Article Citation Format
Ogunsakin, R.; Ebono, F. & Ochei, L.C (2023): Investigating The Impact of High-Frequency Trading Latency On Market Quality: Discrete Event and Agent-Based Simulation Approach. Journal of Advances in Mathematical & Computational Sc.
Vol. 8, No. 1. Pp 125-148

Article Progression Time Stamps

Article Type: Research Article
Manuscript Received 19th Jan, 2023
Final Acceptance: 31st March, 2023

Investigating The Impact of High-Frequency Trading Latency On Market Quality: Discrete Event and Agent-Based Simulation Approach

[1]Ogunsakin, R.; [2]Ebono, F. & [3]Ochei, L.C
Department of Computer Science
University of Port Harcourt
Port Harcourt, Nigeria.
E-mail: [1]rotimi.ogunsakin@gmail.com; [2]fubara.ebono@uniport.edu.ng; [3]laud.ochei@gmail.com

ABSTRACT

This research evaluates the impact of High-Frequency Trading (HFT) latency on market quality, which are liquidity, price discovery and Volatility. To achieve this, a combination of Discrete Event and Agent-Bases simulations are developed to model, investigate and explain the various impact of HFTs latency on market quality. The simulation provides an in-depth analysis of HFT latency and its resulting impacts on market quality. The result of the simulation shows that (1) HFTs volatilise security prices due to their latency dominance and later reverse their strategy to push prices back to their efficient level by trading in opposite direction and, thus, profiting from the bid/ask spread. (2) Latency between HFTs and the exchange (for order submission) is most paramount compared to Market Data-feed latency when investigating latency impact on market quality, and (3) The impact of the data-feed latency is recoverable given that (i) the latency-lag is very small - approximately 5ms (ii) provided that the specific order under consideration is large enough to last for a duration proportional to the data-feed latency-lag (4) The impact of latency-lag for order submission is not recoverable within a trading window. Finally, we conclude that HFTs latency positively impacts liquidity; contributes positively to price efficiency for the fast HFTs, while the slow HFTs incurred a latency cost for pricing information inefficiencies. However, HFTs' contribution to volatility depends on which side of the market we analyse. For example, considering the opening and closing of a security price, it was observed that HFT positively impact volatility. But looking at the intermediaries, that is, in between the opening and closing price, volatility is introduced and cleared before the end of the trading window.

Keywords: High-Frequency Trading, Liquidity model, Financial model, Discrete Event Simulation, Agent Simulation.

1. INTRODUCTION

HFT refers to trading strategies characterised by reliance on very fast access to trading platforms and market information, through the use of computers and other information technology devices [43]. HFTs are part of a broader group of algorithmic trading, where the use of computer programs are employed to implement investment decisions and trading strategies.

High-Frequency Trading (HFT) relies on technological innovation in communication, microprocessor design, and the ability to manage complex algorithms more efficiently. HFT employs different trading strategies, such as electronic liquidity provisioning, statistical arbitrage, liquidity detection and directional strategies. According to the United States Security and Exchange Commission (SEC), High-Frequency Traders (HFTs) are proprietary trading firms that use high-speed computer systems to access market data, submit a large number of orders, and use algorithmic means to maximise their competitiveness and profit [3]. Other characteristics attributed to HFTs include (1)The use of high-speed and sophisticated computer programs to generate, route, and execute orders; (2) The use of co-location services and individual data feeds offered by exchanges and other service providers to minimise network and other types of latencies (like processing speed); (3) Very short time-frames for establishing and liquidating positions; (4) Submission of numerous orders that are cancelled shortly after submission; and (5) Ending a typical trading day in as close to a flat position as possible (that is, not carrying significant, unhedged positions overnight) [3].

Research on HFTs' impact on market quality has used empirical means by analysing trading data and mathematical models to investigate HFTs' impact on market quality [5, 7, 8, 10]. There are also research works that have examined the impact of latency on market quality and HFT Latency [9, 12, 13, 14]. However, to the best of our knowledge, none of the research on the effect of latency on market quality has explored a simulation approach to examine the impact of latency on market quality using different scenarios and varied latency lag between Fast HFTs and Slow HFTs or between HFTs and non-HFTs to gain insight into the fierce latency competition taking place among the HFTs and their corresponding impact on equity market quality.

Consequently, this research proposes and develops a simulation model to simulate the interaction of a minimum of two HFTs with an exchange using Discrete Event Simulation (DES) and Agent-Based Simulation (ABS). The ABS is used to model the interaction of the different HFTs with the order book. These HFTs are configured to trade using the "same strategy" based on the principle of "Zero Intelligence" [15, 16], but with varied latency to the exchange. Since our interest is in examining the effect of latency on market quality, it is natural to keep the strategy constant by allowing all market participants (HFTs and non-HFTs) to trade with the same strategy and vary the parameter we wish to evaluate (Latency). The result of each simulation epoch is visualised and analysed to evaluate the impact of latency on liquidity, price discovery and volatility.

The research aim to analyse and evaluate the impact of HFT latency on equity market quality - liquidity, price discovery and volatility, using DES and ABS. Consequently, identify and understand the different HFT strategies and their underlying mathematical, statistical, quantitative and qualitative models. The developed DES to model the interaction of multiple traders (HFTs with varied latency) with an Exchange, then the ABS to model the interaction of the various traders (HFTs and non-HFTs) with the order book. Finally, collect, analyse and visualise data during the simulation run to examine the impact of latency variation (high and low) on market quality.

The remainder of the paper is structured as follows: In Section 2 contains a review of the literature on the evolution of HFT, the different HFT strategies and their impact on equity market quality, the impact of HFT latency variation on market quality, and the application of simulation in the domain of financial intelligence. Section 3 contains the research design and methodology. Section 4 presents the development and implementation of the simulation model. In Section 5, the simulation model is evaluated using different scenarios, and the results of these different scenarios are analysed and discussed. This research work is concluded in Section 6, which also include a recommendation for further work.

2. LITERATURE REVIEW

2.1 High-Frequency Trading And Algorithmic Trading

Algorithmic trading (AT) is the submission of orders to the exchange by computers that are directly connected to trading platforms without immediate human participation. These computers have sophisticated algorithms built into them that enable them to make trading decisions at extremely high rates, frequently in milliseconds [18], based on the outcome of observed previous market data and other information. HFT is a subset of algorithmic trading (AT), and it shares many of AT's traits, such as the use of complex algorithms to make instantaneous trading decisions.

There are some traits that are unique to AT and not frequently connected to HFT. In order to minimise the impact of big orders relative to a predefined benchmark, the focus is typically on intelligently working orders over time and across markets[19]. As a result, HFT strategies are by nature focused on a highly liquid instrument, and as a prerequisite, HFTs depend on high-speed (low latency) market access, which is made possible by significant investment in high-speed communication linked to the exchange, use of co-location/proximity service, and dedicated/individual data feed. The Table in Figure 2.3 displays traits unique to HFT that are typically not connected to AT.

2.2 High-Frequency Trading (HFT) Strategies

There are well-known strategies, the majority of which were in use before the invention of HFT but were made more efficient through the use of high-speed computing infrastructures, communication networks, and sophisticated algorithms, making it difficult to be able to examine all strategies due to the diversity and opaqueness of the HFT universe. The list of some of the most well-known HFT strategies, as found in the research area, is shown in Figure 2.4 below.

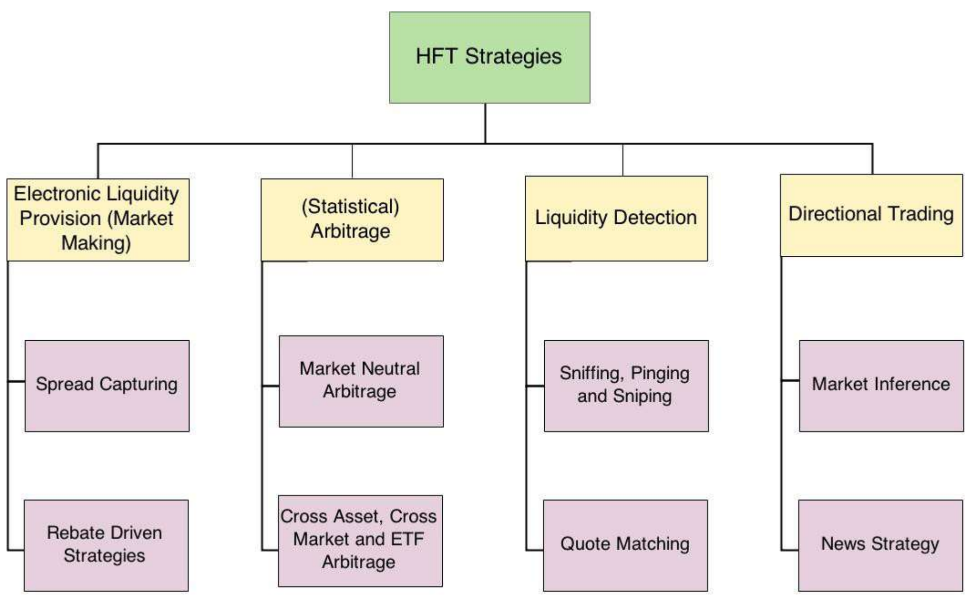


Figure 2.4: Common High Frequency Based Strategies - Adapted from Gomber et al (2011) [19]

Electronic Liquidity Provision (Market Making)
n order to take advantage of the bid-ask spread and concurrently give market participants liquidity, a market maker (in this example, an HFT Trader) puts simultaneous buy and sell limit orders for a financial instrument on both sides of the electronic order book [6, 19]. HFTs who provide liquidity are given rebates as compensation for the risk involved in liquidity provisioning, which encourages the provision of liquidity in the equity market and ensures that all market orders are converted to trade at the market bid-ask price at any given time during the trading hours.

(Statistical) Arbitrage
Two closely linked financial products, such as the S&P 500 characteristics and SPY, whose prices should move in tandem with one another are used in statistical arbitrage (the ticker symbol that tracks S&P 500). An HFT can purchase the low-priced SPY and sell the high-priced S&P 500 in order to profit from the bid-ask spread if the price of the S&P 500 increases as a result of the entry of a buy order but the SPY does not increase right away [6]. The market offers this sort of opportunity for a relatively little time (in microseconds or milliseconds), hence the fastest HFTs will always prevail. There are other types of arbitrage as well, and they often profit from pricing inefficiencies in an asset or market. [19].

Liquidity Detection
In detecting liquidity, HFTs use sophisticated algorithms to identify patterns left on the market by other market players and modify their actions (to buy or sell) in accordance. The majority of the time, HFTs concentrate on large orders and use a number of techniques to identify sliced orders or learn more about electronic order books. This is frequently referred to as "sniffing out" other algorithms or "Pinging" in order books to extract information [19].].

Directional Trading
HFTs uses directional trading method, placing orders in response to order flow cues. This can also take the shape of news, which is automatically processed using text analytics and traded (i.e., bought or sold) depending on information deduced from the news [6].

