Understanding Intra-day Price Formation Process by Agent-based Financial Market Simulation: Calibrating the Extended ChiarellaModel

Kang Gao,
Imperial College London and Simudyne
Email: kang.gao18@imperial.ac.uk

Perukrishnen Vytelingum, Simudyne
Stephen Weston, Deloitte UK and
Imperial College London
Wayne Luk, Imperial College London
Ce Guo, Imperial College London

>

Abstract
This article presents XGB-Chiarella, a powerful new approach for deploying agent-based models to generate realistic intra-day artificial financial price data. This approach is based on agent-based models, calibrated by XGBoost machine learning surrogate. Following the Extended Chiarella model, three types of trading agents are introduced in this agent-based model: fundamental traders, momentum traders, and noise traders. In particular, XGB-Chiarella focuses on configuring the simulation to accurately reflect real market behaviors. Instead of using the original Expectation-Maximization algorithm for parameter estimation, the agent-based Extended Chiarella model is calibrated using XGBoost machine learning surrogate. It is shown that the machine learning surrogate learned in the proposed method is an accurate proxy of the true agent-based market simulation. The proposed calibration method is superior to the original Expectation-Maximization parameter estimation in terms of the distance between historical and simulated stylized facts. With the same underlying model, the proposed methodology is capable of generating realistic price time series in various stocks listed on three different exchanges, which indicate the universality of intra-day price formation process. For the time scale (minutes) chosen in this paper, one agent per category is shown to be sufficient to capture the intra-day price formation process. The proposed XGB-Chiarella approach provides insights that the price formation process comprises the interactions between momentum traders, fundamental traders, and noise traders. It can also be used to enhance risk management by practitioners.

Keywords
Agent-based Model, Financial Market Simulator, Extended Chiarella Model, Price Formation

1 Introduction

1.1 Motivation

In the past decade algorithmic trading has grown rapidly across the world and has become the dominant way securities are traded in financial markets, currently generating more than half of the volume of US equity markets. Constantly improving computer technology and its application by both traders and exchanges, together with the evolution of market micro-structure, automation of price quotation and trade execution have together enabled faster trading. Consequently, intra-day price formation underpinning this trading process has become the focus of intense research attention in recent years as market participants attempt to gain greater insight into how prices are determined and hence improve trading performance.

Price formation determines the price of an asset through interactions between buyers and sellers. It is at the core of the efficient and transparent operation of markets for goods and services. The balance between buyers and sellers provides an effective indicator of demand and supply in a market, where demand and supply are generally significant but not the only driving factors behind price movements. This is because the mechanisms of price discovery indicate what sellers are willing to accept and what buyers are willing to pay, so the price discovery process is concerned with finding an equilibrium (or near equilibrium) price that enables the greatest liquidity for that asset at a given point in time. Beyond supply and demand, attitudes to risk, volatilities, available information and market micro-structure all exert varying levels of influence on the price discovery process.

Lo (2017) explains the process of price formation in properly functioning markets as generally involving market participants engaging in cause-and-effect reasoning along the lines of "if my strategy is x, then the market will respond with y, in which case I will respond with z…." Even this simple process requires that the algorithms of buyers and sellers have some understanding of the other's motives and incentives. Theoretically, this approach would imply that such chain reasoning could infinitely recurse.

However, Sirignano and Cont (2019) define a "price formation mechanism" as a high-level map representing the relationship between asset price and variables such as order flow and market price history. Modeling such a mechanism using stochastic differential equation models, machine learning prediction models and market micro-structure, all provide different ways to represent this map. However, an issue central to the price formation mechanism is the degree to which such a high-level map is universal. That is, whether the price formation mechanism is independent of the particular asset being considered. The universal existence of certain empirical stylized facts seems to be evidence supporting this universality traded. In this work, we present evidence for the existence of such a universal price formation mechanism by proposing the XGB-Chiarella method, which is able to reproduce realistic synthetic data for various stocks on different exchanges. The fact that the XGB-Chiarella method is based on the same underlying agent-based model backs the existence of a universal price formation mechanism.

Our investigation of intra-day price formation process is through financial market simulation using agent-based models. Financial markets are obviously one of the most dynamic systems in existence. With huge potential academic and industrial value, financial market simulation in agent-based models is an exciting new field for exploring behaviors of financial markets. In an agent-based artificial financial market, heterogeneous agents (traders) trade a financial instrument through a realistic trading mechanism for price formation. Unlike traditional economic theories, there is no equilibrium assumption in agent-based financial markets. In addition, traders are no longer assumed to have rational behaviors as in traditional economic theories. The removal of these assumptions makes agent-based financial market simulation more realistic than traditional equilibrium-based economic and financial theories. Various agent-based simulators have been developed in literature. However, there are still gaps in creating ideal agent-based financial market simulators that are capable of generating realistic synthetic market data and shedding light on the intra-day price formation process. Specifically, most existing agent-based financial markets are of lower frequency such as daily or monthly. To investigate intra-day price formation process, higher simulation frequency is needed.

To ensure realism in the generated intra-day financial time series data, parameters of the agent-based model must be calibrated to be as close to real market data as possible. Realistic simulated market data are supposed to exhibit certain characteristics known as stylized facts, which are universally observed in historical financial market data. Some stylized facts originate from behaviors of market participants, while others could be natural consequences of market structure design. Examples of stylized facts include fat tails of returns, volatility clustering, etc. Parameter calibration with respect to certain stylized facts can be extremely challenging due to the huge parameter space and complexity in designing explicit optimization objective function.

To sum up, there are still two challenges of great interest in this field:

- C1: To implement an agent-based financial market simulator which allows for the investigation of intra-day price formation process in financial market.
- C2: To calibrate the agent-based financial market simulator to ensure realism and reproduce common stylized facts.

To address the two challenges, we developed XGB-Chiarella, which is a novel approach to developing and calibrating an intra-day financial market simulator to narrow the existing gaps. The XGB-Chiarella methodology has two essential components: the underlying mathematical model for simulation and the calibration workflow with surrogate modeling. For the underlying model, the simple but powerful Extended Chiarella model (Majewski *et al.,* 2020), which consists of fundamental traders, momentum traders, and noise traders, is used as the underlying model for the XGB-Chiarella method. For the calibration workflow, the method utilizes XGBoost[1] algorithm to build a machine learning surrogate for the purpose of model calibration. We will show that even with only one agent for each type of trader, the XGB-Chiarella method is able to generate realistic price series simulations after proper model parameter calibration process.

1.2 Background and related work

With the rapid development of modern financial markets, price formation process in financial market has been of great interest to both researchers and practitioners for many years. One group of price formation process literature is based on the equilibrium state of financial market. Faias *et al.* (2011) propose a pure exchange economy with a finite number of types of agents and commodities. They analyze the equilibrium price formation in a differential information market, where traders have incomplete and asymmetric information. Jackson (1991) shows that it is possible to have an equilibrium with fully revealing prices and costly information acquisition if the price formation process is modeled explicit and traders are not price-takers. Price formation process is also extensively investigated from the perspective of double auction market. Cason and Friedman (1996) present 14 laboratory experiments that examine the price formation process in the continuous double auction. It is shown that participants in their double auction market experiments succeed in discovering prices that would achieve most of the exchange surplus. The same is true even with no auctioneer and with traders possessing various private information. Gerety and Mulherin (1994) examine the relationships between market structure and stock price formation. It is shown that trading mechanism and price formation provide different explanations for the greater volatility at the opening of trading.

