Predicting the Next Executions Using High-Frequency Data

Ko Sugiura
Graduate School of Economics
Keio University
Tokyo, Japan
Email: ko.sugiura.0720@gmail.com

Teruo Nakatsuma
Faculty of Economics
Keio University
Tokyo, Japan
Email: nakatuma@econ.keio.ac.jp

Kenichiro McAlinn
Department of Statistical Science
Duke University
Durham, USA
Email: kenmcalinn@gmail.com

***Abstract*—With the progression of computer technology, the term "big data" has become more and more popular in the financial markets. In the literature of finance, this term, in many cases, means high-frequency data, whose size almost reaches as much as 10 GB per day. High-frequency trading (HFT) is, now, widely practiced in the financial markets and has become one of the most important factors in price formulation of financial assets. At the same time, a huge amount of data on high-frequency transactions, so-called tick data, became accessible to both market participants as well as academic researchers, which paved the way for studies on the efficacy of the high-frequency trading and the microstructure of the financial markets. The tick data contain all the information of all trades and are recorded in a thousands of a second, or a millisecond. Nevertheless there have been a great deal of works on investigating the features of HFT, and there have been a few works on application of them in forecast. In this paper, we try to develop a new time series model to capture the characteristics in tick data and use it to predict executions in high-frequency trading.**

Keywords–High-Frequency Trading; Tick Data; Executions; Duration Models; Bid-Ask Clustering.

I. INTRODUCTION

With the progression of computer technology, the term "big data" has become more and more popular in the financial markets. In the literature of finance, that word, in many cases, means high-frequency data, whose size almost reach as much as 10 GB per day. High-frequency trading is, now, widely practiced in the financial markets and has become one of the most important factors in price formulation of financial assets. At the same time, a huge amount of data on high-frequency transactions, so-called tick data, became accessible to both market participants as well as academic researchers, which paved the way for studies on the efficacy of the high-frequency trading and the microstructure of the financial markets. The tick data contain all the information of all trades and are recorded in a thousands of a second, or a millisecond. HFT is used not only in the stock markets but also in the markets for stock options and futures. Increased number of attention has been paid to this data, because it may help the mechanism of price formulation for financial assets. In fact, since the end of twentieth century, many researchers have worked on the practical study using tick data, and a lot of characteristics about high-frequency data have been reported.

One of the most famous series of study in tick data is the study on durations. Naturally, when the next execution occurs or when the price moves is the prime interest for market participants, particularly for specialists. It has long been known that there are largely two difficulties in duration data: discreteness of duration data and the sparsity in duration data. In other words, transaction data arrives with irregularly spaced intervals. However, [2] tackled these problems by proposing a new time-series model. Their model succeeded capturing the feature of clustering of durations. Afterwards, many papers have been devoted to their model and the model has a lot of variations and extensions ([1], [6], [8], etc.).

Another fact which is most frequently documented and stylized on high-frequency transaction data is bid-ask bounce. Bid-ask bounce is a phenomenon that execution prices tend to move back and forth between the best-ask and the best-bid. But, it is also pointed that, particularly in much shorter periods, after an execution at best-ask (best-bid), the next execution occurs more likely at best-ask (best-bid). That is, we can observe the runs of executions, which we named *bid-ask clustering*. The histogram of the runs appears to be more fat-tailed than a fair coin toss suggests. This means that executions don't occur completely at random. Despite a vast amount of literature [5][9] on reproducing the bid-ask clustering, there is little literature on application of this feature into forecast of executions.

In this paper, we try to develop a new time series model with combining the duration models and the feature of bid-ask clustering for forecasting executions in stock markets in the context of tick data. Our contribution is that we take explicitely the bid-ask clustering into consideration and that we focus on the best ask/bid pries themselves, not on the spreads or the price movements. From a practical point of view, we need to specify simultaneously the time and the price for the execution. Since these two pieces of information can fortunately be assumed to be independent, we can identify the probability on these two pieces of information separately. Then, our model comprises two parts and is intuitively understandable.