2.3 Impact of HFT Strategies On Market Quality
HFTs trade against market price movement, according to [20], who analysed the contributions of HFT and non-HFT liquidity supply and demand to price change components. As prices diverge from fundamental value, HFTs start a transaction in the opposite direction to bring prices back to their efficient level, boosting price effectiveness and liquidity. As a result, when some HFT gets exceedingly fast, it increases the adverse selection cost for the slower traders and generates negative externalities, according to [21], who looked at the trading equilibrium for a specific degree of HFT. As a consequence of the speedier HFTs' ability to process bid-ask quotation information and adjust their trading strategies before the slower HFTs, this is most likely a result of the slower HFTs being unfairly picked.

The majority of the research on HFT tactics showed that these methods increased liquidity in the stock market while simultaneously creating an unfavourable selection for slower HFTs, leading to negative externalities. Although it is assumed that the improvement in liquidity and the decrease in volatility are connected, [11]'s observation shows that they are not. Several empirical data, such as [22], which show that HFT increases volatility while improving liquidity andprice discovery, supported [11]'s statement. Nonetheless, generally, the bulk of the research agreed that HFTs increase volatility while also stabilising the market's bid-ask quote in the face of considerable price imbalance. The study literature on the influence of HFT on volatility appears to be inconsistently consistent.

The speed of light is believed to be the maximum speed achievable in the trading industry, however this is practically unattainable given the inherent physical constraints imposed by the components required to transfer market data from an exchange to a trader and back again. The connecting medium still presents a challenge to obtaining this speed of light, even in the situation of co-location, when HFTs systems are connected directly to the exchange. Another obstacle is the hardware's clock speed, which determines how quickly trading algorithms are run.

The trading world, where trading decisions are made and carried out in milliseconds, has no place for human traders despite these inherent obstacles to reaching the speed of light [23]. Latency The phrase "the time it takes to access, analyse, and react to market information [24]" is a relative one since what was ultra-low latency yesterday can be low latency today. Low latency, however, might be defined as round-trip times of under ten (single digit) milliseconds [23]. The exchange market has two types of latency: exchange latency and market participant latency, which includes human, non-HFT, and proprietary HFT traders. These two types of delay have a big influence on the calibre of the market.

In the financial market, the phrase "race to zero" [25] has gained currency and refers to the lack of latency (latency = 0). The market players who use market-making methods (Liquidity providers), to whom the bulk of HFTs belong, are where the rivalry for zero latency is particularly pronounced. These HFTs make money by capturing the difference between supply and demand for certain securities.

The latency arms race also has an impact on agency execution providers. Because the SOR (Smart Order Router) used to determine order routes depends on real-time market data, every purchase or sell decision that is made later than that of a rival will incur a fee. The scenario offered by [26] that reads, "Considering latency from the standpoint of a liquidity provider, if the presence of an observable news in the public domain leads his quote to become stale," is the ideal one for this situation. There instantly starts a race where 1) People are attempting to alter his stale comment, and 2) There are lots of people trying to snap his stale quote. Even with the cutting-edge speed, one will always lose to many in a continuous limit order book since messages are processed one at a time in a serial fashion [26].

It is noted in the study that has been evaluated that estimates of latency impact and cost are made using market data, which does not give an understanding of what occurs outside of the exchange. Realizing that the bulk of market players cannot execute orders because they only last a few milliseconds in most quick strategies—those involving low latency [29]. As the trading data does not include these unsuccessful efforts by comparatively slower traders, utilising trading data to analyse the impact of latency may appear biased. Nevertheless, employing a simulation technique allows us to examine what is occurring outside of the exchange by looking at and analysing the exchange queue in addition to testing various situations by altering latency and the number of market players.

2.4. High-Frequency Trading (HFT) Simulation

[24] developed a Discrete Event Simulation (DES) System to capture the fragmentation that exists in cross-market communication due to information transfer delay, which is being utilised by low-latency HFTs to execute arbitrage strategy, while Agent-Based Simulation (ABS) was used to simulate the interaction between HFT and Zero Intelligence trading agent at millisecond level to examine the effect of latency arbitrage on market quality. The simulation approach used in this research work was adopted from that of Wah and Wellman work. That is, using DES to model the interaction of multiple traders (HFTs with varied latency and non-HFTs) with an Exchange and ABS to model the interaction of the various traders (HFTs and non-HFTs) with the order book.

The concept of *Zero Intelligence,* as used by [24], was first proposed and used by [30] in an experiment at the University of Carnegie Mellon, where human traders were replaced by a Zero Intelligence pro- gram that submits a random bid and ask orders to investigate the determining factor of market allocative efficiency. The result of the experiment showed that allocative efficiency in a double auction market derives largely from its structure and is independent of traders' motivation, intelligence or learning.

[31], in their work "The predictive power of zero intelligence in financial markets", also adopted the principle of Zero Intelligence traders to test the statistical mechanism of price formation and the accumulation of stored supply and demand under the simple assumption that orders are placed at random with no underlying motivation or intelligence. The resulting model makes an excellent prediction for transaction cost, price diffusion rate and quantity closely related to supply and demand under natural market conditions, which suggests the price formation mechanism is constrained by the market rather than the strategic behaviour of agents as previously thought.

The Zero Intelligence principle was adopted in this research since the aim of the research is to investigate the effect of latency, and thus, it will be logical to keep the variable (Strategy) constant while we vary latency to test different scenarios. Using Zero Intelligence traders will help to achieve this, and also, since traders' motivation and intelligence do not have any effect on price formation and allocative efficiency of the financial market [24, 30, 31]. The resulting impact of latency on market quality will not be affected by the use of Zero Intelligence traders.

3. RESEARCH DESIGN AND METHODOLOGY

Lacking suitable data to empirically study the effect of latency on market quality, a simulation approach is pursued, which enables the incorporation of causal premises and, specifically, presumptions of how trading behaviour is shaped by environmental conditions [24]. The simulation method is divided into two logical parts which are: Discrete-Event Simulation (DES) used to model the interaction of multiple traders (HFTs with varied latency and non-HFTs) with an Exchange and Agent-Based Simulation (ABS), used to model the interaction of the various traders (HFTs and non-HFTs) with the order book.

3.1. Discrete-Event Simulation (DES)

Discrete-Event Simulation models are described at a level of abstraction where the time base is continuous, but during a bounded time span, only a finite number of relevant events can occur - These events can cause the state of the system to change, but the system's state remains unchanged in-between events [34]. A Discrete-Event System can either be Deterministic or Stochastic. Modelling a financial market require the use of both Deterministic and Stochastic System.

In a financial market, each market participant (HFTs, non-HFTs or a Human trader) generates order deterministically. Meaning every market or limit bid/ask order from each trader have a constant speed which is dependent on the latency of the interconnection between the trader and the exchange. If the departure time is known, the arrival time can be estimated.

Let the Departure time of an order from an HFT = T_d
Let the Arrival at the Exchange = T_a
Let the Latency of the link between the HFT and the Exchange = δ

Thus, the relationship between T_d, T_a and δ can be expressed as follows:

$$T_a = T_d + \delta + d(\tau) \quad (3.1)$$

Where τ is the position of the order at the exchange queue and $d(\tau)$ is the rate at which the order advances in the exchange queue. If there is no queue at the exchange at the time of order arrival or the order is the first to arrive at the exchange, then $d(\tau) = 0$, then:

$$T_a = T_d + \delta : \text{When } d(\tau) = 0 \quad (3.2)$$

Even if we assume that the Latency $\delta = 0$. The existence of a queue at the exchange means the order will not be executed at the Time $T_a = T_d$ as shown below:

$$T_a = T_d + d(\tau) : \text{When } \delta = 0 \quad (3.3)$$

It can be observed from above that even though the departure of orders from a market participant is deterministic (The speed of its connection to the exchange), the execution of the order at the exchange is stochastic and not deterministic due to the likeliness of an exchange queue and the existence of “price” competition among various traders.

Another stochasticity that also exists is the bid/ask quote attached to each order as attribute. We apply the principle of Zero Intelligence as previously elucidated. Meaning each trader generates bid/ask quote at random with no underlined intelligence or strategy, taking from a sets of Uniform Distribution. Therefore, at the time an order departs from a trader’s system, it has both deterministic and stochastic attributes attached to it, which are the speed (Latency) and quotes (Bid/Ask) respectively. As soon as the order arrives at the exchange queue, all attributes become stochastic - These is the areas we are interested in investigating. Thus, a deterministic and stochastic approach is required to successfully build a DES model of acceptable complexity and representation of a real exchange system.

3.2 Agent-Based Simulation (ABS)

Agent-Based Simulation (ABS) or Agent-Based Modelling (ABM) is a modelling approach for simulating actions and interactions of autonomous individuals, referred to as agents, with the view of assessing their effect on the system as a whole [37]. Complex systems like the financial market with thousands of market participants (ATs, HFTs, non-HFTs and Human traders) can best be understood as a system of autonomous agents that follows a set of rules of interaction with other agents and their environment - in this case, the order book [38].

The interaction of HFTs with the order book is modelled based on the principle of Agent-Based Modelling, where the agents, in this case, represent the different traders with specific attributes, which are latency, order and quotes. Each individual trader agent is autonomous and self-contained in that they act independently. Each trader generates and submits an order to the exchange at discrete intervals, which is both deterministic and stochastic, as explained in the previous section. The interaction of individual trader-agent with each other is done on the order book. A bid/ask order submitted by one trader-agent is being executed by another agent and, as a result, changes the system dynamics. Each time an agent interacts with the order book, the depth of the order book either increases or decreases, which in turn changes the probability distribution of the outcome of the next trader-agent interaction with the order book [27]. Summarily, the dynamics of the Simulation Model change from the perspective of an entity (orders) in the system at each point in the simulation depending on the entity’s state and position, as shown in figure 3.1.

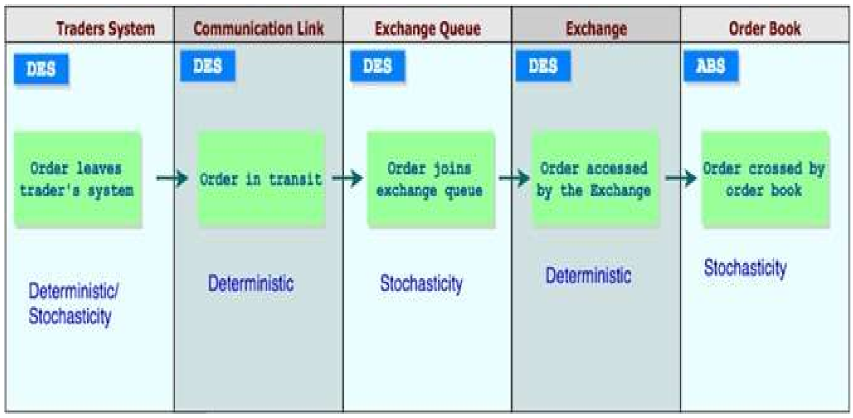


Figure 3.1: Simulation System Dynamics

3.3 Simulation Tool - Simulink and SimEvents

Simulink is a product of Mathworks, a block diagram environment for multi-domain simulation and Model-Based Design with support for simulation, automatic code generation, and continuous testing and verification of embedded systems [41]. In addition, Simulink provides a graphical editor, customisable block libraries, and solvers for modelling and simulating dynamic systems that integrate seamlessly with MATLAB, enabling the incorporation of MATLAB codes and algorithms into models and export of simulation results to MATLAB for further analysis[41]. This flexibility of being able to manipulate Simulink block libraries among other inbuilt libraries suitable for model-based design made it a tool of choice for modelling the Agent-Based Simulation (ABS) part of this research work.