Unlike the above works, in this article we analyze the price formation process in financial markets from another aspect, which focuses on the so-called trend and value effects. Trend and value effects are indispensable when it comes to price formation process in financial market. The two interactive effects pervade all financial markets. Trend refers to the price behavior that positive (negative) returns are more likely to be followed by positive (negative) returns. Value means that the asset price will converge to the intrinsic value of the asset, which indicates that assets with prices higher (lower) than their fundamental value tend to achieve negative (positive) future returns. The trend and value effects correspond to two types of market participants: fundamental trader and momentum trader. Fundamental trader reacts to the difference between fundamental value and market price, while momentum trader reacts to price trends. Lots of models investigating trend and value effects are analyzed in the literature.

Beja and Goldman (1980) build a model that proves that the so-called "speculation on the price trend" plays an important role in the formation of dynamic price behaviors. It is shown that speculative trading can accelerate the convergence to the equilibrium state, but it can also lead to price oscillations and market instability. Frankel and Froot (1986) present a model containing fundamentalists and chartists to explain the dollar price in the early 1980s. Not constrained by the assumption of utterly rational behaviors, in this model each type of trader performs the specific task in a reasonable and realistic way. Their model provides a framework for explaining price formation process in a variety of asset markets. Zeeman (1974) shows that the unstable behaviors in various financial markets can be credited to the interactions between fundamental traders and momentum traders. In one famous paper, Chiarella (1992) proposes the so-called Chiarella model, which consists of fundamentalists and chartists in the artificial financial market. It is shown that the Chiarella model is capable of generating a number of dynamic regimes in financial market that are consistent with empirical evidence. Based on the Chiarella model, Majewski *et al.* (2020) propose an Extended Chiarella model by adding noise traders and changing the demand function of fundamentalists. The Extended Chiarella model investigates the co-existence and interaction between trend and value effects in the framework of agent-based models. The model parameters are estimated using an Expectation-Maximization algorithm. Nevertheless, all the above models share some common drawbacks. Firstly, all the above models are estimated by mathematical derivation. Consequently, all those models generate theoretical results instead of actual simulated results. Secondly, existing models are used to explain daily or monthly price behaviors. There still exists a large gap in successfully explaining intra-day price formation process in terms of trend and value effects.

In this work, we examine the intra-day price formation process in the framework of agent-based models, where trend and value effects are represented by heterogeneous traders. Given an agent-based model, how to effectively calibrate the model to real data is still an open challenge. Successful calibration of an agent-based model enables the model to generate qualitative or quantitative properties that are observed consistently in empirically measured data and cannot be reproduced using traditional equilibrium-based approaches (LeBaron, 2006). Lots of calibration methods have been proposed in the literature. For some simple models such as the CATS model in Bianchi *et al.* (2007), model parameters can be read directly from the data. Analytical method is another class of agent-based model calibration method. For example, Majewski *et al.* (2020) apply Expectation-Maximization method to get the maximum likelihood estimation of the parameters of the Extended Chiarella model. However, for most complex models, model parameters are not directly observable. In addition, most agent-based models that are of interest are incompatible with calibration methods that require analytical solutions. In those cases, the only choice for model calibration is simulation-based method. The most common simulation-based calibration method is the simulated minimum distance method and its variations (Grazzini and Richiardi, 2015). The simulated minimum distance method involves the construction of an objective function that measures the distance between real data and simulated data for a given set of model parameters. Optimization methods are subsequently applied to minimize the distance to get an optimal set of model parameters. In the context of economic agent-based models, popular distance measures include weighted sums of the squared errors between empirical moments and simulated moments. Franke (2009) applies the method of simulated moments to estimate the parameters of an agent-based asset pricing model. Their moment selection emphasizes the reflection of certain stylized facts in financial markets, such as the fat tails and autocorrelation patterns of the daily returns in stock price time series. To explore structural stochastic volatility, Franke and Westerhoff (2012) employ the method of simulated moments to estimate parameters of different candidate agent-based asset pricing models. They also take into consideration the proportion of Monte Carlo simulation runs that yield moments within the empirical moments confidence intervals.

Another large obstacle in the development of robust and widely applicable agent-based model calibration strategies is the computational complexity. Most agent-based models of interest are computational costly to simulate, and the

>

situation is even worse when it comes to large-scale agent-based models. One possible solution to address this challenge is the use of surrogate modeling to help guide parameter space exploration and thus avoid a large amount of intensive agent-based model simulations. Examples of surrogate modeling include kriging (Rasmussen, 2003) and machine learning surrogate approach (Lamperti *et al.*, 2018). Dosi *et al.* (2018) apply kriging method to enable a global exploration of the parameters space for a multi-firm evolutionary simulation model. Sensitivity analysis is also carried out in this kriging framework. Lamperti *et al.* (2018) build machine learning surrogate models to approximate two agent-based models. Experimental results show that their XGBoost-based machine learning surrogate achieves high accuracy in approximating the relationship between model parameters and model outputs. The approach of machine learning surrogate modeling is also the foundation of the calibration method in this article.

1.3 Our contributions

In this paper, we propose XGB-Chiarella—a method for developing and calibrating an agent-based market simulator to generate realistic intra-day financial market data. The underlying model is the Extended Chiarella model (Majewski *et al.*, 2020). We adapted the machine learning surrogate modeling approach in Lamperti *et al.* (2018) to calibrate the agent-based financial market simulator. By reproducing realistic synthetic data for various stocks, we show that there is a universal intra-day price formation process that involves trend and value effects. The main contributions are:

- Addressing challenge C1: The Extended Chiarella model was originally tested on monthly data. To address Challenge C1, we further tested the Extended Chiarella model in intra-day minute data to explain the intra-day price formation process. The model is tested extensively in more than 75 stocks from three major exchanges in the world: Nasdaq, the London Stock Exchange, and the Hong Kong Stock Exchange. For all the stocks on the different exchanges, the experimental results are similar, and the simulator can all produce realistic simulated financial time series. This shows that trend and value effects exist universally in the stock market price formation process, regardless of the exchanges.
- Addressing challenge C2: Instead of the Expectation-Maximization algorithm, we propose a novel application of a recent machine learning surrogate modeling approach (Lamperti *et al.*, 2018) to calibrate the Extended Chiarella model, which addresses challenge C2. The foundation for this calibration method is the surrogate modeling approach in Lamperti *et al.* (2018). Instead of building an XGBoost classifier to predict positive calibration and negative calibration, we train an XGBoost regressor to innovatively predict the actual distance between simulated stylized facts and historical stylized facts. The distance involves not only the return distribution but also the autocorrelations between returns and squared returns. In addition, exploration-exploitation mechanism is introduced in the iterative process of selecting new points in parameter space. In terms of stylized facts distance, results show that the proposed method performs much better than the baseline Expectation-Maximization estimation algorithm.