II. MARKET MICROSTRUCTURE

A. Principles of Financial Market

In general, a market is the platform where people trade something they want. At that place, transactions are made based on the agreement between prospective buyers and prospective sellers. Particularly in the modern financial market, buyers and sellers are matched through electronic servers, and they haggle over the price at a place called the order book. Following certain rules, all actions that take place in the financial market are recorded in order books. As an example of order book, Table I shows a snapshot of the order book

 ISBN: 978-1-61208-423-7

for the stock of Toyota Motor Corporation on 31 April, 2012. In this table, the column labeled "Volume (Ask)" shows how many stocks are on sale and the corresponding price in the middle column is the price at which these stocks will be sold. Such a price is called an ask price. In the same table, the column labeled "Volume (Bid)" shows how many stocks they are willing to buy and the corresponding price in the middle column is the price at which these stocks will be bought. Such a price is called a bid price. The best ask price is the lowest among ask prices while the best bid price is the highest among bid prices. The difference between the best ask price and the best bid price is called the bid-ask spread. Since no one wants to buy stocks at a price above the best ask price or to sell at a price below the best bid price, cells above the best ask price in the left column and those below the best bid price in the right column are empty by construction. Therefore, if they want to sell some of their stocks, they need to look at bid prices. If they want to buy some stocks, on the other hand, they have to consider ask prices.

When it comes to order processing method, two types of method are used; one is a call market and the other is continuous trading. In the former, orders are collected without execution until the certain time, and when the market is called, they start to be simultaneously matched. This style is used in the beginning and the ending of the trading session. In the latter, on the other hand, orders can be executed intermittently while the market is open. This method is mostly used during the trading hours excepting for the opening and closing of the market.

TABLE I. Order Book (31 April, 2012)

Volume (Bid)	Price (Yen)	Volume (Ask)
	⋮	⋮
	2822	23400
	2821	4200
	2820	17200
	2819	10600
	2818	3000
	2817	2100
	2816	2000
	2815	15400
4700	2814	
4400	2813	
5300	2812	
7300	2811	
2100	2810	
8600	2809	
2200	2808	
8300	2807	
⋮	⋮	

Since 1970's, a great deal of attention have been paid to the question how difference in a trading mechanism affects on a price discovery process in financial markets. Studies on this topic caught on especially after 1980's and the field has gained its own name: market microstructure. [7] provides a comprehensive overview of this topic. Although there are a tremendous amount of researches on market microstructure, the characteristics of order executions in a market tend to be translated into three aspects of transactions; prices, volumes and durations. In this section, we review some of the prominent works relating to these variables.

B. Tick Data

Table II shows a typical format of tick data. They are excerpts from by the Nikkei NEEDS database which will be used for our empirical study. As shown in Table II, the data are composed of snapshots of order books. For example, the best ask price is in ③ and seven ask prices are in ④ $\sim$ ⑩ above the best one while the best bid price is in ⑫ and seven bid prices are in ⑬ $\sim$ ⑲ bellow the best one. Additionally, the data also have the information of executions (②). Each line contains a variety of information. A full description of the information is given in Table III.

TABLE II. Tick Data

① 150020120131111 11 7203 0953333002+00000000197 0+0001031200128
② 110020120131111 11 7203 0953 03003+00002814 16 0+0000000400 0
③ 120020120131111 11 7203 0953333004+00002815 0 0+0000015400128
④ 150020120131111 11 7203 0953333004+00002816 1 0+0000002000128
⑤ 150020120131111 11 7203 0953333004+00002817 2 0+0000002100128
⑥ 150020120131111 11 7203 0953333004+00002818 3 0+0000003000128
⑦ 150020120131111 11 7203 0953333004+00002819 4 0+0000010600128
⑧ 150020120131111 11 7203 0953333004+00002820 5 0+0000017200128
⑨ 150020120131111 11 7203 0953333004+00002821 6 0+0000004200128
⑩ 150020120131111 11 7203 0953333004+00002822 7 0+0000023400128
⑪ 150020120131111 11 7203 0953333004+00000000 97 0+0001237300128
⑫ 120020120131111 11 7203 0953333005+00002814128 0+0000004700128
⑬ 150020120131111 11 7203 0953333005+00002813129 0+0000004400128
⑭ 150020120131111 11 7203 0953333005+00002812130 0+0000005300128
⑮ 150020120131111 11 7203 0953333005+00002811131 0+0000007300128
⑯ 150020120131111 11 7203 0953333005+00002810132 0+0000002100128
⑰ 150020120131111 11 7203 0953333005+00002809133 0+0000008600128
⑱ 150020120131111 11 7203 0953333005+00002808134 0+0000002200128
⑲ 150020120131111 11 7203 0953333005+00002807135 0+0000008300128
⑳ 150020120131111 11 7203 0953333005+00000000197 0+0001031200128
㉑ 120020120131111 11 7203 0953333006+00002815 0 0+0000015400128