SimEvents is a sub-component of Simulink which provides a discrete-event simulation engine and component library for Simulink, which can be used to model event-driven communication between components to analyse and optimise end-to-end latencies, throughput, packet loss, and other performance characteristics [36]. The inbuilt libraries of pre-defined blocks, such as queues, servers, and switches, enable the accurate presentation of the system and customise routing, processing delays, prioritisation, and other operations, which make it a choice modelling tool for the Discrete-Event Simulation (DES) part of the research work.

3.4 Summary of Simulation Assumptions

The following assumptions are made in the developed Discrete-Event Simulation (DES) and Agent-Based Simulation (ABS):

1. Order submission per discrete time interval is 1 (i.e quantity of each submit- ted order per a discrete time interval = 1).
2. Orders are exclusively processed based on “Time priority” without consideration for price “Price priority” - This is because we assume all traders uses the same strategy “Zero Intelligence” and therefore, there is no price superiority.
3. The principle of “Zero Intelligence” is used to randomly generates Bid/Ask quotes in the Simulation System to exclusively restrict the simulation to the testing impact of latency by keeping the “Strategy” variable constant.
4. The exchange’s latency is set to 1 microsecond to allow the Exchange to exhaust the milliseconds generated orders in the order queue during a simulation window of 1 second.

5. Simultaneous events that occur during entity (Order) generation are processed round-robin, i.e the respective position of such generated orders at the Exchange queue is determined by the round-robin algorithm as implemented in SimEvents. However, Order processing by the Exchange is exclusively sequential, based on the position of each order at the exchange queue using the First-In-First-Out (FIFO) standard data structure as implemented in SimEvents.
6. All through the Simulation and its evaluation, we refer to Slow HFTs, ATs and Human Traders as Slow HFTs (Abbreviated as sHFTs), While the HFTs assumed to be trading at ultra-low latency are referred to as Fast HFTs (Abbreviated as fHFTs).

4. LIQUIDITY MARKET SIMULATION AND ANALYSIS

4.1 High-Frequency Trading and Latency

The financial market is a complex system of interacting independent systems and agents, of whose collective activities determine the state of the system. Modelling the overall financial system will be overly complex for the research, but modelling only part of the financial system under consideration, which are those areas of the financial market where HFT latency have a direct or indirect influence, and with an acceptable and sufficient level of complexity to investigate the impact of HFT latency on market quality will be best. Figure 4.1 below shows the structure of the modelled financial market using Simulink and SimEvents.

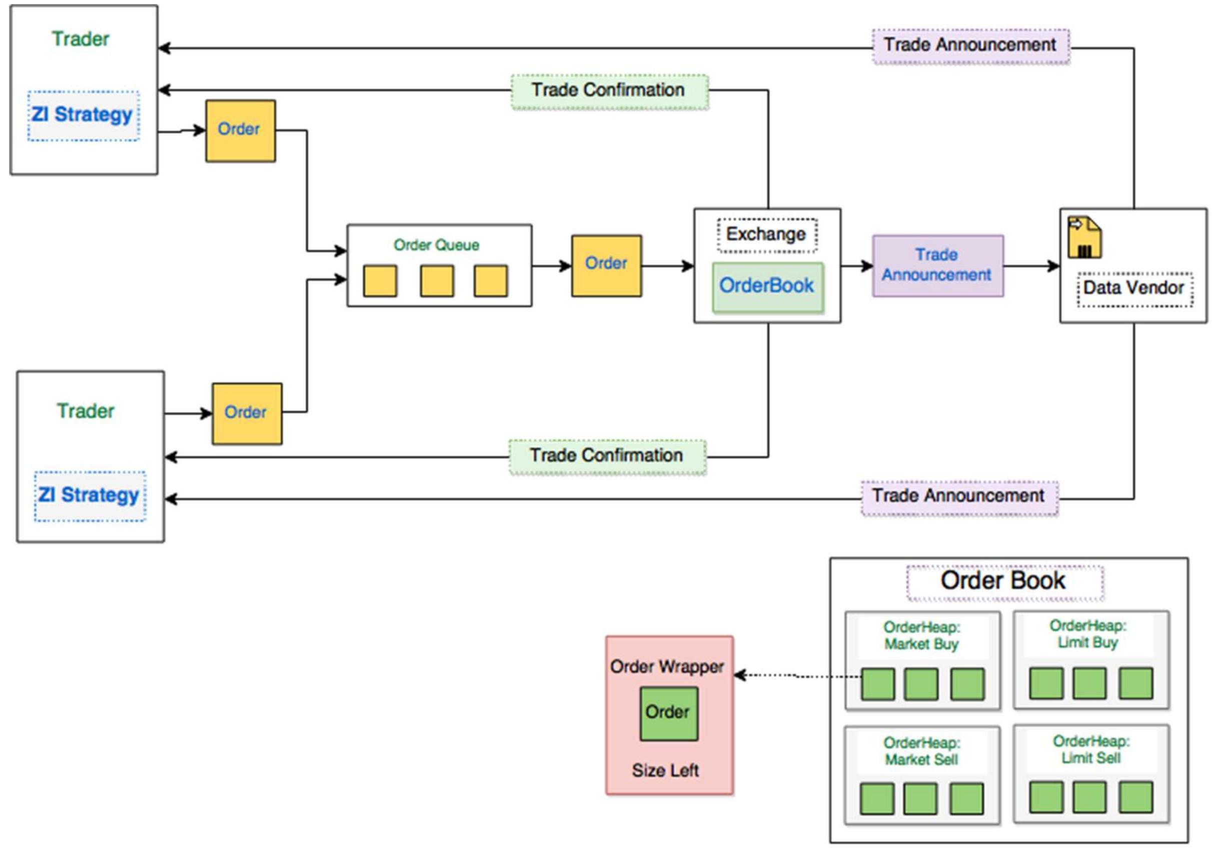


Figure 4.1: Exchange Design: Trader, Exchange, Order Book and Data Vendor

As elucidated previously, all traders in the simulation have no intelligence, and thus, all market and limit orders are randomly generated - following the principle of Zero Intelligence as used in a simulation where the focus of the research is not on strategy [24]. All generated orders are submitted to the Order Queue of the stock exchange market. The exchange order queue is an infinite First-in First-out (FIFO) queue, and so the queue is always available to receive and store arriving orders using the FIFO data structure. Orders are made available to the exchange on a discrete time space which is dependent on the exchange's latency and memory capacity. Orders are routed to the Order Book from the exchange for storage and processing, and this is usually done by the CEP.

The Order Book receives stores, crosses orders and submits trade records to the Exchange. The exchange buffers and processes corresponding trade records from the Order Book, and send trade confirmations to all traders connected to the exchange and trade announcements to Data Vendors. Data Vendors are responsible for notifying traders of reported trades. Order Wrapper wraps orders when available from Order Queue and adds the property SizeLeft, while OrderHeap is a priority queue storing either market-buy, market-sell, limit-buy or limit-sell orders.

This is an overview of the simulation design. To give more technical detail, the simulation model as implemented in SimEvents is divided into modules for clarity and explained under the following subsections.

4.1.1 Order Generation and Exchange

Event-based-entity-generator-block named Fast HFT generates entities (Orders) based on the signals received from the signal port (named: fcn) (see Figure 4.2). The signalling source where the time interval between each entity (Order) generation is generated and sent to the Event-based-entity-generator- block named Fast HFT is the event-based sequence block named Fast HFT Speed Signalling Source. In-between the Fast HFT Speed Signalling Source and Fast HFT block is the Signal-based-function-generator named Fast HFT Trade Generating Source which determines how many entities are generated in each event time.

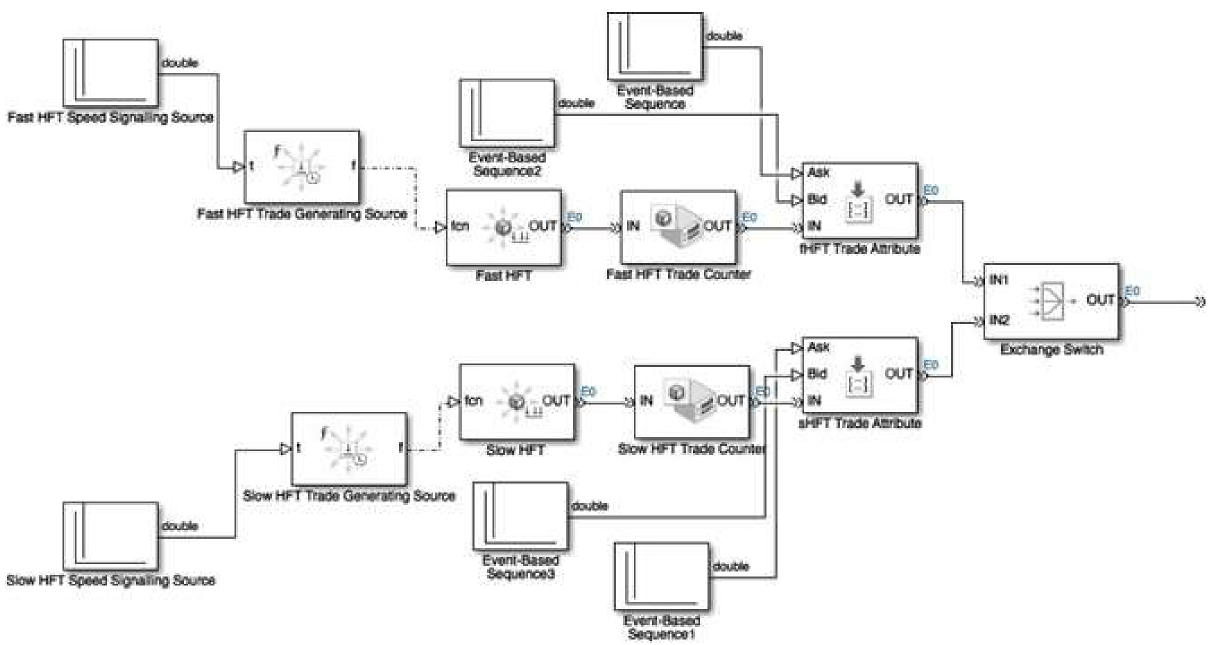


Figure 4.2: Simulation Model: Order Generation and Exchange

For example: if the signal output of the Fast HFT Speed Signalling Source is set to 0.001, which implies 1ms, this means that the Fast HFT will generate entities (Orders) at an interval of 1ms which is the Latency of the Fast HFT. However, the number of entities (Orders) generated at each 1ms time stamp is determined by the Fast HFT Trade Generating Source. If the output of the Fast HFT Trade Generating Source is set to 50, this means the Fast HFT will generate 50 entities (Orders) per 1ms. In the simulation, order size is assumed to be equal to 1 for simplicity, and thus, the Fast HFT Trade Generating Source is used to signify the number of Fast HFT traders we are considering - which means an output of 50 ⇒ 50 Fast HFTs. Due to the limitation of the working computer, a maximum of 200 HFTs in total were successfully simulated, considering that the simulation time-stamp is in milli-seconds. The same analogy as elucidated above applies to the Slow HFT section of the model.