2 Model architecture

This section presents the set-up and components of the agent-based financial market simulator.

2.1 Model set-up

We denote the price of a stock at time t as P_t. The total signed volume traded on the market from t to $t+\Delta$ constitutes the cumulative demand imbalance in the same period. This quantity is denoted as $D(t,t+\Delta)$. This aggregated demand depends on the trading strategies of various types of market participants. Following Majewski *et al.* (2020) and Kyle (1985), the price dynamics is assumed to be governed by a linear price impact mechanism:

$$P_{t+\Delta} - P_t = \lambda D\left(t,t+\Delta\right) \tag{1}$$

where λ is called "Kyle's lambda", which is related to the liquidity of the market and is a first-order approximation of market price sensitivity to market demand and supply. The market participants are assumed to be heterogeneous in their trading decisions. Following the Extended Chiarella model (Majewski *et al.*, 2020), we populate our model with fundamental traders, momentum traders, and noise traders. Since traders of the same type exhibit same behaviors, we only use one agent for each type of trader. This single agent represents the corresponding type of traders in the market. With only three agents in the model, simulation process is significantly accelerated. Each trader is associated with some parameters that control the trading behaviors and the amount of demand generated by the corresponding trader group. We will show the calibration of these parameters in later sections.

2.2 Fundamental trader

Fundamental traders make their trading decisions based on the perceived fundamental value of the stock. The fundamental value is denoted as V_t. A fundamentalist will tend to buy a stock if the stock is under-priced ($V_t - P_t > 0$); otherwise, it will tend to sell the stock. Following the convention in Chiarella (1992), in this work we assume the aggregated demand of fundamental traders is proportional to the level of mispricing. In other words, the aggregated demand of fundamentalists is $\kappa(V_t - P_t)$, where κ controls the overall demand generated by fundamental traders. The fundamental value V_t is an exogenous signal that is input to the model.

2.3 Momentum trader

Momentum traders are also called "Chartists". This group of traders buys and sells financial assets after being influenced by recent price trends. The assumption is to take advantage of upward or downward trend of the stock prices until the trend starts to fade. Instead of looking at the fundamental value of the stock, momentum traders focus more on recent price action and price movement. If the stock price has recently been rising, a long position is established; otherwise, momentum traders will enter a short position.

There are lots of methods to estimate the momentum of stock prices. A common trend signal is the exponentially weighted moving average of past returns with decay rate α. This trend signal is denoted by M_t:

$$M_t = \left(1-\alpha\right)M_{t-1} + \alpha\left(p_t - p_{t-1}\right) \tag{2}$$

where α is the decay rate. Given the trend signal M_t, the demand function of momentum traders is denoted as $f(M_t)$. The demand function $f(M_t)$ must satisfy two conditions:

- $f(M_t)$ is increasing.
- $f''(M_t)^*M_t < 0$

where the first condition is consistent with the nature of momentum trading and the second condition imposes the risk-averse assumption to momentum traders. Consistent with Chiarella (1992), here we choose $f(M_t) = \beta \tanh(\gamma M_t)$ with the requirement that $\gamma > 0$. γ represents the saturation of momentum traders' demand when momentum signals are very large. This phenomenon is partly due to, for example, budget constraints and risk aversion, which is prevalent in real chartists. β controls the overall demand generated by momentum traders. β is also assumed to be positive, i.e., the demand of momentum traders is positive when the momentum signal (M_t) is positive; otherwise, the demand is negative. The choice of this demand function for momentum traders strictly satisfies the two requirements.

2.4 Noise trader

Another group of market participants is noise traders. They are designed so as to capture other market activities that are not reflected by trend-following and value investing. As a result, the cumulative demand of noise traders can be described by a random walk. The random walk is also multiplied by a parameter σ_N, which controls the overall demand level from noise traders. Mathematically, for each step the demand from noise traders is sampled from a normal distribution with zero mean and standard deviation σ_N.

2.5 Model dynamics

2.5.1 Simulation process

Using mathematical formulas to approximate the supply and demand generated by all the participating traders, the resulting model dynamics for $\Delta t \to 0$ can be described by the following dynamical system. Note that here the fundamental value V_t is an exogenous signal that is input to the model, which is a major adaption from the original Extended Chiarella model. In addition, there are only three agents in our simulation, namely, fundamental trader, momentum trader, and noise trader. Each trader creates a demand corresponding to one term in Equation (3). Our simulation results will show that this model is able to generate very realistic artificial financial price time series, even though there are only three agents included in the model.

$$\begin{aligned} D(t, t+\Delta t) &= \underbrace{\kappa'(V_t - P_t)\Delta t}_{\text{Fundamental}} + \underbrace{\beta' \tanh(\gamma M_t)\Delta t}_{\text{Momentum}} + \underbrace{\sigma'_N \epsilon_t \sqrt{\Delta t}}_{\text{Noise}} \\ dM_t &= -\alpha M_t \Delta t + \alpha dP_t \end{aligned} \tag{3}$$

where ϵ_t follows standard normal distribution. Substitute Equation (1) into the above equations, we can get:

$$\begin{aligned} dP_t &= \kappa(V_t - P_t)\Delta t + \beta \tanh(\gamma M_t)\Delta t + \sigma_N \epsilon_t \sqrt{\Delta t} \\ dM_t &= -\alpha M_t \Delta t + \alpha dP_t \end{aligned} \tag{4}$$

where κ, β, σ_N equal to $\lambda\kappa', \lambda\beta', \lambda\sigma'_N$, respectively. Furthermore, the simulator runs according to a discrete-time version of model (4), in which Δt is 1, corresponding to one unit of the smallest simulation time interval:

$$\begin{aligned} P_t - P_{t-1} &= \kappa(V_t - P_t) + \beta \tanh(\gamma M_t) + \sigma_N \epsilon_t \\ M_t &= (1-\alpha) M_{t-1} + \alpha(P_t - P_{t-1}) \end{aligned} \tag{5}$$

The whole simulation runs according to Equation (5). For each step, each trader collects and processes market information. Internal variables associated with each trader are calculated. According to agent types and values of internal variables, demands are generated by the traders. The price evolves according to Equation (1).

2.5.2 Fundamental value from Kalman Smoother

The only remaining unknown variable is the fundamental value of the stock. According to Equation (5), the simulation can proceed only if fundamental value is known and is exogenously input to the model. One difficulty is the non-observability of the fundamental value. According to the economic literature, the fundamental value of a stock equals to the expected value of discounted dividends that the company will pay to shareholders in the future. However, this methodology requires extremely strong assumptions of the future dynamics of the stock dividends. Furthermore, this approach can never reflect the intra-day change of fundamental value, while the consensus fundamental value can indeed vary during the trading day due to the continuous feed of events and news.

In this paper, we propose a novel method, which is to apply Kalman Smoother (Ralaivola and d'Alche Buc, 2005) directly to the stock price time series to get the hidden fundamental value. Note that Equation (5) is a linear dynamical system in V_t, which is treated as a hidden variable of the system. Here the observations are the actual prices traded in real market. The specific Kalman Smoothing algorithm used here can be found in Byron *et al.* (2004). The algorithm is already implemented in the Python package "pykalman".