TABLE III. Definition of Items in Tick Data

$$\underbrace{1200}_{}\ \underbrace{20120131}_{(I)}\ \underbrace{11111}_{}\ \underbrace{7203}_{(II)}\ \underbrace{0953}_{(III)}\ \underbrace{33}_{(IV)}\ \underbrace{30}_{(V)}\ \underbrace{06}_{(VI)}\ \underbrace{+00002815}_{(VII)}\ \underbrace{0}_{(VIII)}\ \underbrace{0+0000015400}_{(IX)}\ 128$$

Number	Item Name	Defenition
(I)	Date of Data	YYYYMMDD (Y: Year, M: Month, D: Day)
(II)	Companies' Codes	Four-digit numbers for companies
(III)	Time 1	HHMM (H: Hour, M: Minute)
(IV)	Classification of Records	"0": Executed "1": Not executed
(V)	Time 2	SS (S: Second)
(VI)	Consecutive Numbers	Consecutive Numbers in the same times
(VII)	Prices	Unit: Yen
(VIII)	Classification of Orders	"16": Executed at the best ask price "48": Executed at the best bid price "0": Other cases
(IX)	Volumes	Unit: Stocks

III. Non-Randomness of Executions

A. Bid-Ask Clustering

Despite the fact that the sample size of tick data is large enough to justify the use of the law of large number in the standard situation, it is recognized among researchers that the variance of a tick-data-based estimator such as realized volatility tends to be extremely high and difficult to obtain a stable estimate. Many researchers proposed possible explanations of this phenomena. One promising answer to this question is that high-frequency tick-by-tick price series we observe contain

 ISBN: 978-1-61208-423-7

some kinds of observation error. One of the most influential component of the error is called *bid-ask bounce*, which stems from back and forth movements of prices between bid and ask prices. There are many works treating this phenomena. Among them, [3] analyzes the mechanism of bid-ask bounce from the perspective of bid-ask spread, and gives an intuitively simple explanation about the cause. Here we shall briefly review his work.

Another well-known phenomenon found in tick data is *bid-ask clustering*. This term refers to the stylized fact that an execution at the best ask (bid) price tends to be followed by another execution at the best ask (bid). Figure 1 shows a histogram of the length of runs in executions[1]. As the length of a run increases, the number of runs are observed more than the geometric distribution (fair-coin toss) implies. This tendency of serial correlation has been analyzed in a number of works. Particularly, many have been devoted for elucidating nature of this feature, or reproducing the phenomena using the agent-based simulations. For example, [9] pointed that the investors' order submissions were exactly influenced by the state of the order book, and this fact indeed generated serial correlation in volume, volatility and order signs. Moreover, [5] considered an order splitting strategy of traders, which split their large orders into smaller ones. Although this strategy was originated from minimization of market impacts, they showed that the minimization strategy leads to the serial correlation.

As we have seen here, there have been a tremendous amount of works on market microstructure. However, there exist only a few number of papers which studied price movements in terms of best ask and bid prices, not bid-ask spreads or execution prices. In our proposed model, we explicitly treat whether execution occurs at the best ask or the best bid. We also incorporate bid-ask clustering into our model and try to take advantage of it in forecasting price movements and making investment strategies. In the next chapter, we will further elaborate these points and lay our framework for prediction of the future execution.

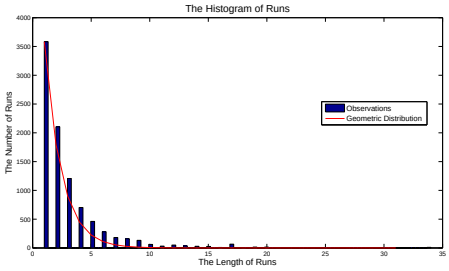

Figure 1. Histogram of Runs (Executions at Best Ask)

IV. Duration Models

A. Autoregressive Conditional Duration (ACD) Model

Although econometricians have traditionally worked on analyzing regularly spaced data, i.e. daily, monthly and yearly data, duration data have some difficulties in modeling. First of all, the data are recorded inherently in irregular time intervals. In order to address this matter, [2] assumed that the arrival times are random variables which follow a point process. The second problem in duration data is that they are necessarily non-negative. Traditionally in the context of finance, the random variables we are interested in may take both negative and positive values. When it comes to duration, however, it is essential to pose a restriction of no-negative on the model. Lastly, it is a well known fact that clusterings can be seen in duration data. This phenomena is thought to stem from a simple causality: the more active a market become, the more transactions we observe. Since the same feature was recognized in volatility and it was modeled by GARCH models, [2] introduced the similar method in duration models.