The Fast HFT Trade Counter and Slow HFT Trade Counter are Entity-scope- blocks used to count and visualise the number of entities (Order) generated by each HFTs. The fHFT Trade Attribute and sHFT Trade Attribute are Set-attribute- block used to set the Bid and Ask attributes for the respective HFTs. The Exchange Switch is a Path-combiner used to route all the Orders to the Exchange Queue, which shall be discussed in the next subsection.

4.1.2 Exchange, CEP and ZI

The Exchange Switch is a Path-combiner used to route all the Orders to the Exchange Queue. The Exchange Queue is an infinite First-in, First-out (FIFO) queue used to hold all generated orders for processing by the exchange (see Figure 4.3). The infinite Exchange Queue enables all HFTs to generate orders without interruption throughout the simulation time. The Exchange. The server is a SimEvent-single-server-block that extracts entities (Orders) from the Exchange Queue for processing, and forward to the Trade Extractor, which is an Attribute-function-block. The Exchange Server's Latency is configured to be in micro-seconds - this allows the Exchange Server to be able to process all orders in the queue during the simulation time.

The Trade Extractor is an Attribute-function-block which represents the CEP (A minimal implementation of the CEP, since the real CEP performs more functions than extracting and forwarding Bid/Ask Orders). The Trade Extractor is used to extract Order-attribute information which is the respective Bid and Ask orders from the different HFTs, before sending them to the Order Book for crossing. In- between the CEP (Trade Extractor) and the Order Book is the ZI Trade Generator, this is used to randomly assign Bids and Ask prices to the respective separated Bids and Ask orders to be extracted and crossed by the Order Book.

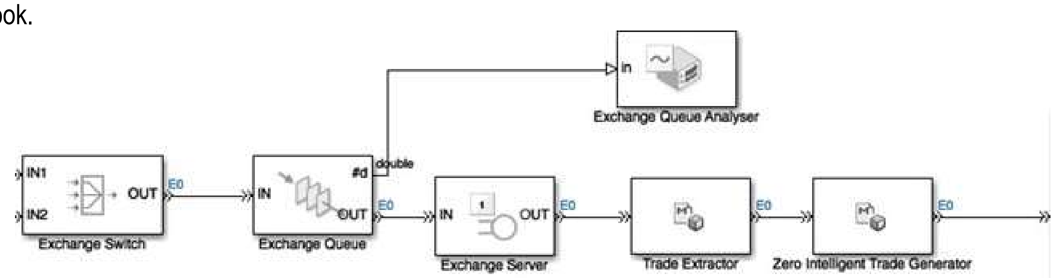


Figure 4.3: Exchange, CEP and ZI

The simulation functions slightly differently from the real system. In reality, the exchange holds Orders in temporary memory in a batch and processes them based on Price-Time-Priority. This means Orders are first sorted based on the best price followed by order arrival time. But in this research work, we assume there is no best or worse prices since our traders have no intelligence - our focus is not on strategy but on Latency, and so it is logical to process Orders solely based on time priority to investigate the effect of latency. More so, competition is assumed to be predominantly speed-based and not strategy [13].

4.1.3 Order Book

The Order Book Simulator is a MATLAB Function block used to simulate the Order Book (see Figure 4.4). The Order Book Simulator contains MATLAB codes and ABS Algorithms where each arriving Bid and Ask Order are treated as an individual and independent agent, whose arrival changes the state of the Order Book - Either decreasing the depth of the Order Book by removing an Order (Successful execution) or increasing the depth by adding Order that could not be instantly executed to the Order Book.

The Order Book Simulator is placed in-between two SimEvents blocks, which are Get Attribute and Set Attributes, respectively. The Get Attribute is used to parameterise the resulting Bid and Ask Order prices from the ZI Trade Generator for the Order Book Simulator for crossing, as shown. While the Get Attribute is used to parameterise the Order Book Simulator execution output. The output contains additional information, which is a Boolean parameter that represents successful execution and an execution price parameter. The Boolean parameter of ”1” and price ”P ” indicates that the Order is successfully executed at price ”P ”. While a boolean parameter of ”0” and price ”P ” indicates that the Order could not be executed at price ”P ” but appended to the Order Book.

The Matlab Function naturally runs based on the Matlab function call time- stamp, which is higher than the simulation time - i.e the function block is naturally slower than the milliseconds' rate at which orders are generated and expected to be processed in the simulation and therefore creating a lag. This necessitates the addition of an additional SimEvent single-server block with Zero Latency. This alters the natural latency of the function block and enables it to process Orders at Zero latency. The resulting output of all Order crossing from the Order Book is sent out through the Get Attribute block.

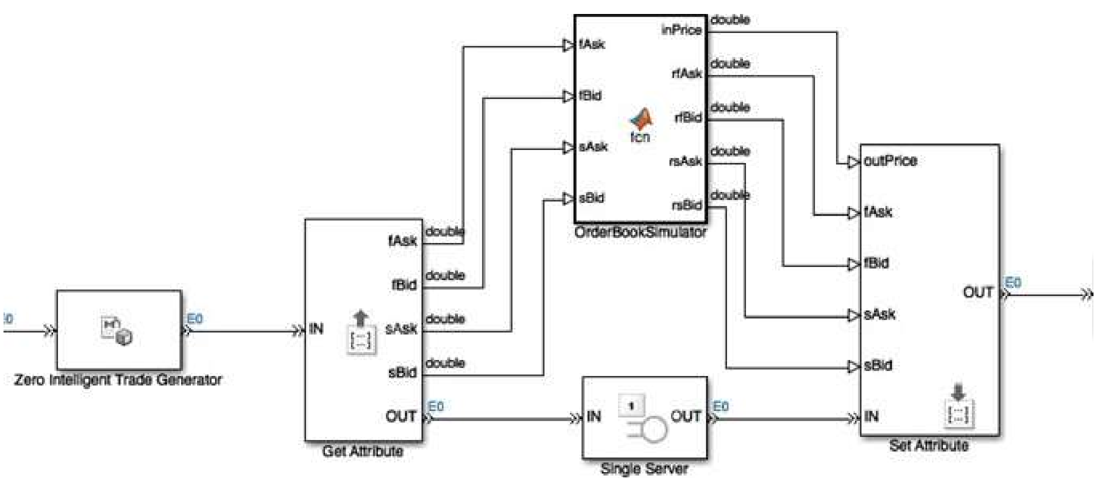


Figure 4.4: Order Book

4.1.4 Trade Confirmation and Announcements

The analyser block attached to the Order Book via the Set Attribute block is an Attribute Scope block used to extract the resulting output of each order as they exit the Order Book (see Figure 4.5). The Analyser block automatically stores, compute and output the statistics of each Ask and Bid order originating from each of the Fast and Slow HFTs, which can be visualised graphically. The Stock Price block automatically computes and outputs the Volume Weighted Average Price (VWAP). The simulation is terminated by an Entity Sink block, a block where all the entities in the simulation go to rest.

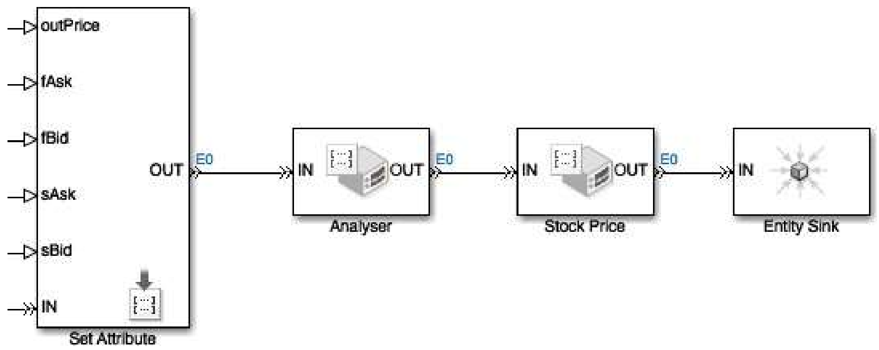


Figure 4.5: Trade Confirmation and Announcement

4.1.5 Market Data Feed

The Market Data Feed is a direct connection between a Trader and a Data Vendor. The link is utilised by the Data Vendor to send market data to all traders connected to its Data Feed, such as price, the volume traded, sizes, latest bid and ask price and the time of the last trade reported by a trading venue (see Figure 4.6). These market data are very important to traders (HFTs, AT, and non-HFTs) in making trade decisions (buying and selling decisions). Thus, the latency between an HFT and a Data Vendor may generate externalities and thus should be considered in the research.

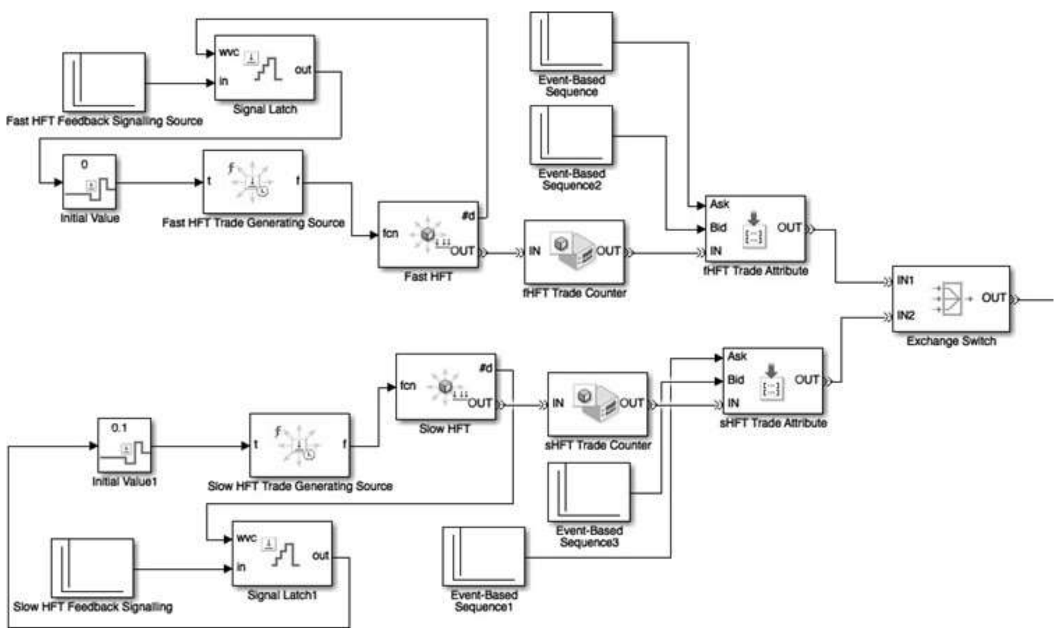


Figure 4.6: Market Data Feed

To model the latency difference and experimentally investigate the impact of HFT Latency on market quality with respect to the latency between HFTs and Data Vendors, we use the Signal Latch and Initial Value Block. The Initial Value Block sends a time delay to the respective HFTs, that is, the time to generate the first entity (Order). After the first entity is generated at this initial latency, the HFT starts receiving Latency signals from the Signalling Source via the Signal Latch, which is the speed used to send the order to the exchange.

