Journal of Business Thought, *Vol 10, DOI: 10.18311/jbt/2019/23355, April 2019 - March 2020*
ISSN (Print) : 2231-1734
ISSN (Online) : 2581-8104

The Role of Precision Timing in Stock Market Price Discovery when Trading through Distributed Ledgers

Daniel Broby[1*], Devraj Basu[1] and Ashwin Arulselvan[2]

[1]Accounting and Finance, Strathclyde Business School, Strathclyde University, 199 Cathederal Street, Glasgow G4 0QU, Scotland, United Kingdom; daniel.broby@strath.ac.uk, devraj.basu@strath.ac.uk
[2]Management Science, Strathclyde Business School, Strathclyde University, 199 Cathederal Street, Glasgow G4 0QU, Scotland, United Kingdom; ashwin.arulselvan@strath.ac.uk

Abstract

This paper investigates the importance of "time of execution" and the relevance of "precision time" in order driven transactions done over distributed ledgers. We created a distributed marketplace using stock market price data from the Toronto Stock Exchange (TMX). We then proceeded to test and measure the impact of timing of orders at the nanosecond level. Whilst price discovery in order driven markets is done instantaneously, with distributed markets, it is necessary to know which order to process first to avoid "front-running". We argue that a protocol for the time of order of receipt and execution should be subject to nanosecond stacking. Our approach incorporates both transitory and permanent price discovery components. It allows for the efficient processing of transactions and the order that are received by a market clearing distributed ledger.

Keywords: Clearing, Coordinated Universal Time, Distributed Ledger, FinTech, High Frequency Trading, Precision Time, Price Discovery, Settlement, Stock Market, Timestamping

JEL classification: C55, D40, G18, L10, O39

1. Introduction

The research builds on the literature of "market microstructure" and "price discovery" in finance and that of "distributed ledgers" in Electrical Engineering (FinTech being a cross disciplinary subject). The central tenent of the theory of market microstructure, as explained by Madhavan[11] is that stock prices do not always equal full-information expectations of their value because of frictions in the marketplace. In this respect, Hasbrouck[9] showed that it is important to determine when price discovery occurs in homogeneous trading in multiple markets, similar in nature to that observed in distributed markets. According to Mainelli and Smith[12], mutual distributed ledgers have the potential to transform financial transactions. Our aim is to show how, in such a transformed world, securities orders need to be processed when sent to distributed ledgers rather than a central stock exchange.

Front-running, according to Bernhardt and Taub[3], is where a market participant uses his advance knowledge of future liquidity to trade in order to generate profits. Where such trading exists, it introduces positive serial correlation to order flow which is undesirable. In order to establish a trading protocol to clear orders sent over the internet and avoid this, we established a Distributed Ledger Test-Bed (DLTB) demonstrator. In such a way, we see how price discovery works in distributed marketplaces and suggest how to avoid positive serial correlation.

Our DLTB was used to test high frequency trades and price discovery using block chain like trades sent through the configured hardware. Our programmed digital instructions mimicked what we believe will be the way certain assets will be traded over the internet in the future. We believe such research is important because, if financial markets adopt such technology, new trading protocols will have to be devised for distributed trading ecosystems.

The technology we utilize is at the core of what has come to be termed FinTech (Financial Technology). It gives market counterparties the ability to transact and store financial assets in a shared database (a ledger). Such

**Author for correspondence*

distributed databases, described from the perspective of securities trading in Andrea and Ruttenberg[15] are capable of operating without a central validation system. Current research in finance focuses solely on stock market clearing which has a central marketplace and validation system. The migration of such processes to the internet will require some of the literature to be extended. In order to do this, we aim to demonstrate that authenticated timing is essential to the function and operation of a distributed ledger in such marketplaces.