3 Model calibration

In this section, we present the methodology for calibrating the agent-based financial market simulator. Calibration means finding an optimal set of model parameters to make the model generate most realistic simulated financial market. Firstly, we describe the real data and the associated stylized facts in financial markets. Next, we define the distance between historical and simulated stylized facts, which acts as the loss function in the calibration process. Finally, the machine learning surrogate modeling for parameter space exploration is presented in detail.

3.1 Data

In the model calibration process, real financial market data are essential to set up the calibration target. We collected stock price data of 75 stocks from three major exchanges in the world—Nasdaq, the London Stock Exchange, and Hong Kong Stock Exchange. Our dataset comprises intra-day minute price data for those 75 stocks, spanning the entire trading period from September 20, 2021, to November 30, 2021. Detailed stocks from each exchange are shown in Table 1.

Table 1: Specific stocks from three exchanges in the dataset.

Exchange	Stocks
Nasdaq	AAPL, HUT, AMD, AAL, MSFT, INTC, UBER, NVDA, SOFI, DKNG, WISH, HON, FB, TSLA, MRNA, AFRM, LCID, CMCSA, HBAN, TLRY, MU, CSX, CSCO, JD, UAL
LSEG	LLOY.L, VOD.L, RR.L, GLEN.L, IAG.L, HSBA.L, BP.L, PRU.L, DGE.L, LGEN.L, AAL.L, AV.L, RDSA.L, STAN.L, ULVR.L, BHP.L, BATS.L, RIO.L, GSK.L, NG.L, REL.L, EZJ.L, JET.L, SMT.L, BRBY.L
HKEX	0700.HK, 3690.HK, 9988.HK, 1299.HK, 1211.HK, 2020.HK, 2269.HK, 0175.HK, 2331.HK, 1024.HK, 0916.HK, 2318.HK, 1918.HK, 9618.HK, 9999.HK, 2333.HK, 9888.HK, 0836.HK, 1919.HK, 0388.HK, 3968.HK, 1772.HK, 1171.HK, 0005.HK, 0027.HK

Figure 1: Cumulative distribution function of historical returns in two stocks (FB, RIO.L) for one trading day, in comparison with a normal distribution reference.

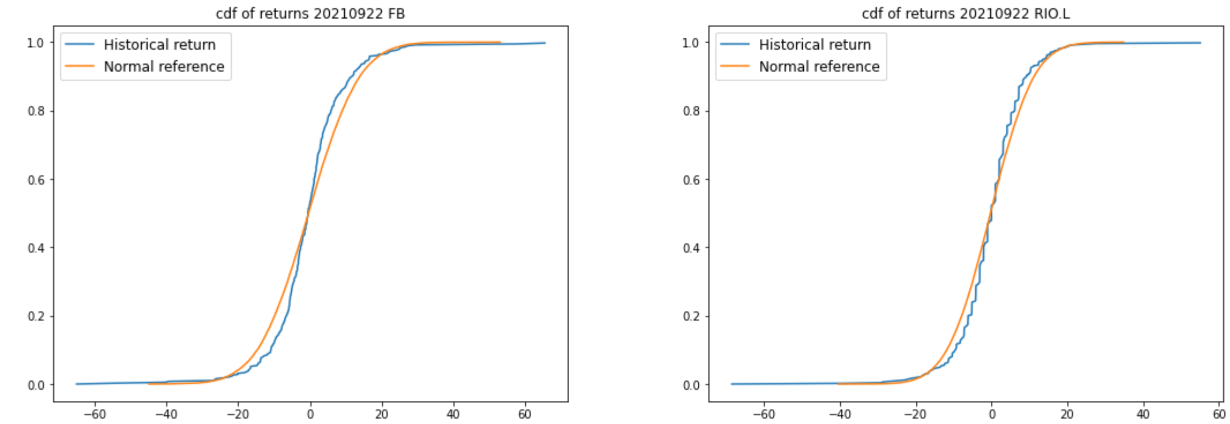
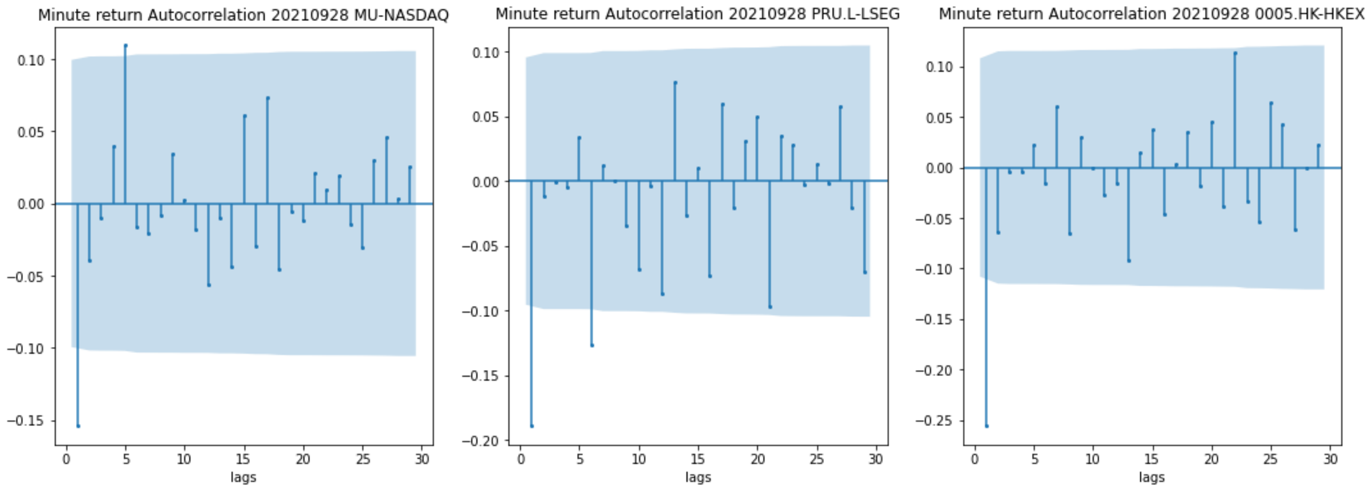


Figure 2: Autocorrelation patterns for returns of three stocks from the three exchanges.



3.2 Stylized facts and loss function

Financial price time series data display some interesting statistical characteristics that are commonly called stylized facts. According to Sewell (2011), stylized facts refer to empirical findings that are so consistent (for example, across a wide range of financial instruments and different time periods) that they are accepted as the truth. A stylized fact is a simplified presentation of an empirical finding in financial market. A successful and realistic financial market simulation is capable of reproducing various stylized facts. These stylized facts include fat-tailed distribution of returns, autocorrelation of returns, and volatility clustering. The loss function used in the calibration process is constructed by measuring the distance between historical and simulated stylized facts.