For example: If the Latency for Order generation is 1ms and the value of the initial value is 5ms. Then the first entity (Order) is generated at 5ms, and the next Order generation is at 10ms and continues to generate at 1ms henceforth. This implies that a delay of 10ms has been introduced for the particular HFT compared to other HFTs in the market. This enables us to simulate and examine the effect of latency between an HFT Trader and a Data Vendor. Various scenarios will be simulated in the next Chapter to examine the effect of Data-feed latency on the market quality.

4.1.6 Sample Simulation Run

The debug window shown in Figure 4.7 below is a snapshot of the simulation run showing the System state, Event list, Event time, Entity advancement and so on. The first order or entity (en1) in the simulation was generated at time 0.01s ⇒ 10ms and originated from the Fast HFT. The order (en1) advances from Fast HFT to the Fast HFT Trade Counter, a block that keeps track of the number of trades originating from the Fast HFT. Advance from there to the fHFT Trade Attribute, where Bid and Ask attributes are assigned to each order. The attribute value of 1 is assigned to the Fast HFT, which will be used to uniquely identify all Orders originating from the Fast HFT by the ZI Order Generator, which randomly assigns prices to the Ask and Bid orders before being crossed by the Order Book. Then proceed to the Exchange Switch, followed by the Exchange Queue until the Order is crossed by the Order Book and finally ends up at the Entity Sink block.

```
Command Window
Executing EntityGeneration Event (ev4)             Time = 0.010000000000000
: Entity = <none>                                  Priority = SYS1
: Block  = Fast HFT
    %..........................................................................%
    Generating Entity (en1)
    : Block = Fast HFT
    %..........................................................................%
    Entity Advancing (en1)
    : From = Fast HFT
    : To   = Fast HFT Trade Counter
    %..........................................................................%
    Executing Scope
    : Block = Fast HFT Trade Counter
    %..........................................................................%
    Entity Advancing (en1)
    : From = Fast HFT Trade Counter
    : To   = fHFT Trade Attribute
    %..........................................................................%
    Setting Attribute on Entity (en1)
    : Ask = 1
    : Bid = 1
    : Block = fHFT Trade Attribute
    %..........................................................................%
    Entity Advancing (en1)
    : From = fHFT Trade Attribute
    : To   = Exchange Switch
    %..........................................................................%
    Entity Advancing (en1)
    : From = Exchange Switch
    : To   = Exchange Queue
    %..........................................................................%
    Queuing Entity (en1)
    : FIFO Pos = 1 of 1
    : Capacity = Inf
    : Block    = Exchange Queue
    %..........................................................................%
    Scheduling NewHeadOfQueue Event (ev5)
    : EventTime = 0.010000000000000 (Now)
    : Priority  = SYS2
    : Entity    = <none>
fx  : Block     = Exchange Queue
```

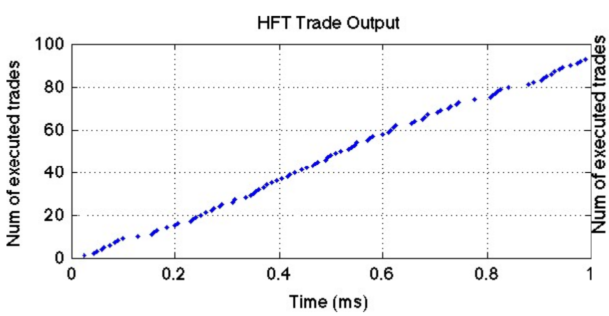
Figure 4.7: Snap-shot: Sample Simulation Run

5. EVALUATION AND RESULTS

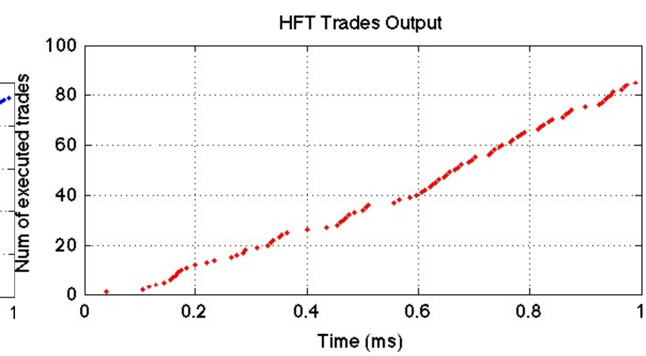
The developed Simulation model investigates the impact of latency on market quality which are Liquidity, Volatility and price discovery and is evaluated using different market setup scenarios. Four major fundamental scenarios are identified and tested, and the result of each scenario is documented, analysed and evaluated based on the proposed mathematical model and results of empirical research in the domain of financial market and High-Frequency Trading (HFT). The result of the simulation provides a novel explanation of HFT activities and the resulting latency impact on market quality. It provides a causal explanation for already established facts and new insights on HFT activities and impacts as related to Latency.

5.1 Scenario 1: HFTs With the Same Latency

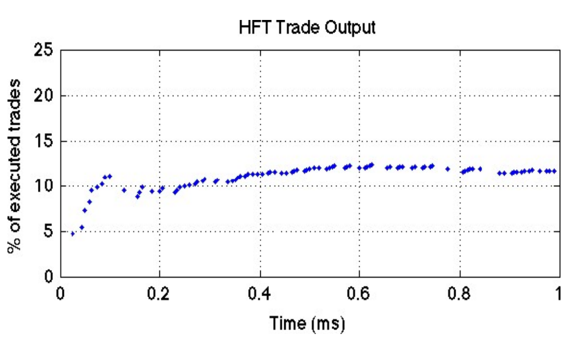
The first evaluation to be performed in the simulation is that of Two HFTs with the same latency to the Exchange (Latency = 5ms). This allows us to ensure the accurate working of the simulation model. Since a one-to-one scenario is very simple to predict in terms of outcome, this will allow us to identify any logical error or misplaced assumption that will negatively impact the result of subsequent simulations, which are critical to the research. A simulation time of 1 Second was chosen for the simulation and subsequent simulation. The result of the simulation is shown in Figure 5.1 below.



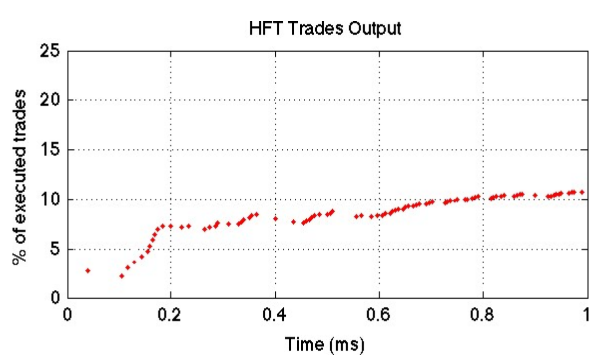
(a) HFT 1



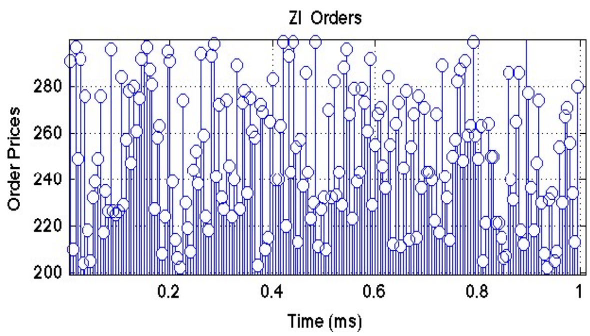
(b) HFT 2



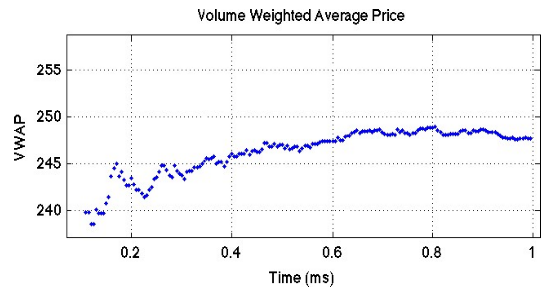
(c) HFT 1



(d) HFT 2



(e) ZI Orders



(f) VWAP

Figure 5.1: Two HFTs with equal latency to the Exchange

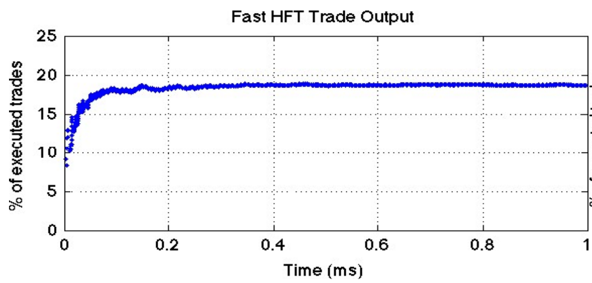
As seen in Figure 5.1 (a) and (b), HFT 1 was able to successfully execute 93 Orders, while HFT 2 successfully executed 85 Orders each. While the percentage of executed orders is 11.7% and 10.68%, respectively, as shown in Figure 5.1 (c) and (d). Despite the fact that orders were randomly generated based on the Zero Intelligence principle, as shown in Figure 5.1 (e) and given that both HFTs have the same latency of 5ms to the Exchange, HFT 1 still execute more trades than HFT 2 with different execution pattern as shown in Figure 5.1 (c) and (d). This shows that even though the Order price is generated from a set of uniform distributions with the boundaries a = 200 and b = 300, as shown in Figure 5.1 (e), where the probability of selecting a value within the boundaries a and b is equal.