For the sake of tackling the problems just mentioned above, [2] introduced Autoregressive Conditional Duration (ACD) models. As its name suggests, the ACD models are specified in terms of the conditional density of the durations. Although we recall here its simplest version for simplicity, the discussion can be generalized into higher orders. Letting $\delta_n = t_i - t_{i-1}$ and ψ_i be the interval between two arrival times and the conditional expectation of the i-th duration, respectively, we have:

$$\psi_i = E_{i-1}(\delta_i | x_{i-1}, \theta), \tag{1}$$

where θ is the other parameters. The ACD models consists of this parameterizations and the following assumption:

$$\delta_i = \psi_i \epsilon_i, \tag{2}$$

where $\{\epsilon_i\}$ is a sequence of $i.i.d.$ random variables with positive support. Although the general form of ACD models can be written by the combination of (1) and (2), there are proposed a number of variations on the assumption of $\{\epsilon_i\}$. Engle and Russell, in their paper, introduced the EACD model in which the "E" represented the exponential assumption on the innovation terms. They mentioned the first order one of the EACD models is often the very successful and this is represented as:

$$\begin{aligned} \psi_i &= \omega + \phi\delta_{i-1} + \kappa\psi_{i-1} \\ \delta_i &= \psi_i \epsilon_i, \end{aligned}$$

where $\{\epsilon_i\}$ follows $Exponential(\lambda)$, $\omega > 0$, and $\phi, \kappa \geq 0$.

B. Stochastic Conditional Duration (SCD) Model

About fifteen years after the appearance of ACD models, [1] introduced a state-space class of parametric models for durations, which they called *stochastic conditional duration (SCD)* models. In their models, a latent variable cause the evolution of the duration, and equally it capture the information which cannot be observed directly. Then, SCD models are composed of two stochastic equations, namely state equation and observation equation, whereas ACD models have a stochastic equation and a deterministic equation. In SCD models, the conditional expected duration of ACD model become a random variable. In terms of shapes of models, ACD models and SCD models are similar to GARCH models and SV models, respectively. The simplest version of SCD models is expressed as

$$\begin{aligned} \psi_i &= \omega + \theta\psi_{i-1} + u_i \\ \delta_i &= exp(\psi_i)\epsilon_i, \end{aligned}$$

where $\{u_i\}$ follows a Gaussian distribution and $\{\epsilon_i\}$ a distribution with positive support. The innovation term of the observation equation can take some form, and [1] mentioned the case

[1] In fact, this series of data reject the null hypothesis in run test.

 ISBN: 978-1-61208-423-7

of Weibull distribution and gamma distribution. Although [1] used the combination of quasi-maximum likelihood estimation and Kalman filter in parameter estimation, we employed a more general method called *particle filter*.

C. Parameter Estimation: Particle Filter

When it comes to the parameter estimation of state-space models, there arise two problems: filtering hidden state variables and estimating model parameters. After the development of Kalman filter, these problems have been discussed in Bayesian framework, which is called *particle filter*. Take a general form of non-Gaussian nonlinear state-space model for time series y_t, for example;

$$\begin{aligned} x_t &= f(x_{t-1}, v_t) \\ y_t &= h(x_t, w_t), \end{aligned}$$

where x_t is a hidden state variable, and v_t and w_t are both noise terms. This model implies the information about two types of distribution: the distribution of x_t conditioned to x_{t-1}, $p(x_t|x_{t-1}, \theta)$, and the distribution of y_t conditioned on x_t, $p(y_t|x_t, \theta)$, where θ represents model parameters. Besides, let the distribution of x_0 and the distribution of θ be $p(x_0|\theta)$ and $p(\theta)$, respectively. Thus, the state-space model can be denoted by

$$\begin{aligned} x_t|x_{t-1} &\sim p(x_t|x_{t-1}, \theta) \\ y_t|x_t &\sim p(y_t|x_t, \theta) \\ x_0 &\sim p(x_0|\theta) \\ \theta &\sim p(\theta), \qquad \text{for } t = 1, \cdots, T. \end{aligned}$$

Ordinary particle filter is interested in only hidden state variables given the model parameters, and its procedure consists prediction step and filtering step. Prediction distribution at $t-1$ is given by filtering distribution at $t-1$ and prediction distribution at $t-1$.