Our contribution is to extend the literature on Efficient Markets to incorporate the concept of time and the concept of information being discounted "instantly" in securities pricing. In this respect, we divide our test sample between liquidity traders (akin to High Frequency Traders) and market participants. We make recommendations on the micro time windows required for execution based on the findings between the time of order arrival and execution of orders, building on Menkveld, Koopman and Lucas[14] findings as relates to price changes at different micro time intervals.

In developing our own DLTB environment, we help extend the literature on price discovery and high frequency trading by practical experimentation. We create timestamped stock trades with Coordinated Universal Time (UTC) generated from atomic clocks and recorded on a distributed ledger that has the capacity to process more than the 2.5 million transactions from the hour of trading data that we analyse.

2. Background

Pinna and Ruttenberg[15] explain the role of distributed ledgers in securities trading. They show how these evolved from the methodology used to establish cryptocurrencies, allowing users to send trades to shared databases without using a central validation system. The dematerialisation of securities markets has shifted financial infrastructure from physical to digital validation. They identify the problem faced with a full roll out of distributed marketplaces, namely that the lack of central validation presents participants with conflicting objectives. Massacci et al.[13] identified the key security properties of a Distributed Futures Market Exchange, namely the confidentiality of positions and the absence of price discrimination.

The ability of distributed ledgers to fulfil this function depends on their technical configuration, particularly as relates the timestamping of the transactions. As the transactions are sent to multiple nodes, the first step in reaching consensus on a valid trade has to be a consensus on the transactions to be processed. We argue that this can only be achieved by timestamping and batching the orders. This element of validation is also, as explained by Broby and Paul[4], important for financial audits.

Existing financial market "clock synchronization and timestamp requirements" mandate that both trading venues and market participants synchronize their clocks to Coordinated Universal Time. The latter is a timestamp that cannot be corrupted. That said, different processing speeds, server capabilities and execution code can result in digitally programmed orders arriving at a marketplace at different times and indeed being incorrectly timestamped by inaccurate servers that do not have access to true UTC time feeds.

In order to evaluate the price impact of these factors, we conducted a high frequency order-book trial. Using nanosecond high frequency data from the TMX, we generated timestamped digital orders of varying programming length, written to execute a series of purchase and sell instructions. These were either stamped with UTC, using an atomic clock or with time that is randomly generated. The programming code orders were then sent to a central clearing house also operating on UTC and the pattern of cleared orders were then analysed.

The results will provide insights into the need for precision timestamping in financial transactions, preferably at the microsecond level. The conclusions will prove useful for regulatory and financial market participants. We also believe our results will provide a benchmark to incorporate the concept of timing into financial asset price discovery and have implications for the timestamping of block chains.

2.1 Literature

From the literature on market microstructure, we draw on Goodhart and O'Hara[6] for effects of market structure on the methodological issues relating to the treatment of time, the effects of intra-day seasonal and the effects of time-varying volatility and the information content of various market data.

An example of the price discovery literature, as relates to our paper, can be found in Broogaard et al.[5] published in the Review of Financial Studies. They examined the role of high frequency trades in price discovery and price efficiency. They extended the "State space model" proposed by Menkveld, Koopman and Lucas[14] and showed how to decompose price movements at the second by second level into permanent and temporary components. We build on their findings on such "near instant" price discovery and the correlation of returns that relate to the price changes. We do this in order to see how a distributed ledger should timestamp incoming orders. We use our findings to propose a standard for trade order timestamping in order to maintain consistency with Efficient Market Theory that suggests that prices should "instantly" reflect all available information. Note that there is a finite distinction between "near instant" and "instant".

A number of other papers, such as Hagstromer and Norden[8] focus on the role of informational liquidity at such micro time points. Hendershott and Menkveld[10] for example, investigated how price deviates from fundamental value with the arrival in a marketplace of asynchronous information. The literature points to the fact that the information and transaction time used in such trading typically occurs within a four second window.

