective to unwind his position on the risky asset so that the expected value of cash at the end of a fixed time horizon is maximized.

Our project adopted a simulation-based framework to analyze the intensities of different orders, and then design efficient liquidation strategies to minimize the market impact of liquidation trades. We built a single-stock exchange to process different types of orders and rank the open orders in order books. Such a system allows us to not only evaluate the price impacts of trading but also test and improve our trading strategies. For simplicity, we assume there are only five types of orders: market buy, market sell, limit buy, limit sell, and cancellation. And thus we can construct four order books for market buy, market sell, limit buy, and limit sell orders respectively.

In Section 2, we discuss the single-stock exchange system. In Section 3, we present a few statistical observations and stylized facts about data. In Section 4, we capture the cross/self-exciting features of order arrivals, estimate the shape of order price and order size, and propose a simulator based with Hawkes process. We also analyze the market impact of liquidation trades. In Section 5, we explore optimal liquidation strategies, and compare our results with the prevalent benchmark. Section 6 we conclude our report and suggest future work.

2. Single-Stock Exchange System

To model the single stock orders and trading, we assume there are five types of orders on the market. They are market buy, market sell, limit buy, limit sell, and cancellation orders. Cancellation orders are not real orders, but signals to cancel prior orders. For simplicity, we call those cancellation signals as cancellation orders. A market order is a buy or sell order in which the broker is to execute the order at the best price currently available. The benefits of placing a market order is that you can almost guarantee the execution for sure, while the shortcoming of placing a market order is that you might end up paying a much higher price for a stock than they expected to, or receiving a much lower price for a stock than they expected to. To offset the shortcoming of market orders, you can execute with limit orders. A limit order is an order to a broker to buy a specified quantity of a security at or below a specified price, or to sell it at or above a specified price (called the limit price).

2.1 System Structure

In this part we will build an exchange for a single stock. The mechanism of this exchange is given as follows:

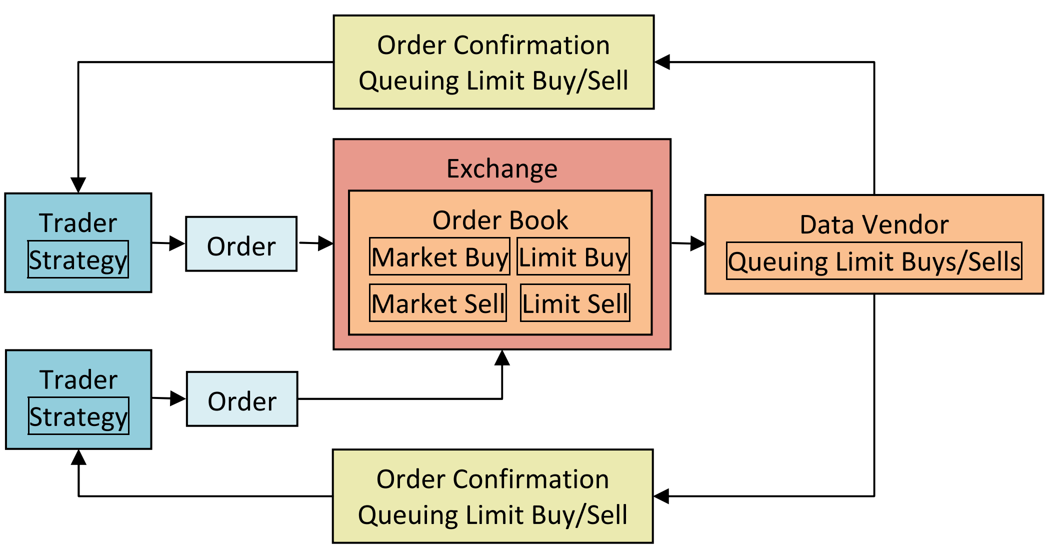

Figure 1. Exchange System Mechanism

2.2 Inputs and Outputs

The system's inputs are: current time, current incoming order, current market price, sorted limit sell orders, sorted limit buy orders, sorted market sell orders, and sorted market buy Orders.

Outputs are: current time, updated market price, market volume, order confirmation, sorted limit buy orders, sorted market sell orders, sorted market buy Orders, and order confirmation, where order confirmation will identify current order ID, counterparty order(s) ID, current executed volume (market volume), execution price (not necessary to equal market price, self indicator for current order, self indicator for counterparty order(s).

2.3 Order Processing Rules

Regarding the inner algorithm for a stock exchange, people tend to first rank the orders, and then process the best buy order (which offers the highest price) with the best sell order (which offers the lowest sell price). However, if we further consider the order matching and queuing dynamics, we will know that we can switch the two steps to reduce the space requirement and computational complexity of our exchange system.

To understand this logic, let's take a limit buy order as an example. If after ranking it is not the limit buy order with the highest bid price that can be matched with sell orders, then there must be another limit buy order with a higher bid price that can be matched with sell orders, or there must be market buy orders waiting to be matched. If so, the limit buy order with a higher bid price or the market buy order should have already been matched before the new limit buy order coming in. So we have proved that we can first match the incoming orders with orders in reverse direction, and then rank the rest of the incoming order in the relevant order book.

Order Type	Processing Algorithm
Market Buy	Step 1: match with orders in the market sell order book; Step 2: match with orders in the limit sell order book; Step 3: the rest is written into the order book for market buy orders, and is to be matched with market/limit sell orders in the future, or to be canceled by cancellation order.
Market Sell	Step 1: match with orders in the market buy order book; Step 2: match with orders in the limit sell order book; Step 3: the rest is written into the market sell order book, and is to be matched with market/limit buy orders in the future, or to be canceled by cancellation order.
Limit Buy	Step 1: match with orders in the market sell order book; Step 2: match with orders in the limit sell order book if the limit buy price is above the lowest limit sell price; Step 3: the rest is written into the order book for limit buy orders, and is to be matched with market/limit sell orders in the future, or to be canceled by cancellation order.
Limit Sell	Step 1: match with orders in the market buy order book; Step 2: match with orders in the limit buy order book if the limit sell price is below the highest limit buy price; Step 3: the rest is written into the order book for limit sell orders, and is to be matched with market/limit buy orders in the future, or to be canceled by cancellation order.
Cancellation	Step 1: Find whether the order the cancellation order is supposed to cancel is still in the order books. Step 2: If so, remove that order. Otherwise, omit this cancellation order.

Table 1. Order Processing Algorithms

2.4 Queuing Rules in Order Books

We maintain four order books for market buy, market sell, limit buy and limit sell orders respectively. And the ranking rules are given as follows:

I. Market buy orders are sorted by ascending arrival time.
II. Market sell orders are sorted by ascending arrival time.
III. Limit buy orders are first sorted by descending price, and then limit buy orders with the same price are sorted by ascending arrival time.
IV. Limit sell orders are first sorted by ascending price, and then limit sell orders with the same price are sorted by ascending arrival time.

3. Descriptive Statistics of Orders

3.1 Raw Data Introduction

We have the Santa Fe Institute's intraday order data for the term from 05/02/2000 0700 GMT to 06/15/2000 1100 GMT. After raw data cleaning, there are 65,535 lines of data that can be used for our project. Each line contains either an order placement or an order cancellation. All events are from the on-book fully electronic market (65% of all orders). Most days run from 0700 to 1700 GMT (market hours are from 0800 to 1630 GMT).