The number of executed trades is not equal because the order-crossing process is stochastic, and the probability of a trade being executed is independent of the prior probability in the distribution from which the order is selected. This confirms that border crossing at the Exchange is exclusively stochastic, just like the real Exchange order book. The VWAP, as shown in Figure 5.1 (f) shows that there was little volatility in price at the beginning of the market and became stable shortly after at approximately 0.6s. This correlates with empirical results of price volatility at the opening of the market [27].

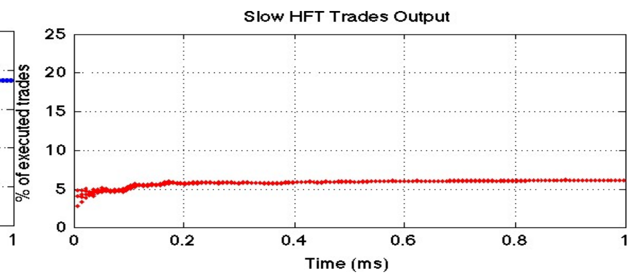
5.2 Scenario 2: HFTs With Varied Latency

We refer to Slow HFTs, ATs and Human Traders as slow HFTs (Abbreviated as sHFTs) . While the HFTs are assumed to be trading at ultra-low latency as Fast HFTs (Abbreviated as fHFTs). From Hasbrouck and Saar [28], they observed that the fastest trader has an effective speed of 2 − 3ms, while the slowest speed was observed to be 200ms in their research on the impact of milliseconds on security trading. In the simulation, a Latency of 2.5ms will be assigned to the Fast HFTs, while the Slow HFTs will be assigned a Latency range of 7.5ms to 102.5ms [23, 28]. From the result of empirical research, HFT constitutes, on average 70% of the equity market. A ratio of 7 : 3 will be used in all scenarios that investigate the impact of HFTs prevalence in terms of latency and numbers in the financial market.

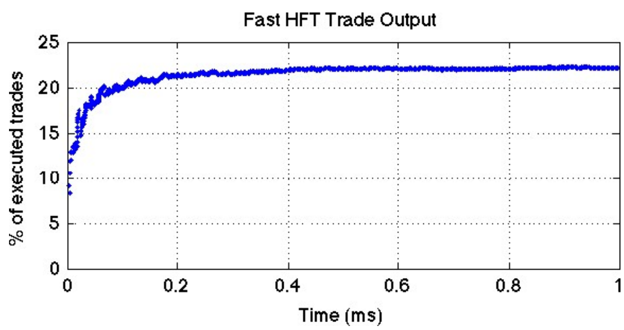
A scenario is used to evaluate the impact of HFT Latency when we have multiple HFTs trading at a different speed. Figure 5.2 shows the result of the simulation when there are 10 Fast HFTs (fHFTs) and 10 Slow HFTs (sHFTs) with Latencies 2.5ms,7.5ms; 2.5ms, 22.5ms; and 2.5ms, 102.5ms, respectively. The latency-lag in each of the simulations are 5ms, 20ms and 100ms, respectively. It can be observed that the percentage of executed trades (Orders) for the fHFTs and sHFTs at Latency 2.5ms and 7.5ms are 18.7 and 6.2, respectively as shown in Figure 5.2 (a) and (b). Comparatively, The sHFT percentage order execution at 1s is 66.8% lesser compare to the fHFT - This is a very huge cost, which can run into Billion of Dollars based on the TABB Group Estimate [23]. Also, making an estimation of the cost of latency from the result shows that the cost of 1ms ⇒ 0.236% loss in Order flow, which implies a loss of 1.18% of sHFT Order-flow for 5ms latency behind the fHFT. This is close to the estimate made by TABB Group [23] That if a broker's electronic trading platform is 5 milliseconds behind the competition, it could loose at least 1% of it's order-flow.



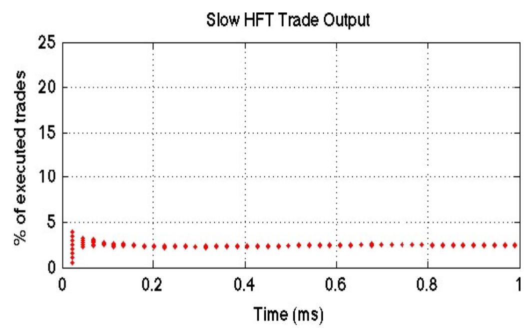
(a) *Latency = 2.5ms*



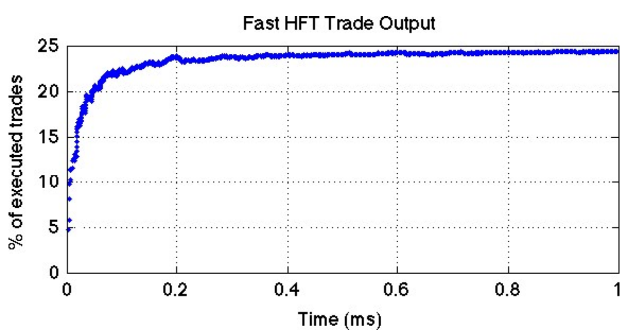
(b) *Latency = 7.5ms*



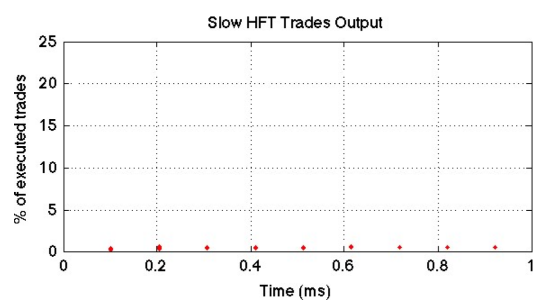
(c) *Latency = 2.5ms*



(d) *Latency = 22.5ms*



(e) *Latency = 2.5ms*



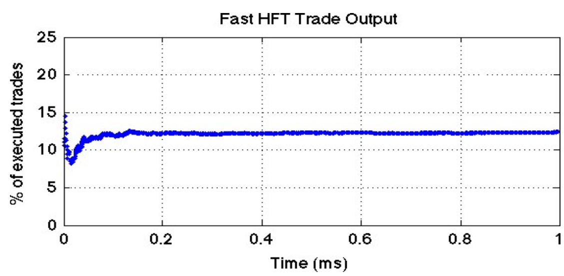
(f) *Latency = 102.5ms*

Figure 5.2: Multiple HFTs with varied latency to the Exchange (10 fHFTs and 10 sHFTs)

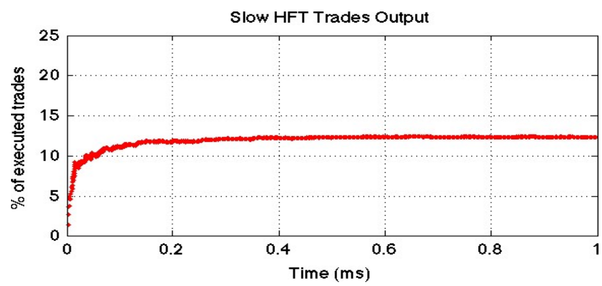
The latency difference is increased from 5ms to 20ms, The percentage of executed trades for the fHFT improved by 20% while that of sHFT worsened by 18% as shown in Figure 5.2 (c) and (d), respectively. We assume the maximum latency difference to be 100ms. Thus, increasing the latency difference by 100ms led to an improvement of 9% of total execution trade for the fHFT while that of sHFT worsened by 12% as shown in Figure 5.2 (e) and (f), respectively. An observation of the impact of latency on the Slow HFT shows that at a maximum latency lag of 100ms, only 0.49% of its total Orders were executed compared to the fHFTs whose total percentage of the order executed is 24.3%. This confirms the dominance of the fHFTs when the latency lag is very large. The slower HFTs are swept under the carpet because the market direction is predominantly determined by the fHFTs at this instance. According to [39], when the HFTs become prevalent, the non-HFTs (Slower Traders) withdraw from the market to avoid negative externalities.

5.3 Scenario 3: HFTs With Same Latency to Exchange But Varied Market Data Latency

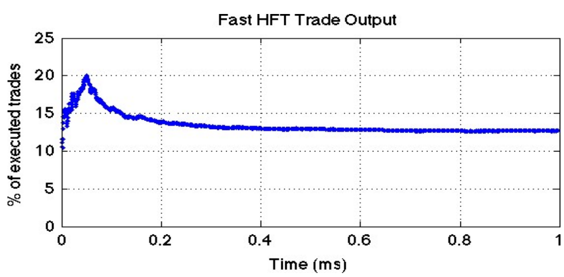
The next scenario run on the developed simulation model is that of multiple HFTs with same latency to the Exchange but varied latency in their market data-feed latency - the latency of the link from the trader to the Data Vendor. The result of the simulation is shown in Figure 5.3 (a) to (f).



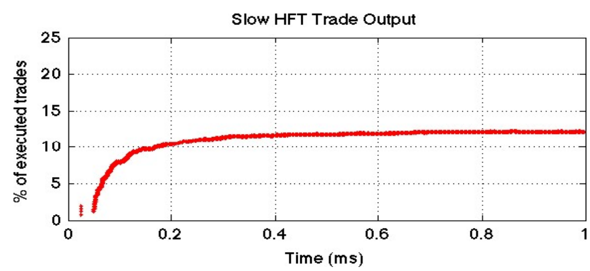
(a) *Data-Feed Latency-lag = 0ms*



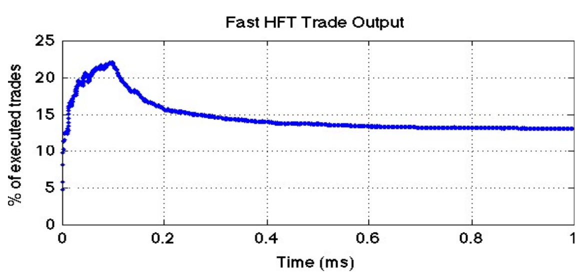
(b) *Data-Feed Latency-lag = 5ms*



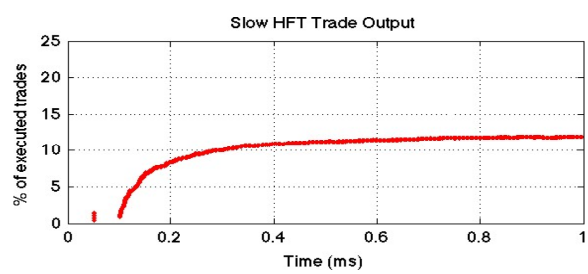
(c) *Data-Feed Latency-lag = 0ms*



(d) *Data-Feed Latency-lag = 50ms*



(e) *Data-Feed Latency-lag = 0ms*



(f) *Data-Feed Latency-lag = 100ms*

Figure 5.3: Multiple HFTs with the same latency to the Exchange but varied market data deed latency

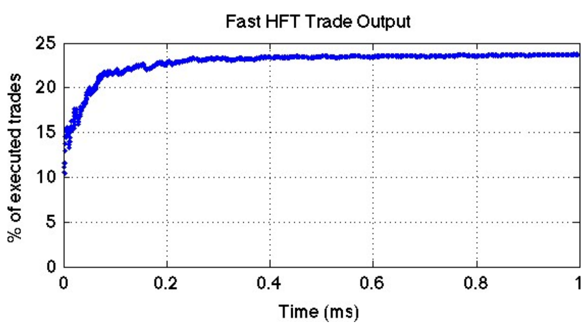
The HFTs with higher data-feed latency are referred to as Slow HFTs and while the HFTs with lower data-feed latency are referred to as Fast HFTs. In reality, HFTs speed are predominantly determined by their order submission latency i.e their respective latency to the Exchange. Mostly, Faster HFTs usually have lower latency both ways. The purpose of this scenario is to examine the effect of Data- feed latency on market quality and to be able to evaluate the effect of Data-feed latency. The order submission latency variable needs to be kept constant.