On the Distributed Ledger side, we build on a paper by Bayer, Haber and Stornetta[1] who propose to merge many "unnoteworthy timestamping events into one noteworthy event, using a tournament run by its participants". In other words, we use micro time windows to regroup orders prior to execution to avoid them being done in the order they were received rather than the order they were timestamped. We also build on Benaloh and de Mare[2] whose timestamping protocols run in fixed time periods as rounds (windows), which they argue allow a participant to efficiently demonstrate to any challenger the round (window) in which it was timestamped.

2.2 Hypothesis

We first test the null hypothesis

H_0: The inclusion of a traceable, high precision timestamp provides no benefit to distributed ledger architecture.

In order to test H_0, we will examine the coefficient of variation of the time required to write the transactions to a distributed ledger. A low coefficient of variation indicates that the distributed ledger architecture requires a high precision timestamp to distinguish between different orders.

We also test two further hypotheses

H_1: The inclusion of a traceable precise timestamp is essential to maintaining the integrity of distributed ledger architecture, regardless of end user application. Precise timing, if delivered via dark fibre may also provide benefits in terms of resilience to GPS denial, security, confidence and the ability to provide post-event forensic analyses.

H_2: In windows of execution time, the expected "transitory" return of orders will be maximized at Xt seconds.

The H_2 hypothesis investigates the effect of having windows of different length on clearing prices in a distributed ledger market. Longer windows by definition allow for greater levels of information to be incorporated. Segmenting the data in this way will allow us to gauge the impact of short term predictability as documented by Broogard et al.[5] for the NASDAQ. We, therefore tested 1. whether longer clearing windows negatively impact liquidity traders because of the information effects and 2. whether the predictability effects could confer a trading advantage.

3. Research Design

In order to prove that an order was executed with all sufficient steps, best execution requires a measurement methodology. Current regulatory guidance suggests that trades need to be recorded in microseconds. The quantitative research on order routing by Hagstromer and Norden[8] shows that speed matters. Millisecond and sometimes microsecond timestamps are critical in the evaluation of order routing for both the sell side and the buy side. In making a recommendation on an execution time window for stock market venues, we gain inspiration from the state space model of Menkveld, Koopman and Lucas[14] that suggests a stock's price can be decomposed into a permanent component and a transitory component.

The hypotheses were tested using financial market data. This was used to understand whether orders executed at time points with the most accuracy (Coordinated Universal Time) have better price discovery than those that are not. We refer to Haber and Stornetta[7] for the method on how to timestamp a digital order.

Financial market clock synchronization and time-stamp requirements mandate that both trading venues and market participants synchronize their clocks to Coordinated Universal Time. At present the RTS prescribe different timestamp granularities for venues depending on their processing speed as well as dependent upon the type of activity engaged in. We believe our results could be used to standardise such regulations.

We take market test data gathered over one hour for the whole of the Toronto Stock Exchange for level one and simulate data for level two in the following manner. A copy of the level one data was stored with the central server and this copy was used to match the level one orders coming in as explained below.

By using financial market data, we aimed to establish the importance of the timestamp to distributed ledger technology in a real world scenario. The null hypothesis, as previously stated is that there is no trading advantage, as measured by trades' timestamped at UTC. To test this hypothesis, the trades generated from the test data order books will be sent to two separate servers.

The first server had the National Physical Laboratory's (NPL) time signal implemented while the second worked on a random server time system. The random timing mechanism for each transaction consisted of adding a random time drawn from an exponential distribution to the NPL timestamp of that transaction. The intensity parameter of the exponential distribution for the stock is the average inter arrival time for that stock, that is one (number of transactions). This setup was designed to mimic the actions of an uninformed high frequency trader who is able to view the order flow and attempts to "free-ride" the order book which is assumed to be generated by informed traders who have incurred a cost in establishing their transaction price.