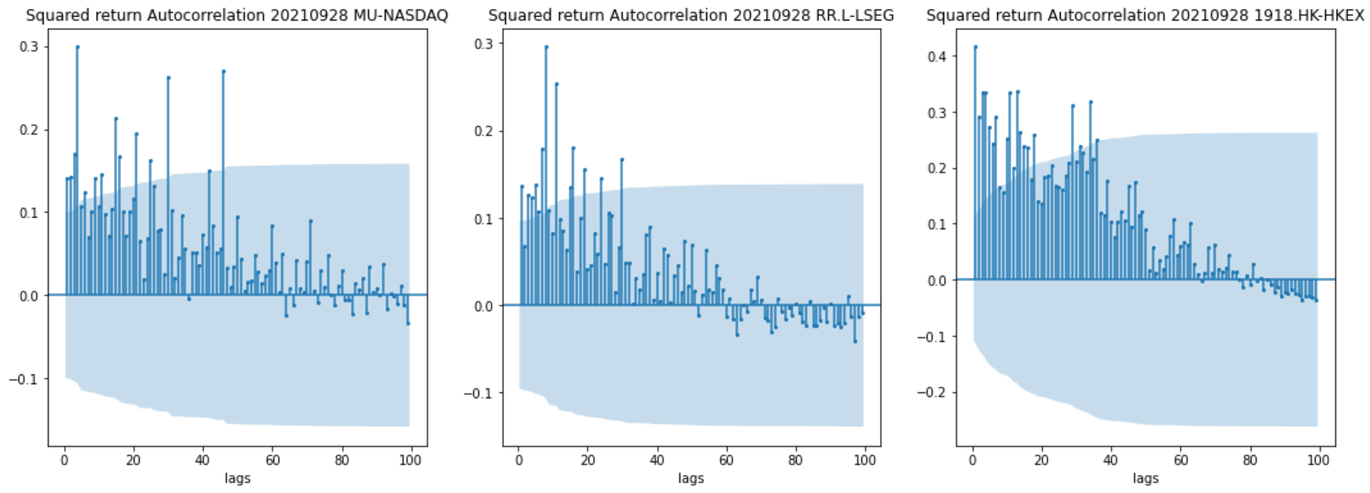
3.2.1 Fat-tailed distribution of returns

The distributions of price returns have been found to be fat-tailed across all timescales. In other words, the return distributions exhibit positive excess kurtosis. Understanding positively kurtotic return distributions is important for risk management, since large price movements are much more likely to occur than in commonly assumed normal distributions. In this paper we focus on intra-day minute price returns. Excess kurtosis of the distribution of minute-level returns is calculated for each stock at each trading day. Table 2 presents the average excess kurtosis for the intra-day minute-level return distributions of six randomly chosen stocks. For each stock, the return distribution has significantly positive excess kurtosis, proving the fat-tail characteristic of return distributions. Figure 1 also shows a comparison between cumulative distribution function of historical returns in two stocks for one trading day and that of a normal distribution with identical mean and standard deviation. Similar results are found in all other stocks from the three exchanges.

Table 2: Average excess kurtosis for return distributions of five stocks.

Stock	FB	AAPL	VOD.L	JET.L	9999.HK	0700.HK
Excess Kurtosis	5.51	5.64	16.20	9.60	9.49	4.66

Figure 3: Autocorrelation patterns for squared returns of three stocks from the three exchanges.



3.2.2 Autocorrelation of returns

Autocorrelation is defined to be a mathematical representation of the degree of similarity between a time series and a lagged version of the same time series. It measures the relationship between a variable's past value and its current value. Take first-order autocorrelation, for example. A positive first-order autocorrelation of returns indicates that a positive (negative) return in one period is prone to be followed by a positive (negative) return in the subsequent period. Instead, if the first-order autocorrelation of returns is negative, a positive (negative) return will usually be followed by a negative (positive) return in the next period. It is observed that the return series lack significant autocorrelation, except for weak, negative autocorrelation on very short timescales. McGroarty *et al.* (2019) show that the negative autocorrelation of returns is significantly stronger at a smaller time horizon and disappears at a longer time horizon. Examination of our data supports this stylized fact. Figure 2 shows the autocorrelation function of minute-level return time series for several stocks up to lag 30. We can see that the autocorrelation is significantly negative for small lags, and the negative autocorrelation gradually disappears for larger lags.

3.2.3 Volatility clustering

Financial price returns often exhibit the volatility clustering property: large changes in prices tend to be followed by large changes, while small changes in prices tend to be followed by small changes. This property results in persistence of the amplitudes of price changes (Cont, 2007). It is found that the volatility clustering property exists on timescales varying from minutes to weeks and months. Volatility clustering also refers to the long memory of square price returns (McGroarty *et al.*, 2019). Consequently, volatility clustering can be manifested by the slow decaying pattern in the autocorrelations of squared returns. Specifically, for short lags the autocorrelation function of squared returns is significantly positive, and the autocorrelation slowly decays with the lags increasing. Figure 3 shows the autocorrelation patterns for squared returns of several randomly chosen stocks from the three exchanges. Autocorrelation for squared returns of other stocks also exhibit similar patterns. It is shown that the volatility clustering stylized fact exists universally in financial markets.

3.2.4 Stylized facts distance as loss function

The target for agent-based model calibration is to find an optimal set of model parameters to make the model generate realistic simulated financial market. To solve this optimization problem, it is essential to have a metric that is able to quantify the "realism" of a simulated financial market. First of all, a realistic simulated financial market must exhibit similar characteristics to real financial market, such as the return distribution and volatility level. In addition, realistic simulated financial data are also required to reproduce other stylized facts such as the autocorrelation patterns in returns and squared returns. Here we design a stylized facts distance to quantify the similarities between simulated and historical financial data. The stylized facts distance is the weighted sum of four quantities: Kolmogorov-Smirnov statistic of return distributions, volatility difference, autocorrelation difference of returns and autocorrelation difference of squared returns:

$$D = w_1{}^*KS + w_2{}^*\Delta_V + w_3{}^*\Delta_{ACF^1} + w_4{}^*\Delta_{ACF^2} \tag{6}$$

Detailed calculations of the four quantities in the stylized facts distance are introduced below.

Kolmogorov-Smirnov statistic is a quantity from Kolmogorov-Smirnov test in statistics. Kolmogorov-Smirnov test is a non-parametric test of the equality of probability distributions that can be used to compare two samples. Here the two samples are simulated returns and historical returns, respectively. The Kolmogorov-Smirnov statistic quantifies a distance between the distribution functions of simulated returns and historical returns. Recall that the empirical distribution function is an estimate of the cumulative distribution function that generated the points in the sample. Let $F_s(x)$ and $F_h(x)$ denote the empirical distribution functions of simulated returns and historical returns, respectively. Then the Kolmogorov-Smirnov statistic is calculated as follows:

$$KS = \sup_x \left| F_s(x) - F_h(x) \right| \tag{7}$$

where $\sup_x$ is the supremum of the set of differences. Intuitively, the statistic represents the largest absolute difference between the two distribution functions across all

>

x values. The smaller the statistic is, the more similar are the simulated and historical returns. The inclusion of Kolmogorov-Smirnov statistic in the stylized facts distance addresses the requirements that the simulated data have similar return distribution to historical data and exhibit fat tails of returns.