As shown above, there are 10 Fast HFTs and 10 Slow HFTs, both with a latency of 2.5ms to the Exchange. The market-data feed latency-lag between the fHFTs and the sHFTs varies as follows: 5ms, 50ms and 100ms, respectively. It can be observed from Figure 5.3 (a) and (b) that the fHFTs (in terms of Data- feed latency) started at 15% but suddenly dropped below 10% at 5ms when the sHFT entered the market. But at the end of the 1sec simulation time, both sill finishes at approximately 12.5%. However, looking at Figure 5.3 (c) and (d), a sharp downward trend is observed at 50ms in (c) as a result of the sHFT order placement. The two HFTs did not end the 1sec trade window with the same execution percentage (fHFT ended at 12.7% while sHFT ended at 12.2%) due to the large latency lag of 50ms. The same applies when the latency lag is 100ms as shown in Figure 5.3 (e) and (f). This shows that the impact of the data-feed latency is recoverable given that: 1. The latency lag is very small - approximately 5ms and 2. Provided that the order is large enough to last for a duration proportional to the data-feed latency-lag.

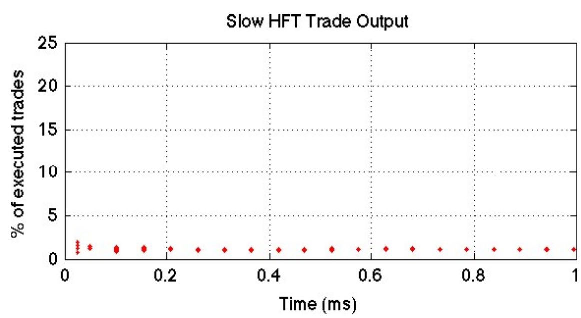
For example: In Figure 5.3 (e), if the order is to last for just 100ms, the fHFTs would have taking a price advantage and execute the orders before the sHFTs submit orders for the same security at 100ms later. And at this time, the sHFT would have incurred a cost. Because the best Bid/Ask price at that instance (at 100ms later) will either be lesser or greater than the Best Bid/Ask possessed by the sHFTs [14]. Thus, this position is un-recoverable even though the order last a lifetime due to the large information gap.

5.4 Scenario 4: HFTs With Varied Latency to Exchange And Varied Market Data Latency

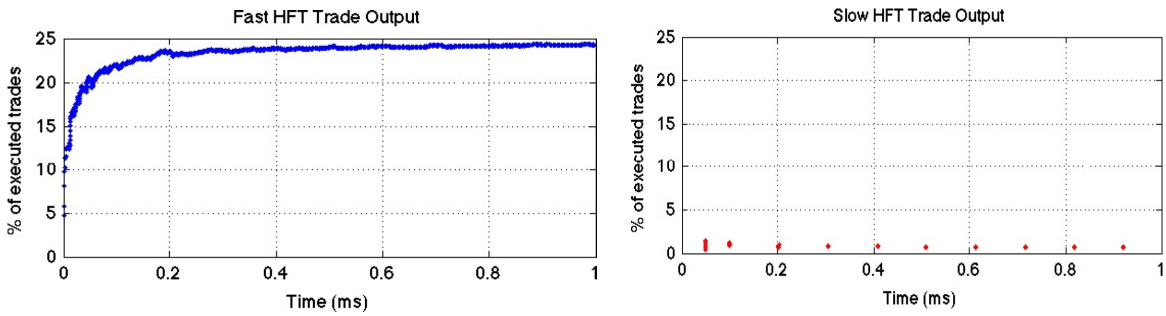
The last scenario run on the simulation is that of Fast HFTs and Slow HFTs (both with varied latency to the exchange and Data Vendor). The result of the simulation as shown in Figure 5.4 (a) to (d) shows that when an HFT has a high latency-lag, both to the exchange and to the data vendor, such has no place in the world of HFT. As shown in Figure 5.4 (a) and (c), the submission of Orders by the sHFTs at 50ms and 100ms has no effect on the percentage of executed orders of the fHFTs and thus, just as emphasised by Viraf “If a broker is 100ms slower than the fastest broker, it may as well sell his FIX (Financial Information eXchange) engine and become a floor broker” [23 p.8].



(a) *Exchange Latency = 2.5ms, Data-Vendor Latency-lag = 0ms*



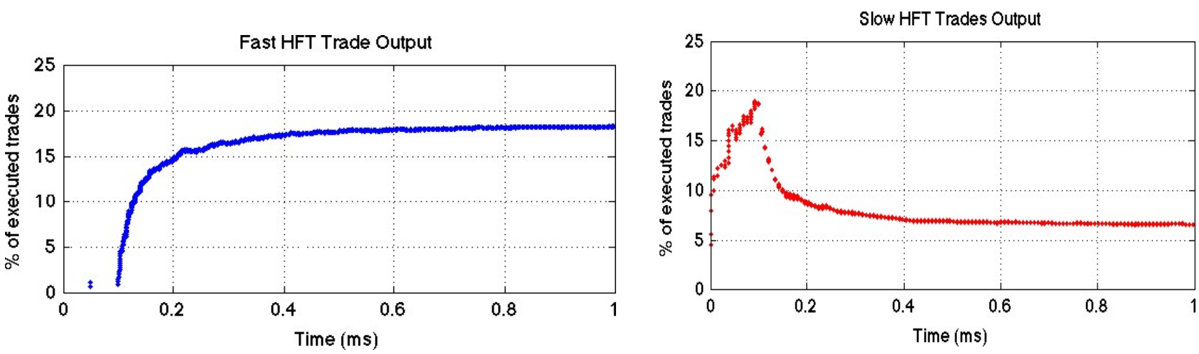
(b) *Exchange Latency-lag = 50ms, Data-Vendor Latency-lag = 50ms*



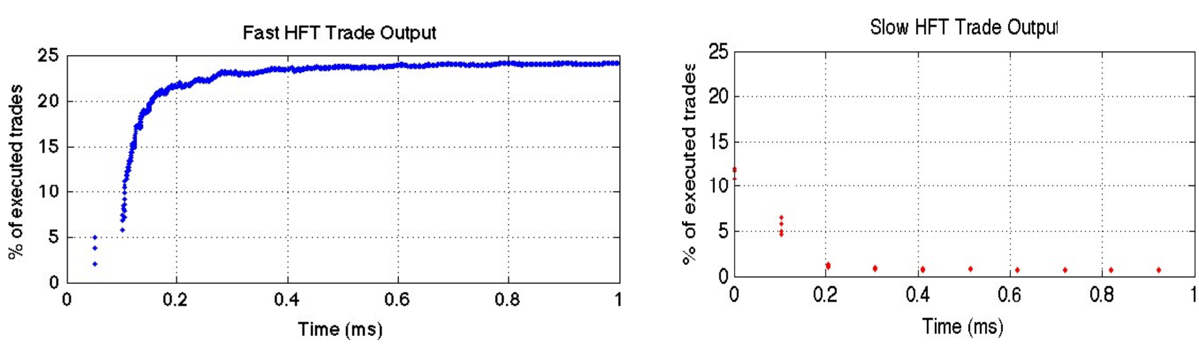
(c) *Exchange Latency = 2.5ms, Data-Vendor Latency Lag = 0ms* (d) *Exchange Latency = 100ms, Data-Vendor Latency Lag = 100ms*

Figure 5.4: Multiple HFTs with varied latency to the Exchange and to the Data Vendor

However, the latency variation is reversed to observe the difference, meaning that a data-vendor latency-lag of 100ms is assigned to the Fast HFTs with an Exchange latency of 2.5ms while a data-vendor latency-lag of 0ms is assigned to the Slow HFTs with Exchange latency-lag of 5ms and 100ms. The result of the simulation is shown in Figure 5.5 (a) to (d).



(a) *Exchange Latency = 2.5ms, Data-Vendor Latency-Latency-lag = 100ms* (b) *Exchange Latency-lag = 5ms, Data-Vendor lag = 0ms*



(c) *Exchange Latency = 2.5ms, Data-Vendor Latency-lag Latency Lag = 100ms* (d) *Exchange Latency = 100ms, Data-Vendor = 0ms*

Figure 5.5: Multiple HFTs with varied latency to the Exchange and to the data vendor (Reverse Test)

The result of the simulation shows that even though the fast HFTs started executing orders at 100ms, less than 100ms mark into the trade, they were already ahead of the Slow HFTs. This shows that the Exchange latency is most paramount compared to Data-Vendor latency when considering latency impact on market quality.

5. CONCLUSIONS

This research work is focused on the analysis and evaluation of the impact of HFT Latency on market quality which are liquidity, price discovery and volatility. To achieve this, a simulation model was built using Simulink and SimEvents from Math-Works. MATLAB Codes and Functions were also written to perform the tasks that are not already explicitly defined in Simulink and SimEvents modules in other to ensure the dynamics of the model are close enough to the real system (Financial Market). Discrete-Event Simulation (DES) paradigm was used to model the interaction of multiple traders (HFTs with varied latency and non-HFTs) with an Exchange. While Agent-Based Simulation (ABS) was used to model the interaction of the various traders (HFTs and non-HFTs) with the order book. Zero Intelligence principle was used to generate Orders by Traders in the simulation. This enables focus on the main variable under consideration (latency). The results of the simulation were presented, analysed and discussed.

The result of the simulation show that:

1. Contrary to what is obtainable in the research literature that HFTs reduce volatility. According to Hendershott and Riordan [20] that when prices deviate from fundamental value, HFTs push prices back to their efficient level by initiating a trade in the opposite direction and thereby contributing to price efficiency and increasing liquidity. Through the result of the Simulation, it is observed that HFTs volatilise security prices due to their latency dominance and later reverse their strategy to push prices back to their efficient level by trading in opposite direction and, thus, profiting from the Bid/Ask spread.
2. The latency between HFTs and the exchange (for Order submission) is most paramount compared to the market data-feed latency when investigating latency impact on market quality.
3. The impact of the data-feed latency is recoverable given that: 1. The latency lag is very small - approximately 5ms, and 2. Provided that the order is large enough to last for a duration proportional to the data-feed latency-lag.
4. The impact of the latency-lag for order submission is not recoverable within a trading window due to incurred latency-cost.