One objective of our experiment was to estimate the price impact that the presence of such an uninformed free-rider could have on an informed trader in the presence of precision timing. The trade data was written on two separate ledgers, one for each server and each was to be timestamped with the corresponding timing mechanism. Each of these orders was sent to the central server where the level two order book was hosted. The level two order book was constructed and tested at arrival time and at execution time, using a range of time windows similar to those identified in Menkveld, Koopman, and Lucas[14] for high frequency trading.

The Level two order book was NPL timestamped and we sought to clear the NPL timestamped trade first. The data was analysed to see the difference between the clearing prices for the two sets of trades. Our experiment compiled the Level one data from the Level two order book and this order differently timestamped, was sent to the second server. In the case where identical orders from two servers arrived at the Level two order book within a batching window, each order was given equal priority, that is half of the volume for each order was cleared at that price. The rest of the order was cleared at the next available price. The price impact for the NPL timestamped order is thus 1/2* (Next Price - Original Price)/Original Price.

3.1 Distributed Ledger Set Up

In our DLTB, we set the distributed ledger up in order to accommodate the timestamping. NPL provided us with an NPL time signal to our transmitters at our various server racks at three separate locations using their internet connectivity. We used three data sources, (i.e., 3 transmitters) with NPL timestamped data and three sources that were un-stamped. Data was provided by TMX and nanosecond market data. NPL Time was then stamped and local system timestamped data sent via transmitters. Using three servers running in parallel we then examined any failures and mechanisms that we needed to mitigate, including the time-cards.

We programmed the hardware to generate a unique ID and then use the NPL time signal to timestamp to one nanosecond. The processing time in the order coding, its generation time and when it arrived at the distributed ledger were both variables that would result in differences in execution against the level two order book. Then a program took the order instructions received at the distributed ledger (anywhere up to 1000 instructions/sec) and extracted from the level two data, the resulting volume weighted executed price. The programme code was written in various length formats so as to ensure different processing times.

3.2 Data Description

Market data provision is provided by TMX (Toronto). The Flat-file database we construct from it has three hours of level one data at the nanosecond frequency.

The data consists of Alpha Level one TMX Quantum Feed:

- Trades
- Quotes
- Symbol and Stock Status

The data is in effect a test data connection between our servers and TSX's market information system. The trading in the Toronto market is order driven and as such we do not make a market with bid-ask spread, simply a record of the available liquidity.

There were 1.28 million buy transactions and 1.30 million sell transactions in our data set with the buy transactions occurring for 2140 stocks with the number of transactions ranging from 1 to 20736 in this one hour period and sell transaction for 2123 stocks ranging from 1 to 22968 transactions.

4. Results

The statistics of the transaction time, which is the time required to write the transaction to a distributed ledger is shown in Table 1. The average transaction time is 1.24 milliseconds for both the buy and sell transactions and the standard deviation for the buy transactions is .042 and that for the sell transactions it is .047. Thus, the coefficient of variation measured as standard deviations in units of mean is .034 or 3.4 percent of mean for the buy transactions and .038 or 3.8 percent for the sell transactions, both of which are quite low. The transaction time deciles are also shown in Table 1 and the decile coefficient of variation for which we use the 3rd and 7th deciles is .0070 for the buy transactions and is .0056 for the sell transactions, so both less than one percent. The lower decile based coefficient of variations shows that for the majority of transactions, excluding outliers, the level of variation in transaction writing time is extremely low, less than one percent of the average transaction time. As the average transaction time is in microseconds (10^{-3} seconds) discrimination at the level of milliseconds (10^{-6} seconds) is required to differentiate between transactions. Hence, Table 1 appears to suggest that a high precision timestamp is required and provides benefit to distributed ledger architecture. Thus, we find little or no support for our first hypothesis H_0.