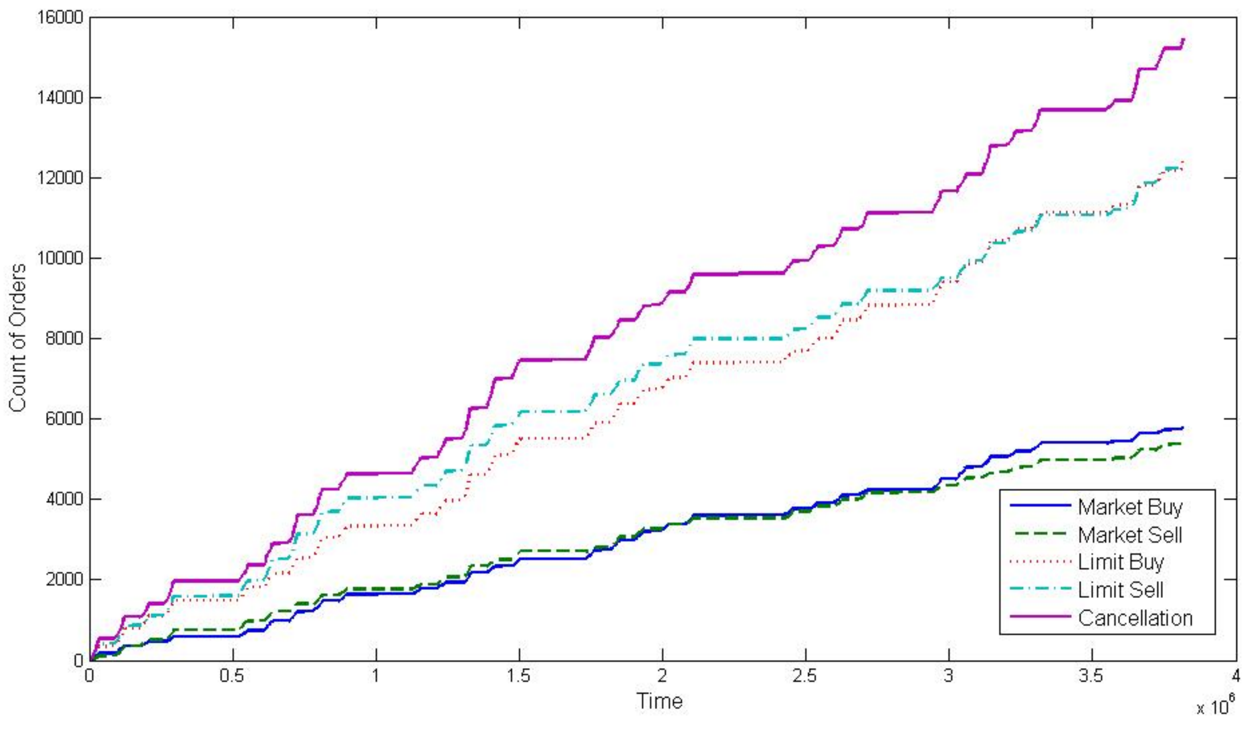

Figure 2. Count of Orders

3.2 Order Price Shape

The second step is to determine at which price level the order will be placed at. Of course, for a market order, it does not carry a price level. Therefore we just need to consider the Limit Buy Order and the Limit Sell Order. Intuitively the order price level should be somewhere near the current Market Mid Price. We make a histogram based on London Exchange Date, which shows up the distribution of (Order Price Level – Market Mid Price). Here is the graph:

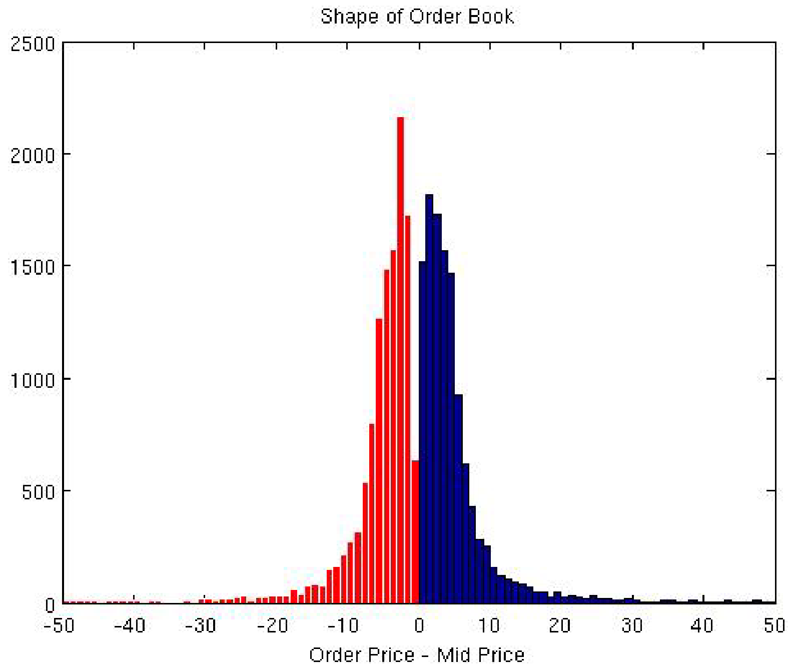

Figure 3. Market Mid Price Less Buy Order Price, and Sell Order Price Less Market Mid Price

The blue bars are the Limit Sell Orders. The red bars are the Limit Buy Orders. We can see that Limit Sell Orders are always higher than the Market Mid Price, while the Limit Buy Orders are always lower than the Market Mid Price. Also they are pretty symmetric about zero. The interesting thing is that the mode is not at 1 tick higher or 1 tick lower than the Market Mid Price. Instead, the mode is somewhere around 3 ticks higher or lower, depends on whether it is a buy order or a sell order. Gamma distributions have such properties, which leads us to fit Gamma Distributions to the data. Here are the results:

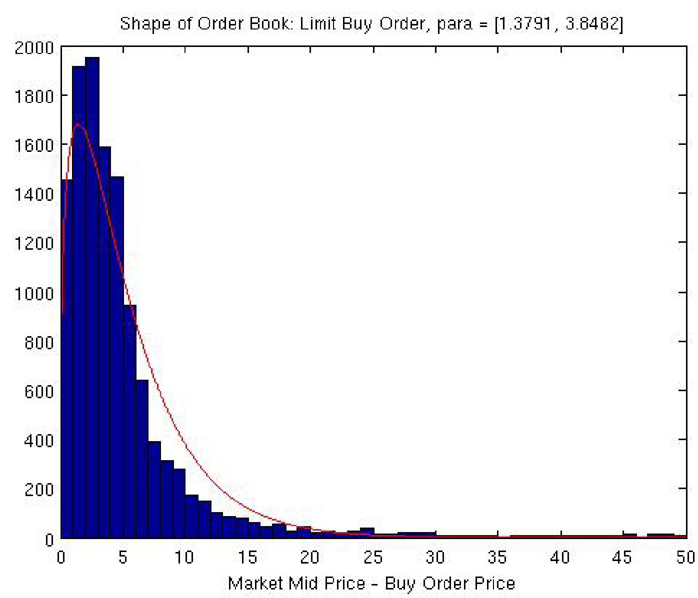

Figure 4. Fitted Shape of Market Mid Price Less Buy Order Price

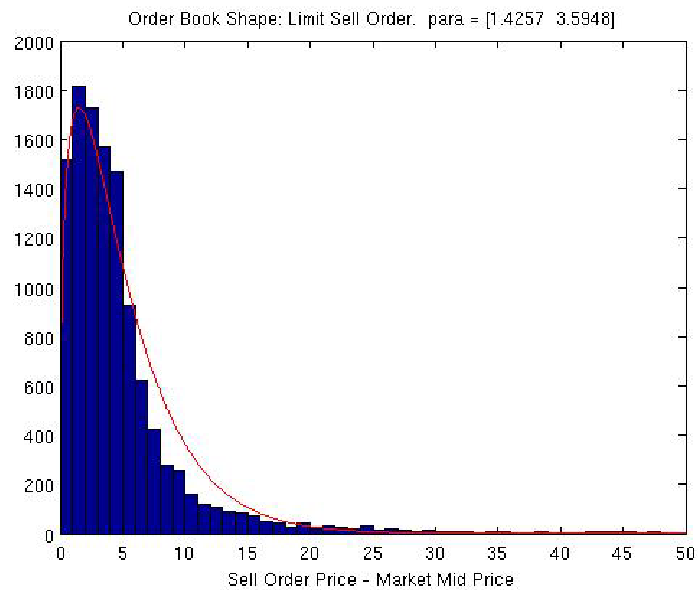

Figure 5. Fitted Shape of Sell Order Price Less Market Mid Price

$\Gamma(\kappa, \theta)$ Distribution	κ	θ
Market Mid Price – Buy Order Price:	1.3791	1.4257
Sell Order Price – Market Mid Price:	3.8402	3.5948

Table 2. Fitted Parameters of Price Shape

3.3 Order Size Distribution

The first step to build up our order simulator is to determine the size of each order. For simplicity, we are assuming the size of the orders are totally independent of its type (Market or Limit, Buy or Sell), also independent of the price level. Based on the Santa Fe Institute data, we have the following histogram:

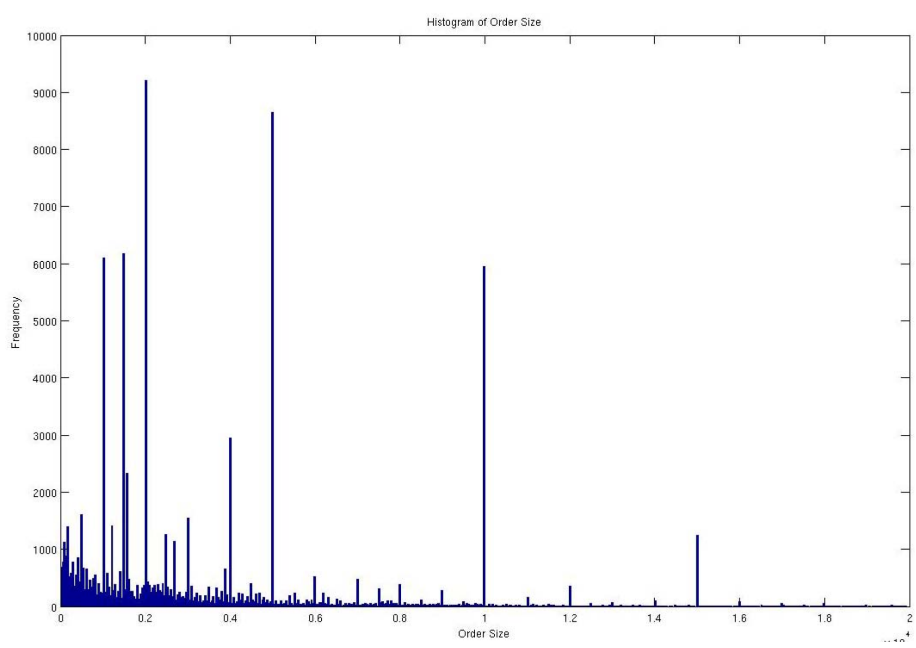

Figure 6. Histogram of Order Size

It's quite clear to see that, most of the orders are actually of size 10000, 5000, 15000, etc. Investor rarely place orders which size is a weird number. So what we do is take out the 12 most frequently placed order size and ignore the rest of them. Therefore we are assuming that our order sizes are drawn independently from a Multinomial Distribution, with parameters in the following table:

Order Size	Percentage	Accumulated Percentage
2000	19.71%	19.71%
5000	19.03%	38.74%
10000	13.07%	51.81%
1000	13.02%	64.83%
1500	12.86%	77.69%
4000	6.39%	84.08%
500	2.93%	87.01%
3000	2.92%	89.92%
15000	2.76%	92.69%
1200	2.69%	95.38%
2500	2.35%	97.72%
2700	2.28%	100.00%

Table 3. Order Size Distribution for Simulation

3.4 Cross Correlation of Order Arrivals

We applied cross correlation to examine the self/cross exciting features of order intensities. In signal processing, cross-correlation is a measure of similarity of two waveforms as a function of a time-lag applied to one of them. Cross correlation is a standard method of estimating the degree to which two series are correlated. Consider two series x(i) and y(i) where i=0,1,2...N-1. The cross correlation r at delay d is defined as

$$r = \frac{\sum_i (x_i - \mu_x)(y_{i-d} - \mu_y)}{\sqrt{\sum_i (x_i - \mu_x)^2 \cdot \sum_i (y_{i-d} - \mu_y)^2}}$$

Here μ_x and μ_y are the means of the corresponding series. If the above is computed for all delays $d = 0, 1, 2, \ldots, N-1$, then it results in a cross correlation series of twice the length as the original series. Setting the max lag as 50, we can get the following pictures for the cross correlation coefficient among different types of orders:

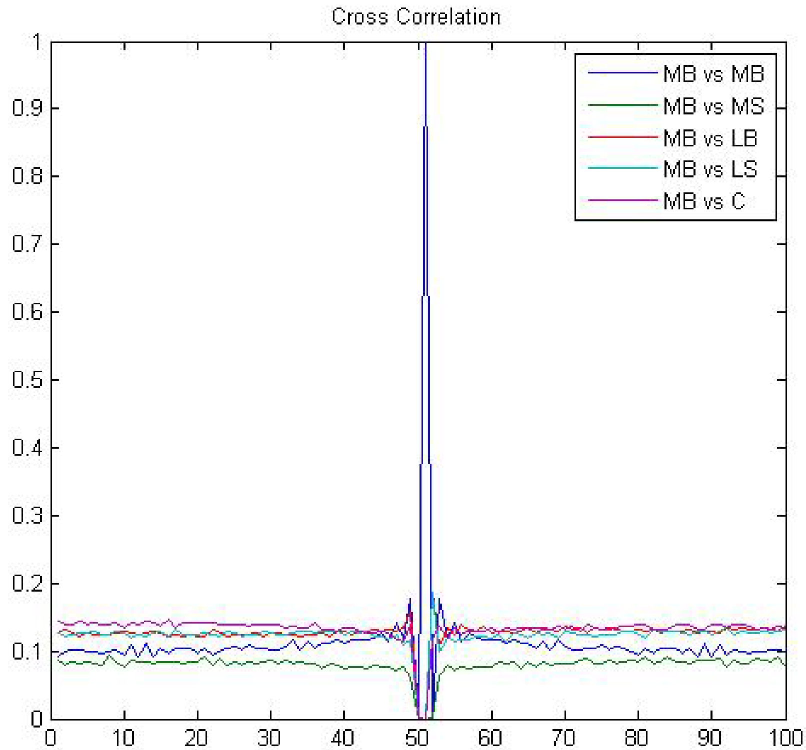

Figure 7. Normalized Cross Correlation between Market Buy Order Arrivals and Other Order Arrivals

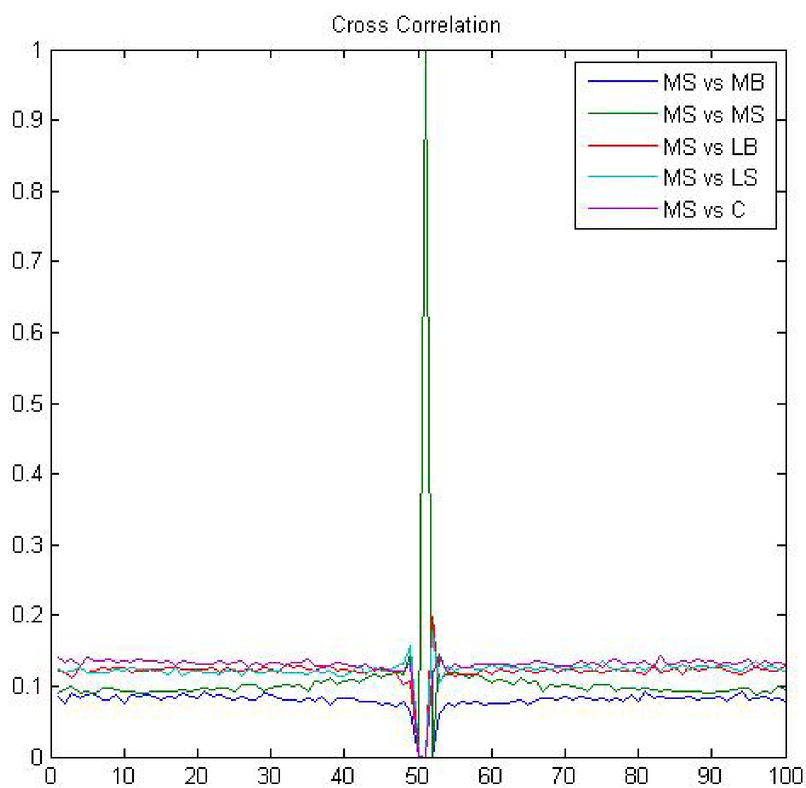

Figure 8. Normalized Cross Correlation between Market Sell Order Arrivals and Other Order Arrivals

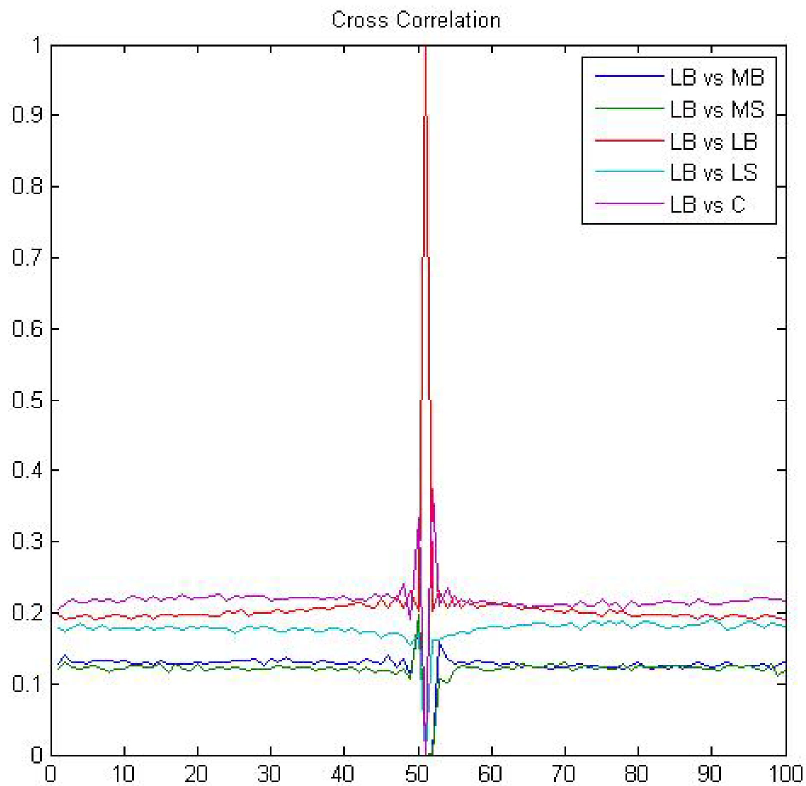

Figure 9. Normalized Cross Correlation between Limit Buy Order Arrivals and Other Order Arrivals