$$\begin{aligned} p(x_t|y_{1:t-1}) &= \int p(x_t, x_{t-1}|y_{1:t-1}) dx_{t-1} \\ &= \int p(x_t|x_{t-1}, y_{1:t-1}) p(x_{t-1}|y_{1:t-1}) dx_{t-1} \\ &= \int p(x_t|x_{t-1}) p(x_{t-1}|y_{1:t-1}) dx_{t-1} \end{aligned}$$

Filtering distribution at time t is obtained by observation distribution at t and prediction distribution at $t-1$.

$$\begin{aligned} p(x_t|y_{1:t}) &= p(x_t|y_{1:t-1}, y_t) \\ &= \frac{p(x_t, y_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \\ &= \frac{p(y_t|x_t, y_{1:t-1}) p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \\ &= \frac{p(y_t|x_t) p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \\ &= \frac{p(y_t|x_t) p(x_t|y_{1:t-1})}{\int p(y_t, x_t|y_{1:t-1}) dx_t} \\ &= \frac{p(y_t|x_t) p(x_t|y_{1:t-1})}{\int p(y_t|x_t) p(x_t|y_{1:t-1}) dx_t}. \end{aligned}$$

Naturally, this equation is nothing but Bayes' theorem[2]. As is often the case with non-linear and non-Gaussian state-space models, these computations, especially integrations are too complicated for analytical implementation. Then, Markov Chain Monte Carlo (MCMC) method started to be used rapidly to accomplish the integration calculus in wide range of the research area at the end of twentieth century, thanks to a remarkable development in computer technology that helps to simulate a good amount of calculation. Particle filter is also a feat of MCMC method[3], and is always implemented by a numerical way. The merit of particle filter is that it enables us to make a on-line estimation of parameters and predictions. In order to estimate hidden states variables and model parameters jointly, [4] proposes the application of extended state vector for parameter estimation, which he calls it *self-organiged state-space model*. We employed his method in our research.

Algorithm 1 Algorithm for Particle Filter

(1) Give an initial set of particles $\{x_{0|0}^{(i)}\}_{i=1}^m$, where m is the number of particles.
(2) Repeat the following steps for $t = 1, \cdots, T$, where T is the length of data.
a. Generate a random numbers which represent state noise $v_t^{(i)} \sim q(v_t)$, for $i = 1, \cdots, m$.
b. Compute $x_{t|t-1}^{(i)} = f(x_{t-1|t-1}^{(i)}, v_t^{(i)})$, for $i = 1, \cdots, m$.
c. Compute $\lambda_t^{(i)} = p(y_t|x_{t|t-1}^{(i)})$, for $i = 1, \cdots, m$.
d. Compute $\beta_t^{(i)} = \lambda_t^{(i)} / \sum_{i=1}^m \lambda_t^{(i)}$, for $i = 1, \cdots, m$.
e. Resample particles $\{x_{t|t}^{(i)}\}_{i=1}^m$ from $\{x_{t|t-1}^{(i)}\}_{i=1}^m$ with the weight $\beta_t^{(i)}$.

V. Empirical Analysis

A. Model Description

We introduced some notations: n, τ, δ, r, and X. Let t be the time measured in millisecond, and n be the number of execution observed by time t. And τ_n is a random variable for representing the time when the n-th execution is observed, and the duration in our interest is represented by δ_{n+1}, satisfying

$$\delta_{n+1} = \tau_{n+1} - \tau_n. \tag{3}$$

We defined X_n as a random variable representing at which price the next execution occurs. That is, X_n equals to -1 when we observe the n-th execution at best ask price, and to 1 when we observe the n-th execution at best bid price:

$$X_n = \begin{cases} -1 & if \ \ best\ ask \\ 1 & if \ \ best\ bid. \end{cases}$$

Since we highlight the continuity of executions in our research, we define r_n as the length of the last run including X_n, and we count it up as follow;

$$r_n = \begin{cases} 1 & if \ \ X_n \neq X_{n-1} \\ r_{n-1} + 1 & if \ \ X_n = X_{n-1}. \end{cases}$$

From the practical view point, we need to know two pieces of information: when the next execution will occur and at

[2] $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$

[3] In fact, some articles call particle filter as Monte Carlo filter.