Table 1. Transaction Time Details

Buy Transactions (in Milliseconds)	
Average Transaction Time	1.241157832
Standard Deviation	0.042861926
Minimum Transaction Time	1.149525
Maximum Transaction Time	2.27233425
Transaction Time Deciles	
1	1.213001186
2	1.224037209
3	1.231090885
4	1.234937147
5	1.237730926
6	1.240347235
7	1.243460403
8	1.248182807
9	1.260513446
10	2.27233425
Sell Transactions (in Milliseconds)	
Average Transaction Time	1.241496517
Standard Deviation	0.047737234
Minimum Transaction Time	1.153183
Maximum Transaction Time	1.9697225
Transaction Time Deciles	
1	1.21227835
2	1.223527828
3	1.230132303
4	1.234666238
5	1.237925604
6	1.240914765
7	1.243963517
8	1.24965368
9	1.265761161
10	1.9697225

Source: Based on Authors' own calculations.

Note: This Table shows the details for the transaction times, which is the time taken for the central server to write the transaction to a distributed ledger, for the buy and sell transactions in the data set. There were 1.27 million buy transactions and 1.3 million sell transactions in our data set.

In order to analyse our next hypothesis H_1, we consider the results of the timing experiment which are shown in Table 2 (for buy orders) and Table 3 (for sell orders). For each batching window (1%, 2%, 5%, 10%, 25% and 50%) of average transaction time, we first analyse the percentage of uniformed "free-rider" transactions that fall in each batching window. This is done across stock deciles sorted on the number of transactions. The average number of transactions varies from three in the bottom decile to just below 2700 in the top decile. There is a clear increase in the percentage of "free-rider" transactions from lower to higher deciles across all batching windows. This is to be expected as the intensity parameter increases as the inter-arrival time is lower, leading to a larger number of lower draws from the exponential distribution.

Table 2. **Transaction Batch Frequency – Buy Orders**

		Factor1.01		Factor1.02		Factor1.05	
Decile	Transactions	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact
1	3	0.00%	0.00%	33.33%	0.03%	33.33%	0.66%
2	8	50.00%	0.69%	50.00%	0.89%	62.50%	1.12%
3	16	43.75%	0.51%	50.00%	0.56%	68.75%	0.77%
4	31	48.39%	0.15%	54.84%	0.22%	74.19%	0.32%
5	56	49.56%	0.13%	56.64%	0.15%	74.34%	0.21%
6	125	48.00%	0.09%	55.60%	0.11%	72.80%	0.14%
7	276	51.63%	0.07%	59.06%	0.08%	77.54%	0.10%
8	496	51.21%	0.04%	58.67%	0.05%	76.11%	0.07%
9	943	51.35%	0.03%	59.09%	0.04%	77.53%	0.05%
10	2686	52.02%	0.04%	59.69%	0.04%	78.26%	0.06%
		Factor1.1		Factor1.25		Factor1.5	
Decile	Transactions	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact
1	3	33.33%	1.39%	66.67%	1.63%	66.67%	1.63%
2	8	75.00%	1.37%	87.50%	1.58%	87.50%	1.59%
3	16	75.00%	0.85%	87.50%	0.91%	87.50%	0.98%
4	31	80.65%	0.37%	90.32%	0.39%	93.55%	0.40%
5	56	81.42%	0.26%	89.38%	0.28%	92.04%	0.28%
6	125	83.20%	0.16%	89.60%	0.17%	92.00%	0.17%
7	276	87.68%	0.12%	94.57%	0.13%	96.56%	0.13%
8	496	86.59%	0.07%	92.84%	0.08%	95.97%	0.08%
9	943	87.97%	0.06%	94.22%	0.07%	97.56%	0.07%
10	2686	88.80%	0.06%	95.81%	0.07%	97.97%	0.07%

Source: Based on Authors' own calculations.

Note: In this table all stocks for which there are transactions are sorted into deciles based on the number of transactions. The second column shows the average number of transactions per decile. For each batching frequency (1%, 2%, 5%, 10%, 25% and 50% of the average transaction buy time shown in Table 1) we compute the frequency of overlaps between informed and uniformed buy transactions and their price impact. A full description of overlaps and price impact is given in Section 3.