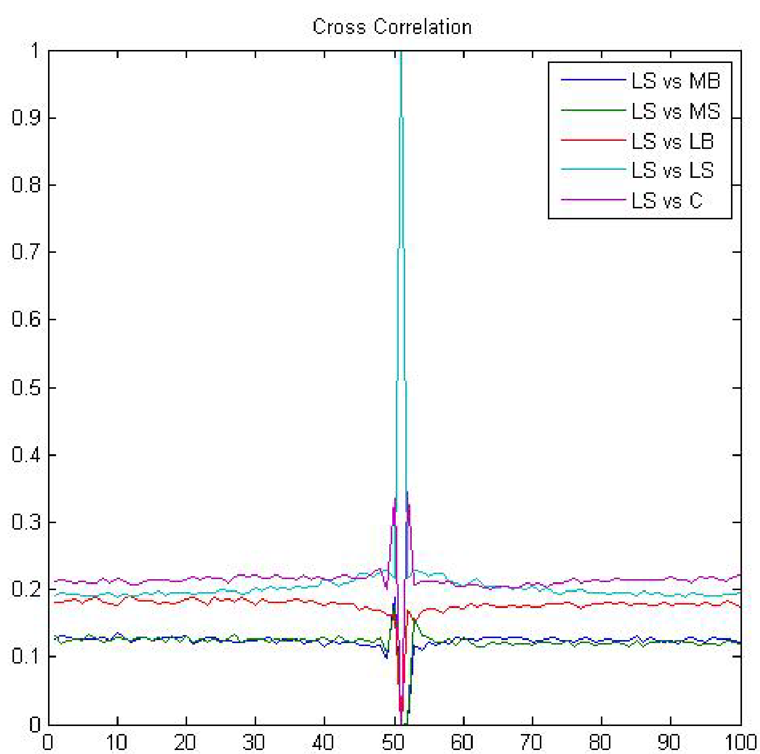

Figure 10. Normalized Cross Correlation between Limit Sell Order Arrivals and Other Order Arrivals

4. Order Arrival Model and Order Simulation

4.1 I-variate Hawkes-E(K) Process [10]

From the analysis of the London Exchange Data, especially the Cross Correlations among different kind of orders, it is quite clear that the arrival of one certain type of order will in some sense trigger the arrival of other types of order. For example, if a proportion of the players in the market are **trend followers**, then when they observe a large amount of Market Sell Orders come in (i.e, they observe the best bid price is plunging), they will follow this trend and place sell orders as well. On the other hand, there is another type of players existing in the market – the **Contrarian Investors** – who seek to do the opposite of what the majority of investors. They have a different philosophy: If under the same situation, i.e, a large amount of Market Sell Orders are coming into the market, then the contrarian investors will place buy orders in the hope of buying the shares in a price lower than it should be. Thus, in the presence of these two types of players in the market, it's actually quite difficult to predict the movement of stock price by using a simple model.

In light of the "Cross-Exciting" and "Self-Exciting" feature of the market, we decided to use a Multivariate Hawkes Process to model the dynamics. Be more specific, let $(N_t^1, N_t^2, N_t^3, N_t^4)$ be the counting process of the four major different orders in the market: Market Buy Order, Market Sell Order, Limit Buy Order, and Limit Sell Order, respectively. Also we consider the existence of Cancellation Order N_t^{cancel}, which empirically amounts to 30% of the total orders.

According to our discussion above, the arrival of a specific order will lead to a higher frequency of arrivals of the other orders. Mathematically, it means the arrival intensity $\lambda_t^i, i = 1,2,3,4$, will perform a jump at the event of order arrival. We assume the cancellation order's intensity maintains a constant level for model simplicity, but in the future if possible we can also take it into the Hawkes Model framework.

As talked above, $(N_t^1, N_t^2, N_t^3, N_t^4)$ are the counting processes, and $(\lambda_t^1, \lambda_t^2, \lambda_t^3, \lambda_t^4)$ are the corresponding intensities. The Hawkes Process is determined in the following way:

$$\lambda_t^i = \mu^i + \sum_{j=1}^{4} \int_0^t \alpha_{ij} e^{-\beta_{ij}(t-u)} dN_u^j \quad , i = 1,2,3,4$$

$$\lambda_t^{cancel} \equiv \mu^{cancel}$$

Where μ^i is the initial intensity of N_t^i, α_{ij} describes how big the jump of the intensity is. Notice α_{ij} is the impact of the jth type of order to the ith type of order. β_{ij} describes how fast the temporary impact vanishes. When β_{ij} is big, it means that the jump in the intensity is only momentary. It will fade away in very short time period.

After writing down the formulas, it is straight forward to build up a Market Order Simulator. Specifically, we use the thinning algorithm as follows:

Algorithm:

- Step 0: Set $N_0^i = 0, i = 1,2,3,4$. Initialize $\lambda_0^i = \mu^i$
- Step 1: At time t, we are able to project the intensity λ_u^i for $any\ u > t$ if nothing happens between $[t, u]$.
- Step 2: Simulate the arrival of 4 types of orders depending on our project of intensity, using a thinning algorithm.

- Step 3: Find out which order comes first, update $t \rightarrow t^*$, also update $\lambda^i_{t^*}$ according to the Hawkes formula.
- Step 4: Go to Step 1.

By using this algorithm, it is very efficient to simulate the market dynamics.

Here is a picture of the evolution of arrival intensities. We can see they move up and down largely simultaneous, just for different sizes. This comes from the Cross Exciting and Self Exciting property.

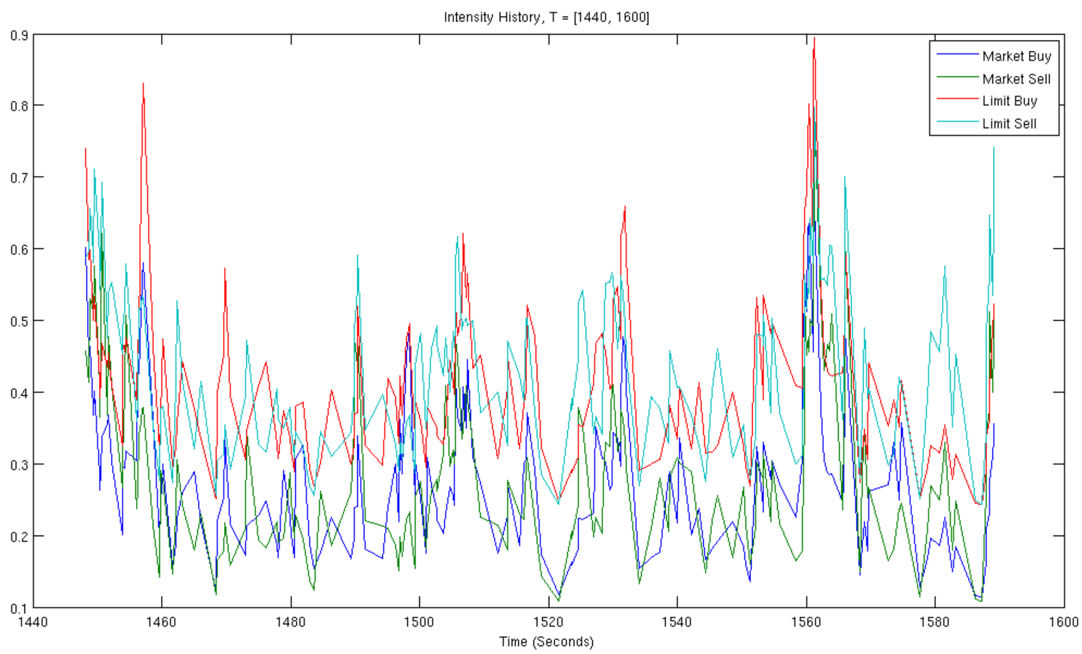

Figure 11. Intensity History of Orders

Also we can have a look at the arrival of Market Buy Orders and the corresponding intensity:

The blue bars represent the number of Market Buy Orders coming within a certain time interval. And the red curve represents the intensity. We can see they have very similar pattern, which is natural since intensity exactly describes the frequency of events happening.