The second part of the stylized facts distance is the absolute volatility difference between simulated returns and historical returns:

$$\Delta_V = \left|V_s - V_h\right| \tag{8}$$

where V_s and V_h denote simulated volatility and historical volatility, respectively. This part addresses the requirement that simulated financial market should be similar to real market in terms of volatility.

The third part of the stylized facts distance is the difference between simulated and historical autocorrelations of returns. This part in the stylized distance measures the model's ability of reproducing autocorrelation patterns commonly found in historical returns. It is shown that financial price return time series lack significant autocorrelation, except for short time scales, where significantly negative autocorrelations exists. This phenomenon is backed by our historical minute-level returns data. For very small lags the autocorrelations are negative, while for larger lags the autocorrelations become insignificant. To measure the distance in autocorrelation patterns between simulated data and historical data, we invoke the autocorrelation function of returns and calculate the average absolute difference between autocorrelations of simulated return time series and historical return time series for various lags:

$$\Delta_{ACF^1} = \frac{\sum_{l\ in\ lags} \left|ACF_s\left(l,r\right) - ACF_h\left(l,r\right)\right|}{\left|lags\right|} \tag{9}$$

where $ACF_s(l,r)$, $ACF_h(l,r)$ are the autocorrelation function of lag l for simulated returns and historical returns, respectively. $|lags|$ denotes the number of lags used in the calculation. Because the empirical autocorrelations are negative for very small lags and close to zero for larger lags, it is not necessary to consider all of the autocorrelation coefficients. Empirical evidence suggests that the autocorrelation pattern is well represented by the coefficients for three lags: 1, 10, 20. Also, to reduce the effects of accidental outliers, the autocorrelation function is smoothed by calculating the three-lag average. That is, the lag 1 autocorrelation is calculated as the average autocorrelation of lag 1, 2, 3, and so as the calculation for lag 10 and lag 20. In total, autocorrelations of 9 lags (1, 2, 3, 10, 11, 12, 20, 21, 22) are considered and included in the calculation.

The last part of the stylized facts distance is the difference between simulated and historical autocorrelations of squared returns. The replication of autocorrelation patterns in squared returns indicates the model's capability to reproduce the volatility clustering stylized fact. It is shown empirically that large price changes tend to be followed by other large price changes, known as the volatility clustering phenomenon. Consequently, though there are generally no significant patterns in autocorrelations of returns, the autocorrelations of squared returns are significantly positive, especially for small time lags. Also, as time lag increases, the autocorrelation of squared returns displays a slowly decaying pattern, as shown in Figure 3. Similar to the difference between autocorrelations of returns Δ_{ACF^1}, the difference between autocorrelations of squared returns is calculated as follows:

$$\Delta_{ACF^2} = \frac{\sum_{l\ in\ lags} \left|ACF_s\left(l,r^2\right) - ACF_h\left(l,r^2\right)\right|}{\left|lags\right|} \tag{10}$$

where $ACF_s(l,r^2)$, $ACF_h(l,r^2)$ are the autocorrelation function of lag l for simulated squared returns and historical squared returns, respectively. $|lags|$ denotes the number of lags used in the calculation. Unlike the case for autocorrelations of returns calculation, here we use a different $lags$. Because empirical autocorrelations of squared returns are significantly positive and slowly decaying, we consider the autocorrelations of squared returns from a minimal lag length of one minute up to a maximal lag length of 20 minutes. In total, the autocorrelations of 20 lags (1, 2, 3, ..., 18, 19, 20) are considered and included in the calculation.

The above four parts, along with the corresponding weights, constitute the stylized facts distance in Equation (6). In our experiments, there is no preference for any one of the four stylized facts. Thus, all the weights are equal to 1. Since it happens that the four quantities are of the same orders of magnitude, there is no need to adjust weights either. Also note that the stylized facts distance is a function of model parameters. In other words, given a set of model parameters, there is a unique stylized facts distance calculated from the simulated time series, which correspond to that particular set of model parameters. Let θ denote the vector of model parameters to be estimated, Equation (6) can be rewritten as:

$$D\left(\theta\right) = w_1{}^{*}KS\left(\theta\right) + w_2{}^{*}\Delta_V\left(\theta\right) + w_3{}^{*}\Delta_{ACF^1}\left(\theta\right) + w_4{}^{*}\Delta_{ACF^2}\left(\theta\right) \tag{11}$$

The smaller $D(\theta)$ is, the more realistic is the simulation. Thus, $D(\theta)$ serves as the loss function that the calibration method aims to minimize by finding an optimal set of model parameters. Let Θ denote the admissible set for model parameter vector θ, the calibration target is to find the optimal model parameter vector $\breve{\theta}$ that minimizes the stylized facts distance:

$$\breve{\theta} = \arg\min_{\theta\in\Theta} D\left(\theta\right) \tag{12}$$

The calibration method is presented in detail in subsequent sections.

3.3 Surrogate modelling calibration method

3.3.1 Specification

We propose a novel surrogate modeling approach to calibrate the model parameters. That is, to find an optimal set of model parameters to minimize the loss function—stylized facts distance. The approach is adapted from the machine learning surrogate modeling methodology in Lamperti *et al.* (2018). The original methodology mainly trains an XGBoost classifier to classify a positive calibration or negative calibration (Lamperti *et al.*, 2018), while in our approach we train an XGBoost regressor to directly approximate the mapping from model parameters to stylized facts distance. Another novelty in our approach is the introduction of exploration-exploitation mechanism in selecting new points in parameter space. The advantage of surrogate modeling approach is that it significantly reduces the computational cost of calibrating an ABM with several model parameters. Using only a limited budget (N) of ABM evaluations, the XGBoost surrogate model is proved to be a fairly good approximation of the mapping from model parameters to the target stylized facts distance. The surrogate provides a costless way to predict the model's response and allows for efficiently finding the optimal point in the model parameter space that minimizes the loss function.

With regard to the Extended Chiarella simulation model, during calibration the model can be represented as a mapping M: $\theta \rightarrow D(\theta)$ from a vector of model parameters θ into the stylized facts distance $D(\theta)$. Generally, the number of parameters ranges from several to dozens. In our case, we have three parameters to calibrate: κ, β, and σ_N. The values for parameters γ and α are fixed in advance. Parameter α indicates the typical horizon of trend computation for momentum traders. Following the original Extended Chiarella model (Majewski *et al.*, 2020), α is relatively small to represent a relatively low frequency momentum trader. Here we take the value 0.1 for α. Other values of α are also tested, and similar results are obtained as long as α is smaller than 0.4. For larger α value, the low frequency momentum traders will become high frequency momentum traders, which would be an extension of the Extended Chiarella model. We will present this extension in a separate paper. For γ, the value is fixed to be 10. Since γ appears in the same term as β in Equation (5), the model calibration will have difficulties pinning down the value of parameters γ and β if both parameters are to be estimated. As a result, fixing γ significantly improves the robustness of the model calibration process. We have also tested other values of γ and the results are similar.