The variation across deciles varies from 50 percent for the smallest batching window, ranging from 0 percent for the lowest decile to 50 percent for the top decile to around 30 percent for the largest batching window (from 67 percent to 97 percent across the deciles). For the most heavily traded stock decile, there is an overlap of just over 50 percent for the smallest batching window rising to 99 percent for the largest batching window. This suggests that very precise price discrimination is required in order to be able to distinguish between informed and "free-riding" transactions even when the batching window is very small, of the order of microseconds. Hence, we find support for our second hypothesis H_1 in that a precise timestamp will be required in our setting to maintain the integrity of the clearing mechanism. Thus, our conclusion is that the orders will have to be regrouped by UTC rather than orders received and executed. This is relevant to High Frequency Trading. It is also extremely important for the emerging financial block chains that have to be timestamped. At present, these timestamps are ordinal and as such potentially vulnerable.

The third hypothesis H_2 relates to the choice of batching window and here the price impact results from Tables 2 and 3 provide some insights. For all batching windows beyond 10 percent there is a clear link between price impact and liquidity with price impact being higher

Table 3. **Transaction Batch Frequency - Sell Orders**

		Factor1.01		Factor1.02		Factor1.05	
Decile	Transactions	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact
1	2	0.00%	0.00%	0.00%	0.00%	50.00%	0.25%
2	7	42.86%	0.60%	42.86%	0.75%	57.14%	0.99%
3	14	50.00%	0.41%	57.14%	0.48%	71.43%	0.64%
4	29	44.83%	0.15%	51.72%	0.17%	68.97%	0.22%
5	54	46.30%	0.09%	54.63%	0.10%	70.37%	0.13%
6	113	47.35%	0.07%	55.75%	0.08%	75.22%	0.11%
7	256	50.88%	0.06%	59.26%	0.07%	76.41%	0.09%
8	515	51.02%	0.04%	58.58%	0.05%	77.50%	0.06%
9	936	52.16%	0.03%	59.69%	0.04%	77.52%	0.05%
10	2648	52.65%	0.03%	60.52%	0.04%	79.42%	0.05%
		Factor1.1		Factor1.25		Factor1.5	
Decile	Transactions	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact	% of Overlaps	% Price Impact
1	3	50.00%	0.59%	50.00%	0.73%	50.00%	0.78%
2	8	71.43%	1.15%	85.71%	1.31%	85.71%	1.33%
3	16	78.57%	0.72%	85.71%	0.80%	85.71%	0.82%
4	31	79.31%	0.29%	86.21%	0.32%	89.66%	0.32%
5	56	81.48%	0.15%	88.89%	0.16%	91.67%	0.16%
6	125	83.63%	0.13%	89.82%	0.14%	92.48%	0.14%
7	276	86.16%	0.10%	93.18%	0.11%	95.52%	0.11%
8	496	87.68%	0.07%	93.99%	0.08%	96.41%	0.08%
9	943	88.41%	0.06%	95.57%	0.06%	97.92%	0.06%
10	2686	89.94%	0.05%	96.53%	0.06%	98.96%	0.06%

Source: Based on Authors' own calculations.

Note: In this table all stocks for which there are transactions are sorted into deciles based on the number of transactions. The second column shows the average number of transactions per decile. For each batching frequency (1%, 2%, 5%, 10%, 25% and 50% of the average transaction sell time shown in Table 1) we compute the frequency of overlaps between informed and uniformed sell transactions and their price impact. A full description of overlaps and price impact is given in Section 4.

for stocks with lower liquidity. From the 10 percent batching window onwards the price impact ranges from around 1.5 percent for the least liquid deciles to less than 0.1 percent for the most liquid deciles. Thus, for our data, which seems a quite typical dataset, the high number of overlap transactions for highly liquid stocks do not seem to translate into much price impact possibly due to the fact that prices of consecutive transactions are quite close together. For the less liquid stocks, the lower proportion of overlaps is somewhat offset by the potentially greater price difference in consecutive transactions. Thus, we are led to the conclusion that smaller batching windows are better in lowering the price impact for less liquid stocks and that a batching interval of upto 110 percent of average transaction or writing time works best in our setting.