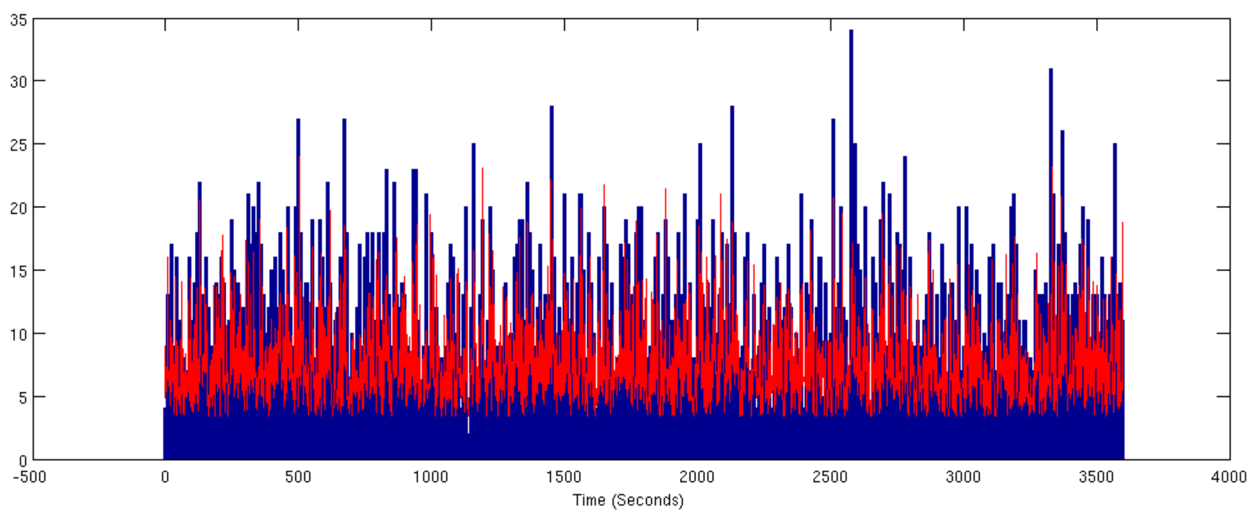

Figure 12. Count of Order Arrivals and Order Intensities

4.2 Parameter Calibration and Setup

We model the arrivals of market buy, market sell, limit buy and limit sell orders by 4-variate Hawkes process.

$$\lambda_t^{(i)} = \mu^i + \sum_{j=1}^{4} \int_{u<t} \alpha_{ij} e^{-\beta_{ij}(t-u)} dN_u^{(j)}, i = 1, 2, 3, 4$$

Given a set of high frequency data of market arrivals, we calibrate the parameters by maximizing the log likelihood function:

$$L = -\sum_{k=1}^{n} \sum_{j=1}^{4} \int_{t_{k-1}}^{t_k} \lambda_j(s) ds + \sum_{k=1}^{n} \sum_{j=1}^{4} \int_{t_0}^{t_n} \log \lambda_j(t_i -) dN_j(t_i)$$

t_k are the arrivals times of all orders. We implement by Matlab optimization toolbox. Note that the first term can be rewritten as

$$\begin{aligned}
&\sum_{k=1}^{n} \sum_{j=1}^{4} \int_{t_{k-1}}^{t_k} \lambda_j(s) ds = \sum_{j=1}^{4} \sum_{k=1}^{n} \int_{t_{k-1}}^{t_k} \lambda_j(s) ds \\
&= \sum_{j=1}^{4} \left\{ \mu_j \sum_{k=1}^{n} \int_{t_{k-1}}^{t_k} ds + \sum_{i=1}^{4} [\sum_{k=1}^{n} \int_{t_{k-1}}^{t_k} \alpha_{ij} e^{-\beta_{ij}(s-t_k)} ds] \right\} \\
&= \sum_{j=1}^{4} \left\{ \mu_j (t_n - t_0) + \sum_{i=1}^{4} \sum_{k=1}^{n} \left(-\frac{\alpha_{ij}}{\beta_{ij}} e^{-\beta_{ij}(s-t_k)} \Bigg|_{s=t_k}^{t_n} \right) \right\}
\end{aligned}$$

Therefore we can avoid evaluating numerical integrals in Matlab. However, we tried a number of "fmin" functions in Matlab but did not get satisfactory calibration results: we first derive a dataset by simulation with known parameters and try to calibrate, but the results are not necessarily close to the true values. Also, the estimated parameters are too sensitive to the initial guess values given real data. Possibly this is due to the large number of parameters: $4 \times 4\alpha_{ij}$'s, $4 \times 4\beta_{ij}$'s and $4\mu_i$'s, a total of 36.

Nonetheless if we set some parameters to be constants the number of parameters can be reduced dramatically. For example, if we assume the limit order book and the market order book are independent, the number of parameters is now 8 + 8 + 4 = 20. This improves the numerical properties of the optimization problem.

Alternatively, we use the result of descriptive statistics of raw data and the result of cross correlation coefficient to set up the parameters of Hawkes process:

$$\alpha_{ij} = 0.05 * \begin{pmatrix} 3 & 1 & 2 & 1 \\ 1 & 3 & 1 & 2 \\ 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 3 \end{pmatrix}$$

$$\beta_{ij} \equiv 1$$

$$\mu = (0.1125, 0.1055, 0.2417, 0.2396)$$
$$\mu^{cancel} = 0.3004$$

The μ^i are chosen to be proportional to the number of each type of orders from London Exchange Data. α_{ij} is set up in this way so that a Market Order will have a stronger impact on Limit Orders in the same direction than the Market Orders in the same direction. There are some empirical researches on why this is true.

4.3 Market Impact of Orders

In this part, we want to explore the market impact caused by a large Market Sell order we place at the beginning of a fixed time horizon. It is a very interesting question, and strongly related to the focus of our paper: Optimal Liquidation Strategy. We want to see whether a big size Market Order will permanently affect the price of the stock, or just temporarily and will return back after a certain 'resilience period'.

We did a simulated experiment as follows:

Order Type and Size: We place a Market Sell Order at the beginning of the day. The size varies from 100K shares to 1M shares.

Results: Each column is the simulated mean of the market mid price at some certain time. Each column represents different size of Market Sell Order we place at the very beginning. We call it 'First Order'.

Order	Time(sec)									
Size	360	720	1080	1440	1800	2160	2520	2880	3240	3600
0 K	2000.32	2000.17	2000.47	2000.76	2000.97	2000.88	2000.98	2001.14	2001.26	2001.33
100K	2004.71	2004.87	2005.28	2005.64	2005.51	2005.82	2005.99	2006.16	2006.26	2006.45
300K	2004.30	2004.35	2004.27	2004.46	2004.63	2004.67	2004.86	2004.99	2005.11	2005.12
500K	2005.86	2003.48	2003.50	2003.50	2003.53	2003.66	2003.68	2004.04	2004.15	2004.15
1M	2008.33	2005.13	2003.09	2002.98	2002.94	2003.05	2003.09	2003.17	2003.20	2003.40

Table 4. Market Mid Price History with Different Size of Market Sell Order We Place at Initial Time