Instead of classifying a positive calibration or negative calibration as in the original method (Lamperti *et al.*, 2018), here our calibration objective is obvious: finding an optimal set of values for κ, β, and σ_N to minimize the stylized facts distance. Our calibration measure is a real-valued number providing a quantitative assessment of the realism of the simulated market. An XGBoost machine learning surrogate model is trained to approximate the mapping from model parameters to stylized facts distance and help to find the optimal parameters. Detailed implementations are presented below.

3.3.2 Implementation

The whole calibration methodology proceeds with the following steps.

Step1. Initialization

The process starts with drawing a relatively large pool of parameter combinations. Each combination is a vector containing a value for each parameter: κ, β, and σ_N. The requirement is that the pool should be a good approximation of the whole parameter space. In our experiments, we use Sobol sampling (Morokoff and Caflisch, 1994) to implement the sampling routine. Sobol sampling is capable of guaranteeing uniformity of distribution even though the sampled set has a small number of points. It is shown that in terms of uniformity properties, Sobol sequences outperform the sequences generated by other sampling techniques such as Latin Hypercube sampling (Kucherenko *et al.*, 2015). Other advantages of Sobol sampling include efficient implementation and faster sampling speed. The sampling number and quality of the pool of parameter combinations dominate the ability of the whole process to learn a good surrogate model. As a result, faster sampler is preferred so that more samples can be obtained in limited computational time. In our experiments, we use Sobol sampling to sample 16,384 (2^{14}) points as the pool of parameter combinations.

After the pool of parameter combinations is obtained, an initial set of samples is chosen randomly from the pool as the initial training set. Each point in the set of initialization samples is evaluated by actually running the agent-based model with the corresponding parameter combination. The corresponding stylized facts distance is calculated, which will act as the true label associated with that point in the training set. After this step, we obtain an initial training set of parameter combinations with corresponding stylized facts distances as labels. The size of the initial training set in our experiments is 2,000.

Step 2. Surrogate model training

Given a training set of evaluated parameter combinations and corresponding stylized facts distances, an XGBoost regressor is learned over the training set in order to build the surrogate model. The input is the vector of model parameters to be calibrated, which in our experiments has dimension three. The output is the stylized facts distance, which is a scalar. The XGBoost regressor is trained to fit the mapping from model parameters to stylized facts distance. Implemented under the Gradient Boosting framework, XGBoost is a machine learning algorithm designed to be highly flexible, efficient, and portable (Chen and Guestrin, 2016). The XGBoost algorithm builds an ensemble of simple decision trees, which are subsequently aggregated to improve the prediction performance. Details on the XGBoost algorithm can be found in Chen and Guestrin (2016).

Remark 1. One difficulty in training the XGBoost regressor is how to tune hyper-parameters of the XGBoost algorithm. Here we employ the Bayesian Optimization method based on Gaussian Process (Snoek et al., 2012) to fine-tune the hyper-parameters of XGBoost. In the framework of Bayesian Optimization, performance of the XGBoost regressor is modeled as a sample from a Gaussian Process. The Gaussian Process then guides the exploration of the hyper-parameter space and helps to find an optimal set of hyper-parameters of XGBoost. Note that here the focus is the exploration of hyper-parameter space of the XGBoost machine learning algorithm, which is different from the parameter space of the agent-based model. Details on the Bayesian Optimization with Gaussian Process can be found in Snoek *et al.* (2012).

Remark 2. Another technique we have applied during the training process is to clip the values of training labels into a relatively small range. Specifically, for any training point, if the calculated stylized facts distance is too large, the distance value will be replaced with a relatively smaller value. The logic for this operation is that our calibration focus is the area of the model parameter space where stylized facts distances are small. In other words, parameter combinations with large stylized facts distances are of no interest and it is not important to have precise surrogate model approximations of stylized facts distances in those areas. If the range of distance value is not restricted, some large input labels would utterly bias the learning process of the XGBoost surrogate model. In these circumstances, the XGBoost surrogate model would wrongly try to fit those large outlier values. Consequently, the model is no longer a good approximation of the mapping from agent-based model parameters to stylized facts distance. By restricting the values of training labels to a relatively smaller range, the bias is successfully corrected, and experimental results show that the surrogate model predicts the stylized facts distances quite precisely. Since optimal stylized facts distances in our experiments are generally smaller than 0.8, we restrict the training labels to the range of $(0, 1]$.

Step 3. Surrogate model prediction

Once the surrogate XGBoost model is trained, it is used to predict the stylized facts distances over the set of remaining unlabeled parameter combinations. That is, the stylized facts distances that would be generated if those unlabeled parameter combinations were to be used in agent-based model simulation. The surrogate model predictions are used to guide further exploration of the model parameter space, as specified in the next step.

>

Step 4. Supplement training set

Given the XGBoost surrogate model predictions, a subset of the unlabeled parameter combinations is drawn from the pool sampled in the first step. This subset of the unlabeled parameter combinations is evaluated in the agent-based model simulation, and the true stylized facts distances are calculated and subsequently assigned as the true labels of these samples. This set of newly labeled points is then added to the training set of labeled parameter combinations. There are two critical issues in this process: How many new points to be drawn into the subset of the unlabeled parameter combinations and how to select the points. For the first issue, the original method recommends that the number of new points to be drawn in this stage is the logarithm of the computational budget (Lamperti *et al.*, 2018). However, in our experiments, it turns out that this rule would draw too few samples. Consequently, more iterations are required, which significantly reduce the computational efficiency. After lots of testing, in our method, around two percent of total samples are drawn at each iteration, which in our case is 300 points. As for how to choose those unlabeled parameter combinations, we utilize the predictions made by the XGBoost surrogate model. The unlabeled points are sorted according to the predicted stylized facts distances. The points with smaller predicted stylized facts distances are selected, as the optimal parameter combination is more likely to exist among or near those points. However, not all points are selected according to predicted stylized facts distances. We also randomly choose some points from the unlabeled parameter combinations to avoid occasional bias induced by the XGBoost surrogate model. Specifically, 200 points are selected according to the "small predicted stylized facts distance" principle and 100 points are selected randomly.

Remark 3. The way of selecting the subset of unlabeled points implements an exploration-exploitation mechanism, which is a novel aspect of our methodology compared to the original method. Around two-thirds of the points are selected according to the predicted stylized facts distances. In this way, we exploit the information given by the XGBoost surrogate model and the model intelligently helps to direct the selection of new samples. The reason is that the optimal parameter combination is more likely to exist among the points with smaller predicted distances. This is true as long as the surrogate is a fairly good approximation of the real mapping from parameter combinations to stylized facts distances. However, exploration is also essential if the aim is to find a global minimum. There might be multiple local minimums inside the model parameter space. Completely exploiting the information given by the surrogate model may get the method stuck in a local minimum. Randomly sampling some unlabeled points helps to avoid this problem and contributes to better exploration of the whole model parameter space.

Step 5. Iterations

After the training set is supplemented by the newly drawn samples and the corresponding labels, a new XGBoost surrogate model is trained using the new training set. The procedure is identical to previous steps. In other words, the previous "training-predicting-supplement" (step 2 to step 4) process is repeated until the budget of computational time is reached. In our experimental settings, we find that generally less than five iterations are required to build a fairly good surrogate model, whose stylized facts distance prediction error is less than five percent at the predicted optimal point.