which price the execution will occur. For this purpose, we set the target probability as bellow:

$$P(X_{n+1} = k, \tau_{n+1} \in (t, t + \Delta t] | X_n, \tau_n, r_n) \qquad k = -1, 1$$

where Δt denotes a time window which will be fixed before simulation[4]. Under the condition of independence on X_{n+1} and τ_{n+1}, the target probability can be decomposed into two parts by the law of conditional probability:

$$\begin{aligned} &P(X_{n+1} = k, \tau_{n+1} \in (t, t + \Delta t] | X_n, \tau_n, r_n) \\ &= P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) P(X_{n+1} = k | X_n, r_n), \end{aligned}$$

and they were estimated separately: $P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n)$ was estimated by duration models and $P(X_{n+1} = k | X_n, r_n)$ was by historical frequency.

For the sake of applying the duration models, we rewrite $P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n)$ in the context of durations using the equation (3). Substituting it, we obtain a following duration representation:

$$\begin{aligned} &P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) \\ &= P(t < \tau_{n+1} \leq t + \Delta t | \tau_n, r_n) \\ &= P(t < \tau_n + \delta_{n+1} \leq t + \Delta t | \tau_n, r_n) \\ &= P(t - \tau_n < \delta_{n+1} \leq t - \tau_n + \Delta t | \tau_n, r_n). \end{aligned}$$

We estimated this by the ACD model and the SCD model. The parameter estimation of both models were conducted through particle filter, because it enables us to update on line the parameter estimates. Although particle filter is usable in continuous time, we, in the process of particle filter, update this probability as we observe a new order or execution. When we observe an execution, we update the probability with recalculating a predictive distribution. On the other hand, when we observe an order, we update the probability without recalculating a predictive distribution. Calibrations of the probability were conducted by moving a time window, Δt, on the predictive distribution. Thus, when we observe an execution, the probability is calculated as

$$\begin{aligned} &P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) \\ &= P(0 < \delta_{n+1} \leq \Delta t | \tau_n, r_n) \\ &= \frac{\int_0^{\Delta t} f(\delta_{n+1} | \tau_n) d\delta_{n+1}}{\int_0^{\infty} f(\delta_{n+1|} | \tau_n) d\delta_{n+1}} \\ &= \frac{\int_0^{\Delta t} f(\delta_{n+1} | \tau_n) d\delta_{n+1}}{\int_0^{\Delta t} f(\delta_{n+1|} | \tau_n) d\delta_{n+1} + \int_{\Delta t}^{\infty} f(\delta_{n+1|} | \tau_n) d\delta_{n+1}} \\ &= \int_0^{\Delta t} f(\delta_{n+1} | \tau_n) d\delta_{n+1}, \end{aligned}$$

where $f(\cdot)$ denotes a predictive distribution. Similarly, when we observe an order, the probability is given by

$$\begin{aligned} &P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) \\ &= P(t - \tau_n < \delta_{n+1} \leq t - \tau_n + \Delta t | \tau_n, r_n) \\ &= \frac{\int_{t-\tau_n}^{t-\tau_n+\Delta t} f(\delta_{n+1} | \tau_n) d\delta_{n+1}}{\int_{t-\tau_n}^{\infty} f(\delta_{n+1|} | \tau_n) d\delta_{n+1}} \\ &= \frac{\int_{t-\tau_n}^{t-\tau_n+\Delta t} f(\delta_{n+1} | \tau_n) d\delta_{n+1}}{\int_{t-\tau_n}^{t-\tau_n+\Delta t} f(\delta_{n+1|} | \tau_n) d\delta_{n+1} + \int_{t-\tau_n+\Delta t}^{\infty} f(\delta_{n+1|} | \tau_n) d\delta_{n+1}}. \end{aligned}$$

After estimating the duration, we calculate the probability $P(X_{n+1} = k | X_n, r_n)$ using the histogram of length of runs. When we observed $X_n = k$ and a run of executions whose length was $\bar{r}$, the probability we wanted to know was given by

$$\begin{aligned} &P(X_{n+1} = k | X_n = k, r_n = \bar{r}) \\ &= \frac{\sum_{i=n+1}^{\infty} P(X_{i+1} = k | X_i = k, r_i = \bar{r} + i - n)}{P(X_{n+1} \neq k | X_n = k, r_n = \bar{r}) + \sum_{i=n+1}^{\infty} P(X_{i+1} = k | X_i = k, r_i = \bar{r} + i - n)}. \end{aligned}$$