5. Conclusion

This research demonstrated a proof of concept in the use of precision timing for distributed ledgers in the context of stock market clearing. We find support for the hypotheses that precision timing is required to maintain the integrity of distributed ledgers in clearing. The role of distributed ledgers and precision timing is becoming ever more relevant as FinTech companies adopt block chain as a financial transmission tool. Should stock market trading migrate from a central marketplace to a distributed ledger marketplace; our protocols will have real world application.

We propose a mechanism for execution of blockchain orders in a distributed ledger marketplace that will reduce the potential for flash crashes based on these results. We also hope this will lead to future research that will create the conditions for a "level playing field" for those with different technological capabilities in order routing and delivery speeds to the distributed ledgers.

We believe our research brings important insights from the finance fields of microstructure and price discovery that will prove relevant for future distributed ledgers and blockchain marketplace development. We believe the size and scale of our study, the use of an atomic clock, a test bed distributed ledger and both level one and level two stock market data, all add to the credibility of our findings.

6. References

1. Bayer D, Haber S, Scott Stornetta W. Improving the efficiency and reliability of digital timestamping. Sequences II: Methods in Communication, Security and Computer Science. 1993:329–34.
2. Benaloh J, De Mare M. Efficient broadcast time-stamping. Clarkson University: Department of Mathematics and Computer Science TR; 1991. p. 91–101.
3. Bernhardt D, Taub B. Front-running dynamics. JET. 2008; 138(1):288–96. https://doi.org/10.1016/j.jet.2007.05.005
4. Broby D, Paul G. The financial auditing of distributed ledgers, blockchain and cryptocurrencies. Journal of Financial Transformation. 2017; 46:76–88.
5. Brogaard J, Hendershott T, Riordan R. High-frequency trading and price discovery. Rev Financ Stud. 2014; 27(8):2267–306. https://doi.org/10.1093/rfs/hhu032
6. Goodhart CAE, O'Hara M. High frequency data in financial markets: Issues and applications. J Empir Financ. 1997; 4(2):73–114.
7. Haber S, Stornetta WS. How to timestamp a digital document. Conference on the Theory and Application of Cryptography; Berlin, Heidelberg: Springer; 1990. p. 437–55.
8. Hagstromer B, Norden L. The diversity of high-frequency traders. J Fin Markets. 2013; 16(4):741–70. https://doi.org/10.1016/j.finmar.2013.05.009
9. Hasbrouck J. One security, many markets: Determining the contributions to price discovery. JOF. 1995; 50(4):1175–99. https://doi.org/10.1111/j.1540-6261.1995.tb04054.x
10. Hendershott T, Menkveld AJ. Price pressures. J Financ Econ. 2014; 114(3):405–23. https://doi.org/10.1016/j.jfineco.2014.08.001
11. Madhavan A. Market microstructure: A survey. J Fin Markets. 2000; 3(3): 205–58. https://doi.org/10.1016/S1386-4181(00)00007-0
12. Mainelli M, Smith M. Sharing ledgers for sharing economies: An exploration of mutual distributed ledgers (aka blockchain technology). The Journal of Financial Perspectives. 2015; 3(3):38–69.
13. Massacci F, Ngo N, Nie J, Venturi D, Williams J. FuturesMEX: Secure distributed, futures market exchange. Unpublished Journal Submission. 2017.
14. Menkveld AJ, Koopman SJ, Lucas A. Modeling around-the-clock price discovery for cross-listed stocks using state space methods. J Bus Econ Stat. 2007; 25(2):213–25. https://doi.org/10.1198/073500106000000594
15. Pinna A, Ruttenberg W. Distributed ledger technologies in securities post-trading revolution or evolution? 2016.