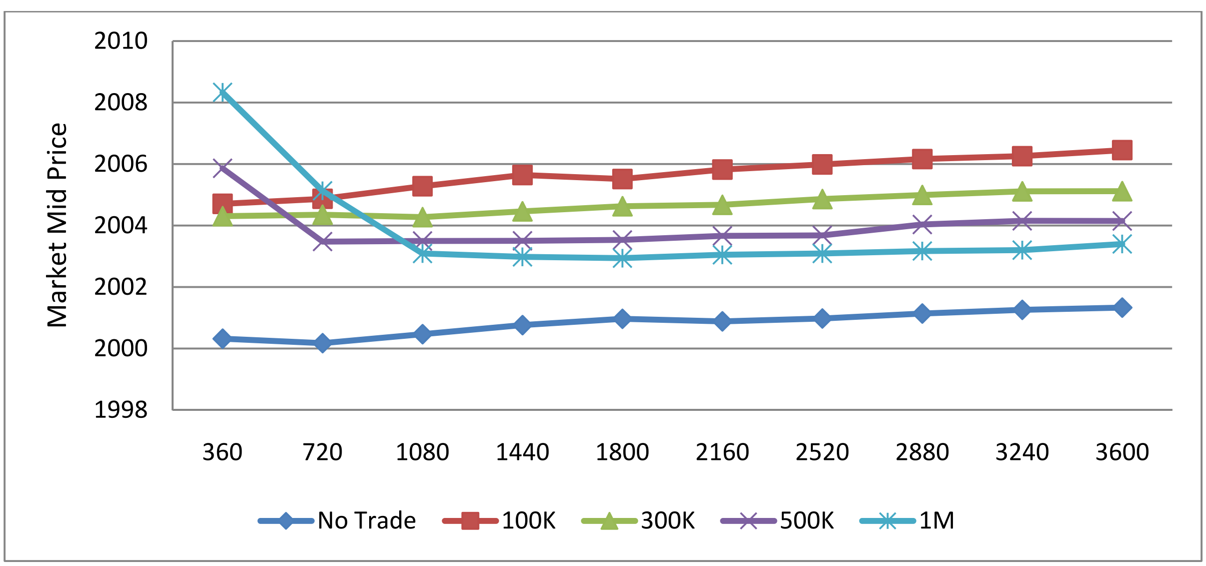

Figure 13. Market Mid Price History with Different Size of Market Sell Order We Place at Initial Time

There are several interesting features we observe. First of all, placing an order at the beginning, no matter what size it is, actually move the price upward to some degree. This can be explained in this way: Placing a Market Sell Order at the very beginning prevents the accumulation of Limit Buy Order Book. I.e, whenever a limit buy order book comes in, it will be matched with our placed Sell Order. Therefore the investors can only observe one side of the order book (the Sell Side) while the buy side of the order book is empty for the first few minutes. Hence the best ask price will be taken as the market mid price, which is higher than it would be if there were some Limit Buy orders in the market.

The explanation above can also be used to explain why at very beginning, the larger the 'First Order' is, the higher the market mid price is. However, as time goes, the 'First Order' will be eventually eaten up by Buy orders. After that, another side effect of placing the 'First Order' takes over. The existence of a large Market Sell Order, although prevents the market price from going down, also prevents the market price from moving up since no Limit Sell Order will be matched until the 'First Order' is fully fulfilled. The larger the size of 'First Order', the longer the effect is. That's why we see, when time goes beyond 1080 seconds, the larger the initial size, the lower the market mid price.

The third interesting phenomenon we see is: The shift of the Market Mid Price we caused, does not vanish as time goes. This is because, our model is based upon the philosophy that, an investor will place his limit order according to some certain distribution around the market mid price. So if you influence the market mid price, this impact will stay forever.

5. Optimal Liquidation Strategies

In this part, we will design the efficient liquidation strategies. To tell our orders from orders placed by other traders, we identify ourselves as "Trader Joe", and thus the orders placed by us are called Trader Joe's orders. Transaction cost or commission fees are not counted.

5.1 Scenario Setup

Though the focus of the project is to simulate the order book, it would still be helpful to explore some trading strategies which will give us some insights into the properties of the order book. The problem setup is as follows: Our Trader Joe has to liquidate some shares in a given period of time. For simplicity, he employs strategies that only involve market sell orders to guarantee execution, and aims to make the liquidation average price as high as possible. In particular, his liquidation average price is compared with the market benchmark of the Volume Weigh Average Price (VWAP).

$$P_{VWAP} = \frac{\sum_j P_j Q_j}{\sum_j Q_j}$$

We simulate the order book 20 times for each strategy and take the mean as the average liquidation price. Note that we do consider the market impact of the orders Trader Joe places: the orders changed the price as well as the arrival intensity of all kinds of orders.

5.2 Naïve Split Strategy

Trader Joe equally split the time horizon into N intervals of length Δt, and places market sell orders of the same volume at $\Delta t, 2\Delta t, ...(N-1)\Delta t$. The number of total shares Trader Joe liquidates is such designed that most of the last market order can be executed in time Δt. We hope that by splitting a large order into a number of smaller trades, the market impact alleviates, giving us better average prices. **Figure 14** illustrates this strategy.

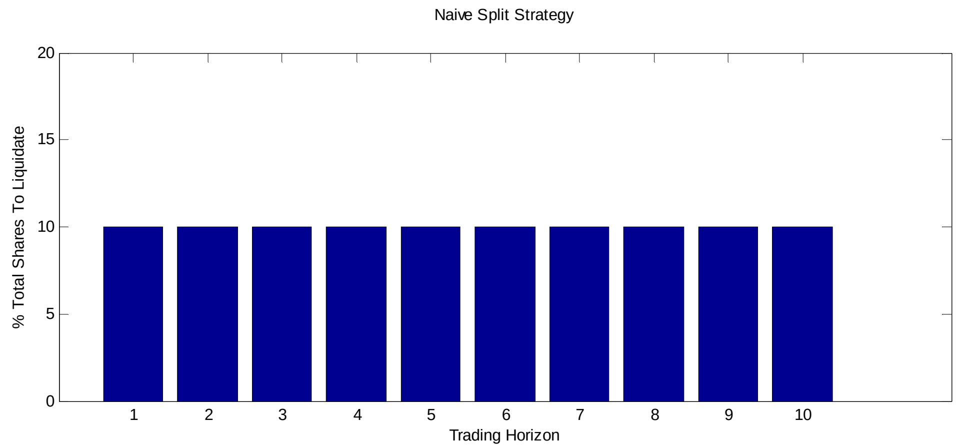

Figure 14

Figure 15 shows how the naïve split strategy performs. The y-axis is the percentage difference of liquidation average price minus VWAP. Note that we cannot beat VWAP by this strategy. We also observe two interesting monotonicities: first, the strategy works better with increasing number of splits. This implies the permanent market impact is not linear. Second, the average price worsens as the number of shares to liquidate increases, which is obviously due to larger market impacts.

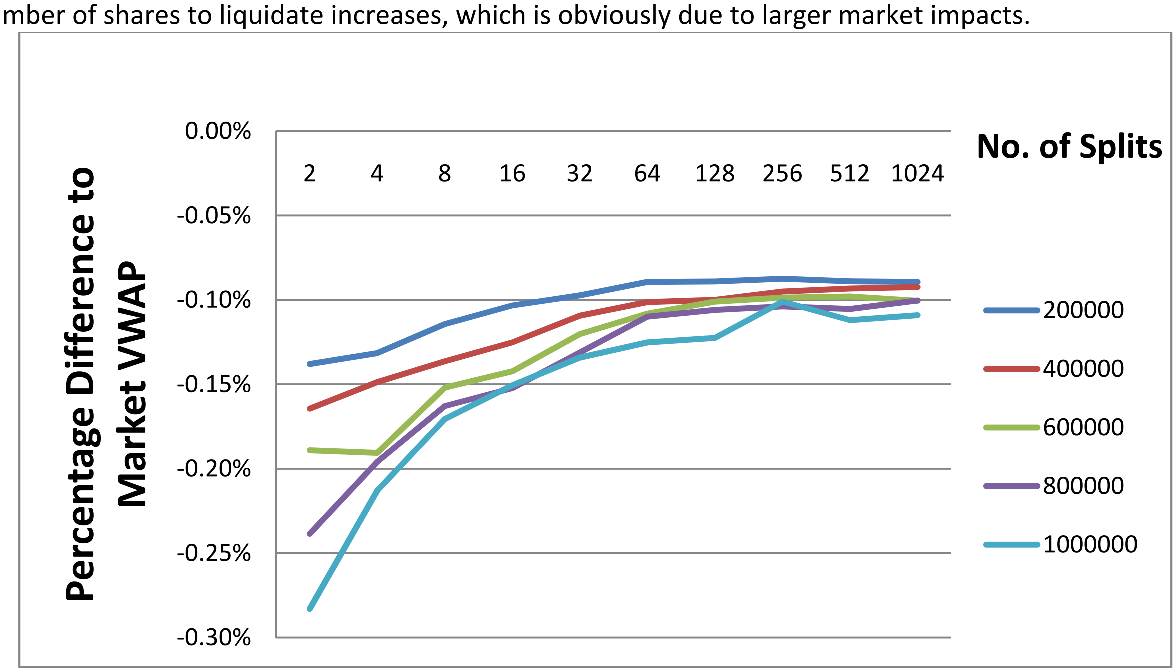

Figure 15 Performance of Naïve Split Strategy

5.3 Valley Strategy

With the valley strategy Trader Joe still trades (see **Figure 16**) at the same time spots but with different volumes. He places α percent of total shares in the first and last trades (cliff trades), and equal volume on every smaller trades in the middle. The valley strategy aims to improve the naïve split strategy. The first trade takes place without any previous market impact. Knowing the first big trade creates a large market impact, we place smaller trades afterwards to wait for the market to "cool down". We place a large trade in the end because we do not care about the market impact created by it; therefore, in theory, the two big trades are least affected by market impacts.