B. Algorithms

In order to compare the performance of our model, we introduced 5 types of algorithms. The difference comes from the estimation method of two probabilities we divided. In *Model 1* and *Model 2*, $P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n)$ of both models were estimated by the SCD models. But $P(X_{n+1} = k | X_n, r_n)$ of the former model was given by the "bid-ask clustering" or the histogram of length of runs, while that of the latter was by a completely random method, namely a fair coin toss. Similarly in *Model 3* and *Model 4*, the $P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n)$ in both models were calculated through the ACD models, whereas $P(X_{n+1} = k | X_n, r_n)$ of the former was by the "bid-ask clustering" and that of the latter was by a fair coin toss. Lastly, *Model 5* was comprise of completely and totally random method, that is, both probability $P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n)$ and $P(X_{n+1} = k | X_n, r_n)$ were given by fair coin tosses. Since it is reported that the SCD model fit better than the ACD model, we expected the Model 1 to show the best performance. Using these algorithms, we made predictions about executions: whether execution occurs in Δt or not, and if does, at which best prices the execution occurs. Then, our prediction was categorized into three types: *no execution*, *execution at best ask price* and *execution at best bid price*. The algorithms of the Model 1 is stated bellow as an example:

Algorithm 2 Model 1

(Step 1) *Execution* **or** *No Execution*
We estimate $P(\tau_{n+1} \in (t, t+\Delta t] | \tau_n, r_n)$ by the SCD model, and we predict

$$\begin{cases} No\ Execution & if\ P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) < 0.5 \\ Execution & if\ P(\tau_{n+1} \in (t, t + \Delta t] | \tau_n, r_n) > 0.5 \end{cases}$$

(Step 2) *Best Ask* **or** *Best Bid*
If we predict *Execution*, we predict

$$X_{n+1} = \begin{cases} 1 & if\ P(X_{n+1} | X_n, \tau_n) > 0.5 \\ -1 & if\ P(X_{n+1} | X_n, \tau_n) < 0.5 \end{cases}$$

[4]Note that a trivial fact:

$$\begin{aligned} &P(X_{n+1} = -k, \tau_{n+1} \in (t, t + \Delta t] | X_n, \tau_n, r_n) \\ &= 1 - P(X_{n+1} = k, \tau_{n+1} \in (t, t + \Delta t] | X_n, \tau_n, r_n). \end{aligned}$$

C. Data Description

We applied the proposed model into the real stock data of Toyota Motor Corporation which contained the signs of every order and execution. We used the data of 4th-18th January, 2012 (10 trading days) as a learning period and the data of 19th-31st January, 2012 (9 trading days) as a prediction period. And we omitted the first 30 minutes, because we intended to eliminate the influence of call market method adopted just before opening of the market. Then, the time of the data ranges from 9:30 to 11:30 and from 13:00 to 15:00.

TABLE IV. Statistical Information of the Data Used

	4th-18th Jan	19th-31st Jan
Number of Observations	301517	367320
Best Ask (%)	5.50%	6.54%
No Execution (%)	88.86%	87.74%
Best Bid (%)	5.64%	5.72%

D. Empirical Results

For the sake of summarizing the results, we broke the observations down into the following table, which was used in [10]:

		Actual			
		Best Ask	No Execution	Best Bid	
Predicted	Best Ask	N_{11}	N_{12}	N_{13}	$N_{1.}$
	No Execution	N_{21}	N_{22}	N_{23}	$N_{2.}$
	Best Bid	N_{31}	N_{32}	N_{33}	$N_{3.}$
		$N_{.1}$	$N_{.2}$	$N_{.3}$	N

In the empirical analysis, we made a forecast about executions as we observed an order and/or execution. And Δt after, we examined whether the forecasts were right or wrong. For example, if we forecast there will be a execution at best ask price in Δt at time s, and actually there is a execution at best ask between time s and $s+\Delta t$, we count this prediction adding one to N_{11}. In order to summarize this table, we defined some measures to compare the performance:

- $\alpha = \dfrac{N_{11} + N_{22} + N_{33}}{N}$
- $\beta = \dfrac{N_{11} + N_{33}}{N_{.1} + N_{.3}}$
- $\gamma = \dfrac{N_{11} + N_{33}}{(N_{11} + N_{13}) + (N_{31} + N_{33})}$
- $\delta_1 = \dfrac{N_{11}}{N_{.1}}$, $\delta_2 = \dfrac{N_{22}}{N_{.2}}$, $\delta_3 = \dfrac{N_{33}}{N_{.3}}$

α is the ratio of correct predictions among all the predictions. β is the ratio of correct predictions when we observe executions. γ is the ratio of correct predictions when we predicted executions. δ_1, δ_2 and δ_3 are the ratio of correct predictions when we predicted executions at best ask, when we predicted no executions and when we predicted executions at best bid, respectively.