In summary, from the results of Simulation Modelling, we conclude as follows, that HFT latency:

1. Positively impact liquidity, but do so by volatilising price and later re-stabilising it. Menkveld[11] empirically confirms that inventory reverts to the mean position numerous times within a trading day, which implies the presence of a high rate of liquidity. The result of the simulations shows the presence of volatility when fHFTs are dominant "latency wise" - the observed volatility can only be caused by the dominant HFTs, which are the fHFTs, but no sign of volatility when fHFTs dominate in terms of "latency" and "number" which implies that the volatility is caused and cleared by the fHFTs which are the dominant party.
2. Contributes positively to price efficiency, but only to the fast HFTs, while the slow HFTs in-cured a latency cost for pricing information inefficiencies proportional to the latency-lag between the sHFTs and the market data-feed vendors.
3. Contribution to volatility depends on which side of the market we analyse, which is actually the reason for the conflicting results in the research literature [22]. Considering the opening and closing of a security price, it was observed that HFT positively impact volatility. But looking at the intermediaries, that is, in-between the opening and closing price. Volatility is introduced and cleared before the end of the 1 sec simulation window.

A future research should incorporate multiple exchanges to evaluate the impact of multiple exchanges' latency differences on market quality—both from the market participant perspective and the perspective of the exchange. Also, the principle of Zero Intelligence was used to exclusively restrict the simulation to testing the impact of latency. A future work should include non-zero intelligence agents and strategies in order to evaluate the correlation between strategy and latency, and their respective impact on market quality.

REFERENCES

[1] Riordan, R., & Storkenmaier, A. (2012). Latency, liquidity andprice discovery. Journal of Financial Markets, 15, 416–437.
[2] CFA Magazine: The Impact of High-Frequency Trading on Markets [Online]. Available at: https://www.rbccm.com/globalequity/file-569694.pdf [Accessed on: 20 June 2015].
[3] United States Commodities and Futures Trading Commission and Securities and Exchange Commission (2010), "Findings regarding the market events of May 6, 2010," Report of the Staffs of the CFTC and SEC to the Joint Advisory Committee on Emerging Regulatory Issues, September 30, 2010.
[4] Gomber,P.,Arndt, B.,Lutat,M., Uhle,T. (2011) High Frequency Trading [Online]. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1858626 [Accessed on: 1 June 2015].
[5] Brogaard, J. A. (2010). High frequency trading and its impact on market quality. Northwestern University Kellogg School of Management Working Paper [Online]. Available at: http://heartland.org/sites/default/files/htf.pdf [Accessed on: 20 June 2015].
[6] Jones, C. M. (2013). What do we know about high-frequency trading? Charles M. Jones* Columbia Business School Version 3.4: March 20, 2013. Columbia Business School.
[7] Hendershott, T., Jones, C. M., & Menkveld, A. J. (2011). Does Algorithmic Trading Increase Liquidity? Journal of Finance, 66(1), 1–33.
[8] Kirilenko, A., Kyle, A., & Samadi, M. (2010). The flash crash: The impact of high-frequency trading on an electronic market [Online]. Available at: http://www.ftm.nl/wpcontent/ up-loads/content/files/Onderzoek%20Flash%20Crash.pdf [Accessed on: 20 June 2015].
[9] Ye, M. (2012). The Externalities of High Frequency Trading, 1–48 [Online]. Available at: http://www.sec.gov/divisions/riskfin/seminar/ye031513.pdf [Accessed on: 20 June 2015]
[10] Zhang, X. F. (2010). The Effect of High-Frequency Trading on Stock Volatility andprice discovery, 1–53. [Online]. Available at: http://mitsloan.mit.edu/groups/template/pdf/Zhang.pdf [Accessed on: 20 June 2015]
[11] Menkveld, A. J. (2013). High-frequency trading and the new market makers. Journal of Financial Markets, 16(4), 712–740.
[12] Moallemi, C. C., & Sağlam, M. (2013). OR Forum—The Cost of Latency in High Frequency Trading. Operations Research, 61(5), 1070–1086. doi:10.1287/opre.2013.1165
[13] Viraf,W.(2008). The Value of a Millisecond: Finding the Optimal speed of a Trading Infrastructure [Online]. Available at: http://www.tabbgroup.com/PublicationDetail.aspx?PublicationID=346 [Accessed on: 20 June 2015]
[14] Ende, B., Uhle, T., & Weber, M. C. (2011). The Impact of a Millisecond : Measuring Latency Effects in Securities Trading. Proceedings of the 10th International Conference on Wirtschaftsinformatik (WI), (February), 27–37.
[15] Dhahahjay, K.D. & Shyam, K. (n.d.). Allocative Efficiency of Markets With Zero-Intelligence Traders: Market as a Partial Substitute for Individual Rationality. The Journal of Political Economy, 101(1), 119–137.
[16] Farmer, J. D., Patelli, P., & Zovko, I. I. (2005). The predictive power of zero intelligence in financial markets. Proceedings of the National Academy of Sciences of the United States of America, 102, 2254–2259.
[17] Brogaard, J. (2010). High Frequency Trading and its Impact on Market Quality, 5th Annual Conference on Empirical Legal Studies Paper. http://ssrn.com/ paper=1641387.
[18] Chaboud, A., Erik, H., Clara, V., & Ben, C. (2009). Rise of the Machines: Algorithmic Trading in the Foreign Exchange Market. [Online]. Available at: http://ssrn.com/ paper=1501135 [Accessed on: 21 June 2015]

[19] Gomber, P., Arndt, B., Lutat, M., & Uhle, T. (2011) High-Frequency Trading [Online]. Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1858626 [Accessed on: 12 July 2015]
[20] Hendershott, T., & Riordan, T. (2011), "High frequency trading and price discovery" working paper, UC Berkeley.
[21] Biais, B., & Moinas, S. (2011). Equilibrium High Frequency Trading [Online]. Available at: http://www.lse.ac.uk/fmg/events/conferences/pastconferences/ 2012/PWC-Conference_7-8June/Papers-and slides/ Bruno_Biais_paper.pdf [Accessed on: 22 July 2015].
[22] Boehmer, E., Fong, K., & Wu, J. (2012). International evidence on algorithmic trading. Available at SSRN 2022034, 2012, 49.
[23] Viraf, W. (2008). The Value of a Millisecond: Finding the Op- timal speed of a Trading Infrastructure, (April). Retrieved from http://www.tabbgroup.com/PublicationDetail.aspx?PublicationID=346
[24] Wah, E., & Wellman, M. (2013). Latency arbitrage, market fragmenta- tion, and efficiency: a two-market model. Proceedings of the Fourteenth ACM Conference on Electronic Commerce, 1(212), 855–872. Retrieved from http://dl.acm.org/citation.cfm?id=2482577
[25] Barr, D., Benos, E., Braun-munzinger, K., Butterworth, E., Chich- kanov, P., Cornelius, M., Meeks, R. (2012). Financial arms races, (April). Based on a speech delivered at the Institute for New Economic Thinking, Berlin 14 April 2012. [Online]. Available at: http://www.bankofengland.co.uk/publications/Documents/speeches/2012 /speech565.pdf [Accessed on: 20 June 2015]
[26] Budish, E., Cramton, P., & Shim, J. (2014). A Market Design Approach to the HFT Debate : The Case for Frequent Batch Auctions. [Online]. Available at: http://faculty.chicagobooth.edu/eric.budish/research/HFT-FrequentBatchAuctions.pdf [Accessed on: 20 June 2015]
[27] Ende, B., Uhle, T., & Weber, M. C. (2011). The Impact of a Millisecond : Measuring Latency Effects in Securities Trading. Proceedings of the 10th International Conference on Wirtschaftsinformatik (WI), (February).
[28] Hasbrouck, J., & Saar, G. (2013). Low-latency trading. Journal of Financial Markets, 16(4), 646–679.
[29] Goldstein, M. A., Kumar, P., & Graves, F. C. (2014). Computerized and High-Frequency Trading, 49(2), 1–35.
[30] Gode, D. K. and Sunder, S. (1993). Allocative efficiency of markets with zero- intelligence traders: Market as a partial substitute for individual rationality. Journal of Political Economy 101, 1, 119–137.
[31] Farmer, J. D., Patelli, P., & Zovko, I. I. (2005). The predictive power of zero intelligence in financial markets. Proceedings of the National Academy of Sciences of the United States of America, 102, 2254–2259.
[32] Bokil, V. A. (2009). Introduction to Mathematical Modelling. [Online]. Avail- able at: http://www.mesacc.edu/~davvu4111/IntroToModel.pdf [Accessed on: 20 June 2015].
[33] Dym, C. L. (1980). Principles of Mathematical Modelling. American Journal of Physics, 48(11), 994.
[34] Morgan, C. B., Banks, J., & Carson, J. S. (1984). Discrete-Event System Simulation. Technometrics, 26(2), 195.
[35] Karnon, J., Stahl, J., Brennan, A., Caro, J. J., Mar, J., & Möller, J. (2012). Modelling using Discrete Event Simulation: A Report of the ISPOR-SMDM Modelling Good Research Practices Task Force-4 Background to The Task Force. Value in Health, 15(6), 821–827.
[36] Mathworks SimEvents User's Guide for R2015a (2015). [Online]. Available at: http://cn.mathworks.com/help/pdf_doc/simevents/simevents_ug.pdf [Accessed on: 20 June 2015]
[37] Zheng, H., Son, Y.-J., Chiu, Y.-C., Head, L., Feng, Y., Xo, H., . . . Hickman, M. (2013). A Primer for Agent-Based Simulation and Modelling in Transport- ation Applications.
[38] Maidstone, R. (2012). Discrete Event Simulation, System Dynamics and Agent Based Simulation: Discussion and Comparison. System, 1–6.
[39] Biais, B., & Moinas, S. (2011). Equilibrium High Frequency Trading [Online]. Available at: http://www.lse.ac.uk/fmg/events/conferences/pastconferences/ 2012/PWC-Conference_7-8June/Papers-andslides/ Bruno_Biais_paper.pdf [Accessed on: 22 April 2015]

[40] Hendershott, T., & and Riordan, T. (2011), "High frequency trading and price discovery" working paper, UC Berkeley
[41] Mathworks Simulink Intro R2015a (2015). [Online]. Available at: http://uk.mathworks.com/products/simulink/ [Accessed on: 20 June 2015]
[42] Davis, J. P., Bingham, C. B., & Eisenhardt, K. M. (2007). Developing Theory Through Simulation Methods. Academy of Management Review, 32(2), 480–499.
[43] Bonabeau, E. (2009). Decisions 2.0 : The Power of Collective Intelligence. MIT Sloan Management Review, 50(50211), 45–52.
[44] Biais, B., & Foucault, T. (2014). HFT and market quality. Bankers, Markets & Investors, 128(1), 5-19.