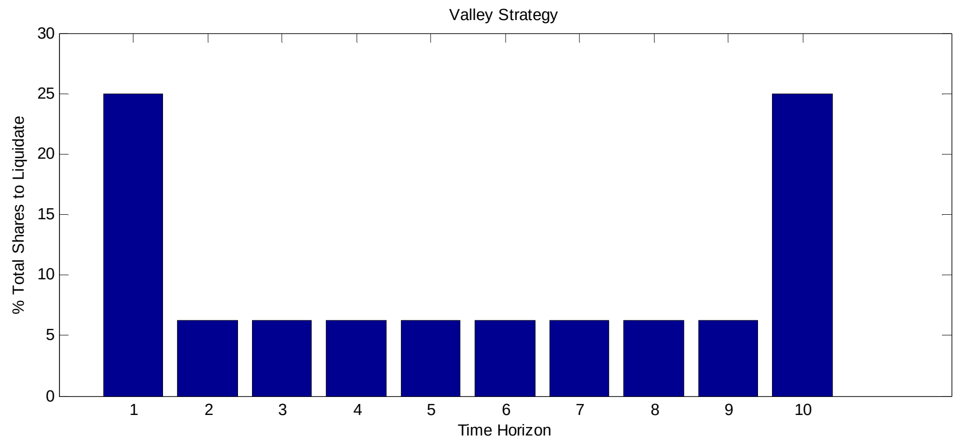

Figure 16

Fixing all the common parameters, we found that the valley strategy works better than the naïve one (**Figure 17**), and the inverted V-shape suggests there exists an optimal α^*.

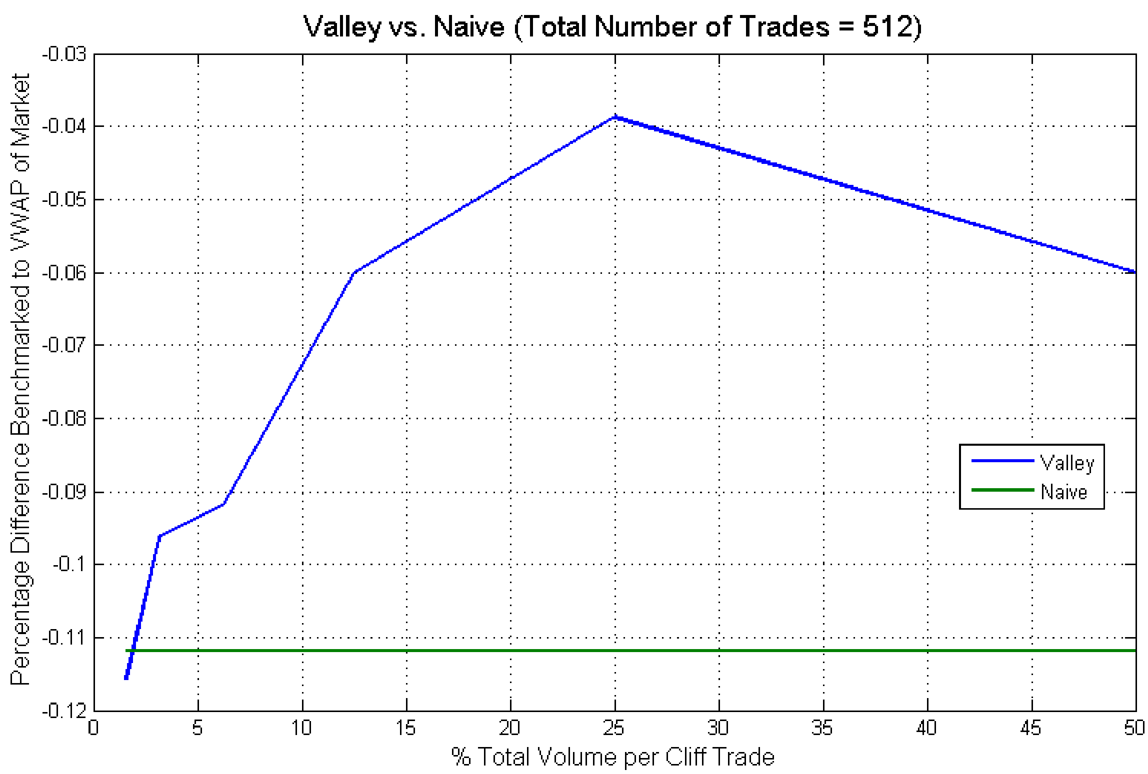

Figure 17 Performance of Valley Strategy

Figure 18 shows the performance with varying number of stocks to liquidate. It suggests the optimal α^* is not independent of the liquidation parameters. Still, both lines are under 0.

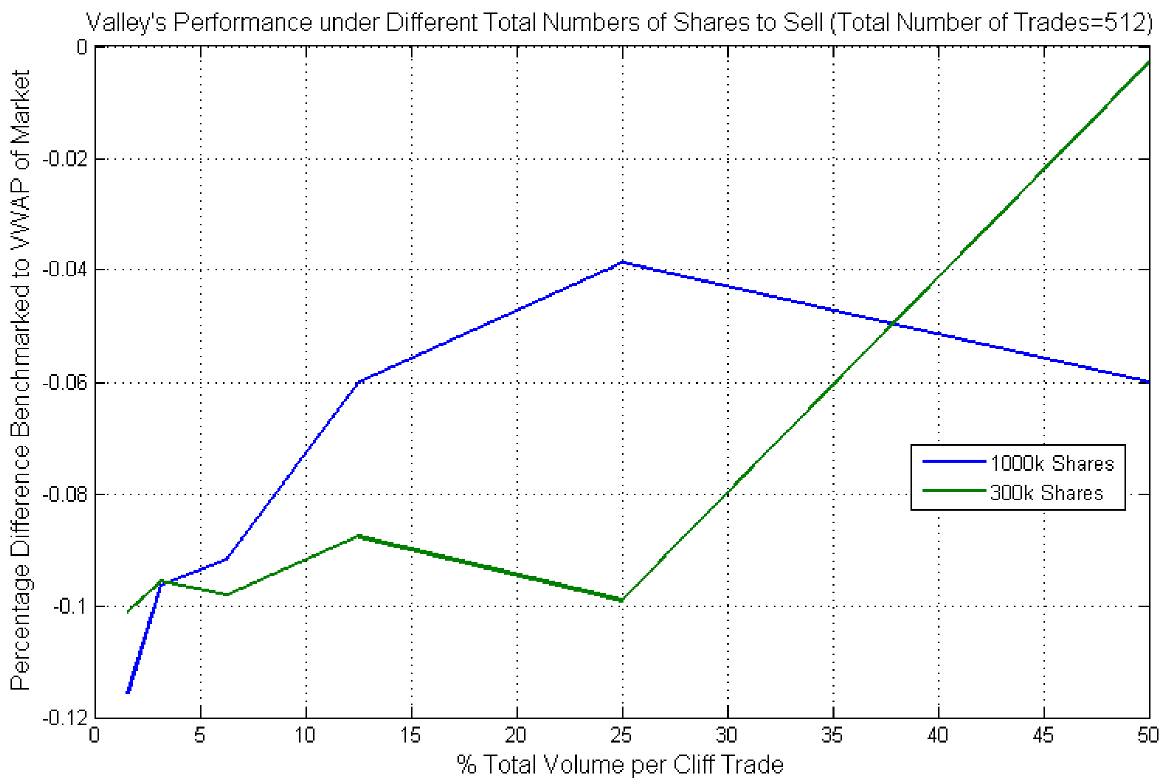

Figure 18

5.4 Relative Volume Strategy

The timing and volume of the trades in the two strategies above can be determined before the opening hours; however, any smart trader would take the current market condition into account before making decisions. Therefore we devised a relative volume (RV) strategy for our Trader Joe. Denote the total volume of all outstanding buy orders in the limit order book at time t as V_t^B; V_t^S for sell orders. Define

$$RV(t) = \frac{V_t^S}{V_t^B}$$

Therefore, given some t, small $RV(t)$ indicates that there are more demands than supply in the market, suggesting this could be a good timing to sell. We specify a threshold RV^* for Trader Joe: once $RV(t)$ falls below the threshold, we sell $\bar{V}$ shares. $\bar{V}$ equals the average volume attached to an order in the market: in this way we hope not to push the price upwards for two ticks.