The simulation results are summarized in the TABLE V, using the measures mentioned above. As for the case with $\Delta t = 1$, Model 5 performed best in β, δ_1 and δ_3. Model 2 was the best model for δ_2. The remaining measures α and γ are takes the highest in Model 1, which shows the second best performance in terms of the other measures. Regarding the case with $\Delta t = 2$, Model 1 outperformed all the other models in all measures.

TABLE V. Performance Measures for the Five Models

	Model 1	Model 2	Model 3	Model 4	Model 5
α	0.5612	0.5174	0.4815	0.4702	0.3675
β	0.2322	0.1458	0.1091	0.0854	0.2500
γ	0.7696	0.4985	0.6314	0.4943	0.5011
δ_1	0.2288	0.1409	0.1087	0.0844	0.2511
δ_2	0.9331	0.9373	0.9026	0.9052	0.5002
δ_3	0.2360	0.1513	0.1096	0.0866	0.2488

	Model 1	Model 2	Model 3	Model 4	Model 5
α	0.5292	0.4557	0.4542	0.4004	0.3403
β	0.4435	0.3301	0.4022	0.3217	0.2505
γ	0.6726	0.4998	0.6308	0.5006	0.5002
δ_1	0.4402	0.3260	0.4006	0.3140	0.2515
δ_2	0.6808	0.6775	0.5460	0.5396	0.4988
δ_3	0.4472	0.3348	0.4041	0.3305	0.2495

$*$ The above table is for the case with $\Delta t = 1$ and the bellow one is for the case with $\Delta t = 2$

VI. Conclusion & Discussion

In our model, we take the feature of bid-ask clustering explicitly into consideration. This arrangement makes it possible to forecast next executions more precisely. Despite the good performance of our model, this doesn't immediately suggest that people can make money from the financial markets, because there is a general rule of price-priority and time-priority in the markets. However, it may bring us an insight about formation of market trends. Moreover, with further studies on the bid-ask clustering, the accuracy of the model can be improved. For example, it might be useful if we take not only the length of runs but also volumes and prices into consideration.

Acknowledgment

This work is supported in part by a Grant-in-Aid for the Leading Graduate School program for "Science for Development of Super Mature Society" from the Ministry of Education, Culture, Sport, Science, and Technology in Japan.

References

[1] Bauwens, L., and Veredas, D. (2004), The Stochastic Conditional Duration Model: A Latent Variable Model for the Analysis of Financial Durations, Journal of Econometrics, **119(2)**, 381-482.

[2] Engle, R.F., and Russell, J.E. (1998a), Autoregressive Conditional Duration: a new model for irregularly spaced transaction data, Econometrica, **66**, 1127- 1162.

[3] Harris, L. (2002) Trading and Exchanges: Market Microstrucuture for Practitioners, *Oxford University Press*

[4] Kitagawa, G. (1998) A Self-Organizing State-Space Model, *Journal of the American Statistical Association*, **93**, 443.

[5] Lillo, F. and Farmer, D. (2004) The Long Memory of the Efficient Market, *Studies in Nonlinear Dynamics & Econometrics*, **8(3)**, 1.

[6] Ng, K. H, Allen, D. E, and Peiris, S. (2009) Fitting Weibull ACD Models to High Frequency Transactions Data: A Semi-parametric Approach based on Estimating Functions, *Working paper of the School of Accounting, Finance and Economics*, Edith Cowan University.

[7] O'Hara, M. (1995) Market Microstructure Theory, Blackwall.

[8] Vuorenmaa, T. A. (2011) A q-Weibull Autoregressive Conditional Duration Model with an Application to NYSE and HSE Data, *SSRN Working Paper Series.*

[9] Yamamoto, R. (2010) Order Aggressiveness, Pre-Trade Transparency, and Long Memory in an Order-Driven Market, *Journal of Economic Dynamics and Control*, **35**, 1938-1963.

[10] Zuccolotto, P. (2004), Forecasting tick-by-tick price movements, *Statistica & Applicazioni*, **II**, 1.

 ISBN: 978-1-61208-423-7