An extra problem with this strategy is the risk of not getting executed. We set up an alert time $AlertT$ and if we did not finish liquidation before $T - AlertT$, we dump all the remaining shares in a single market order at time $T - AlertT$. The $AlertT$ is given by

$$AlertT = 1.5 \times \frac{V^{total}}{\bar{D}}$$

V^{total} is the total number of shares to liquidate; $\bar{D}$ is the average demand per second (including market buys and limit buys, estimated by long-terms means of the Hawkes intensities given simulation parameters). The coefficient 1.5 gives some buffer time. Since a portion of the shares would be liquidated before $T - AlertT$, very likely we are able to finish liquidation before the closing bell (this is also verified by simulation).

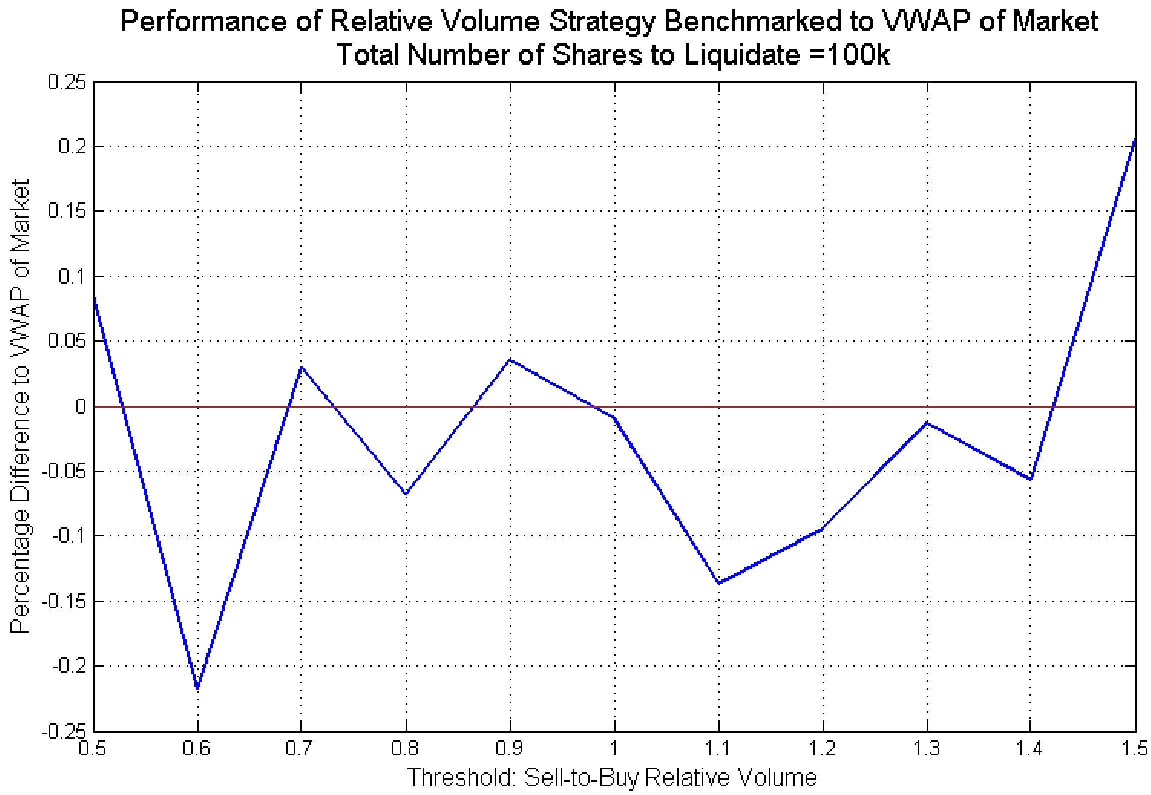

Figure 19

Figure 19 shows some joyfully surprising results. With smart choice of RV^* , it is possible to beat VWAP with market sell orders.

6. Conclusion and Future Work

In this project, we built a single-stock exchange system to match orders and rank open orders in order books. We reduced the computational complexity by analyzing the mechanism of order matching. Based on the Santa Fe Institute data for intraday orders, we discovered the self/cross-exciting features of order arrivals, estimated the order price shape, order volume shape, parameters for Hawkes process.

Placing a Market Sell Order at the very beginning prevents the accumulation of Limit Buy Order Book. I.e, whenever a limit buy order book comes in, it will be matched with our placed Sell Order. Therefore the investors can only observe one side of the order book (the Sell Side) while the buy side of the order book is empty for the first few minutes. Hence the best ask price will be taken as the market mid price, which is higher than it would be if there were some Limit Buy orders in the market.

The explanation above can also be used to explain why at very beginning, the larger the 'First Order' is, the higher the market mid price is. However, as time goes, the 'First Order' will be eventually eaten up by Buy orders. After that, another side effect of placing the 'First Order' takes over. The existence of a large Market Sell Order, although prevents the market price from going down, also prevents the market price from moving up since no Limit Sell Order will be matched until the 'First Order' is fully fulfilled. The larger the size of 'First Order', the longer the effect is. That's why we see, when time goes beyond 1080 seconds, the larger the initial size, the lower the market mid price.

The third interesting phenomenon we see is: The shift of the Market Mid Price we caused does not vanish as time goes. The reason is: our model is based upon the philosophy that an investor will place his limit order according to some certain distribution around the market mid price. So, if you influence the market mid price, this impact will stay forever.

In general, market orders guarantee execution while limit orders generate premium; therefore smarter strategies would incorporate both orders. This is much more complicated than strategies involving market sell orders only: Trader Joe needs to incorporate the information of i) number of remaining shares; ii) trend of the trading price; iii) risk of not getting executed into decision-making. We leave this to further research.

Our future work includes:

I. Assign different parameter values to Hawkes process simulator, and test the trading strategies' sensitivity;
II. Improve the performance of RV liquidation strategy by monitoring the order book, and placing limit sell orders instead of market sell orders for the horizon between $T-1.5LT(Q)$ and T;
III. Design RV buying strategy to achieve buying price lower than the market's VWAP;
IV. Combine the liquidation and buying strategies to design optimal market making strategies;
V. Incorporate transaction cost into the strategy designing;
VI. Explore algorithm to calibrate high-dimension Hawkes process.

7. Acknowledgement

We appreciate Prof. Kay Giesecke for defining a topic for us to work on, and guiding us through the whole process of our project. We also truly appreciate Gerry Tsoukalas, the teaching assistant for MS&E 444. Gerry was always accessible to us whenever we had a problem or whenever we needed any data or guidance, and shared with us a lot of smart thoughts which greatly accelerated our progress.

8. Reference

[1] http://www.ft.com/cms/s/0/c4baf670-1bfe-11df-a5e1-00144feab49a.html?nclick_check=1
[2] Berkowitz, S., D. Logue, and E. Noser. "The Total Cost of Transactions on the NYSE." Journal of Finance, 41 (1988), pp. 97-112.
[3] D. Bertsimas, A.W. Lo, Optimal control of execution costs, Journal of Financial Markets, 1 (1998), 1–50.
[4] R. Almgren, N. Chriss, Optimal execution of portfolio transactions, Journal of Risk, 3(2) (2000), 1–30.
[5] R.F. Almgren, Optimal execution with nonlinear impact functions and trading-enhanced risk. Applied Mathematical Finance, 10 (2003), 1–18.
[6] A. Subramanian, R.A. Jarrow The liquidity discount, Mathematical Finance, 11(4) (2001), 447–474.
[7] F.A. Longstaff, Optimal portfolio choice and the valuation of illiquid securities, Review of Financial Studies, 14(2) (2001), 407–431.
[8] D. Duffie, A. Ziegler, Liquidation risk, Financial Analysts Journal, May/June issue, (2003), 42–51.
[9] Optimal execution strategy of liquidation, Kwok, Yue Kuen, Lau, Ka Wo, Journal of industrial and management optimization, v. 2, no. 2, May 2006, p. 135-144
[10] A Cluster Process Representation of a Self-Exciting Process, Alan G. Hawkes and David Oakes, Journal of Applied Probability, Vol. 11, No. 3 (Sep., 1974), pp. 493-503.