4 Results and evaluation

The whole methodology is run on 75 stocks from three exchanges: Nasdaq, the London Stock Exchange, and the Hong Kong Stock Exchange. The main results are the stylized facts distances of the calibrated model, compared with the baseline stylized facts distances where model parameters are estimated by Expectation-Maximization algorithm. We also present the error rate of XGBoost surrogate model prediction for stylized facts distance at the predicted optimal point to evaluate the surrogate model prediction accuracy. Finally, we show the methodology's capability of reproducing autocorrelation patterns in return series, which is an advantage of XGB-Chiarella methodology over other traditional models.

4.1 Stylized facts distance

Table 3 shows the stylized facts distances of the calibrated model for all stocks, averaged by trading days. The corresponding standard deviations are also presented. For stocks in Hong Kong Stock Exchange, the XGB-Chiarella method outperforms EM estimation algorithm for all 25 stocks. As for stocks on Nasdaq, in 22 out of 25 stocks, the performance of the XGB-Chiarella method is better than the EM estimation method. On average, the XGB-Chiarella method achieves around 10 percent smaller stylized distances than the EM baseline in these two exchanges, showing that the simulated market is calibrated to be more realistic. The standard deviations of the stylized facts distance are also generally a bit smaller for the proposed XGB-Chiarella method in Nasdaq and the Hong Kong Stock Exchange. For stocks on the London Stock Exchange, the performance of the XGB-Chiarella method is not as good as the performance in the other two markets, with smaller advantage over the EM baseline estimation algorithm. However, it is still valid to say that the XGB-Chiarella methodology outperforms the baseline since smaller stylized facts distance is achieved in more than half of the stocks. Overall, the results show that the proposed the XGB-Chiarella methodology outperforms the EM baseline in terms of stylized facts distance and the realism of the market simulation.

Results in Table 3 are the mean and standard deviation of stylized facts distances across all trading days in our data span. To scrutinize each trading day for individual stock, Table 4 shows the number of trading days when the XGB-Chiarella methodology outperforms the EM baseline in terms of stylized facts distance, and the percentage of these trading days out of all trading days. In Table 4, the column "Better" represents the number of trading days when the performance of the XGB-Chiarella method is better than the baseline EM estimation, while column "Total" represents the number of all trading days in our experiments. For the performance in different exchanges, results here are similar to the results in Table 3. For most stocks on Nasdaq and the Hong Kong Stock Exchange, the XGB-Chiarella method performs better than the EM baseline for more than 85 percent of total trading days. As for stocks listed on the London Stock Exchange, this percentage is slightly lower, but is still on average more than 70 percent. On the whole, for most trading days, the XGB-Chiarella method is capable of creating a simulated financial market with smaller stylized facts distance, indicating that the proposed XGB-Chiarella method is able to generate more realistic artificial financial markets.

4.2 Surrogate model performance

To evaluate the performance of the surrogate model approximation, we present the prediction error rate when the surrogate model is used to predict the real stylized facts distance. Table 5 shows a comparison between the surrogate model predicted and the actual stylized facts distance at the surrogate predicted optimal point in model parameter space for one trading day. Very similar results are achieved for other trading days. The results show that the XGBoost surrogate model is a very accurate proxy for the true model around the predicted optimal point for each stock on each trading day, with predicted stylized facts distance very close to the actual stylized facts distance generated in agent-based model simulation. For most

Table 3: Average stylized facts distance comparison between XGB-Chiarella calibration and Expectation-Maximization method. The stylized facts distance is the average across all trading days for each stock; values in parentheses are corresponding standard deviations. There are several rare cases where distances are much larger, showing algorithms fail to obtain a reasonable set of parameters during the parameter estimation process.

NASDAQ	XGBChiarella	EMalgorithm	LSEG	XGBChiarella	EMalgorithm	HKEX	XGBChiarella	EMalgorithm
AAL	0.32 (0.07)	0.36 (0.09)	AAL.L	0.26 (0.17)	0.26 (0.09)	0005.HK	0.4 (0.04)	0.42 (0.05)
AAPL	0.25 (0.07)	0.28 (0.08)	AV.L	0.27 (0.05)	0.28 (0.07)	0027.HK	0.43 (0.07)	0.45 (0.07)
AFRM	0.28 (0.08)	0.34 (0.09)	BATS.L	0.24 (0.04)	0.25 (0.05)	0175.HK	0.43 (0.05)	0.45 (0.06)
AMD	0.29 (0.08)	0.32 (0.08)	BHP.L	0.27 (0.26)	0.26 (0.13)	0388.HK	0.35 (0.07)	0.37 (0.07)
CMCSA	0.32 (0.08)	0.36 (0.09)	BP.L	0.21 (0.06)	0.22 (0.05)	0700.HK	0.32 (0.06)	0.35 (0.06)
CSCO	0.28 (0.07)	0.3 (0.09)	BRBY.L	0.31 (0.07)	0.32 (0.07)	0836.HK	0.39 (0.07)	0.43 (0.09)
CSX	0.32 (0.07)	0.35 (0.08)	DGE.L	0.24 (0.06)	0.24 (0.07)	0916.HK	0.42 (0.08)	0.45 (0.08)
DKNG	0.31 (0.09)	0.36 (0.09)	EZJ.L	25.5 (182.0)	25.2 (179.6)	1024.HK	0.35 (0.08)	0.39 (0.08)
FB	0.28 (0.08)	0.31 (0.08)	GLEN.L	0.24 (0.07)	0.25 (0.1)	1171.HK	0.42 (0.07)	0.44 (0.08)
HBAN	0.39 (0.09)	0.36 (0.08)	GSK.L	0.24 (0.06)	0.25 (0.08)	1211.HK	0.37 (0.08)	0.41 (0.08)
HON	0.28 (0.09)	0.3 (0.1)	HSBA.L	0.23 (0.09)	0.27 (0.36)	1299.HK	0.4 (0.07)	0.43 (0.08)
HUT	0.27 (0.07)	29.5 (206.5)	IAG.L	26.7 (191.0)	25.7 (183.4)	1772.HK	0.39 (0.09)	0.44 (0.09)
INTC	0.28 (0.07)	0.31 (0.08)	JET.L	0.35 (0.27)	0.32 (0.08)	1918.HK	0.44 (0.08)	0.49 (0.1)
JD	0.34 (0.09)	0.38 (0.09)	LGEN.L	0.28 (0.05)	0.29 (0.08)	1919.HK	0.44 (0.07)	0.46 (0.08)
LCID	0.29 (0.07)	6.35 (42.4)	LLOY.L	0.23 (0.06)	0.24 (0.07)	2020.HK	0.35 (0.09)	0.39 (0.08)
MRNA	0.32 (0.09)	0.35 (0.09)	NG.L	0.25 (0.06)	0.25 (0.07)	2269.HK	0.37 (0.08)	0.41 (0.09)
MSFT	0.26 (0.07)	0.29 (0.08)	PRU.L	0.31 (0.07)	0.33 (0.06)	2318.HK	0.37 (0.07)	0.4 (0.07)
MU	0.34 (0.09)	0.39 (0.1)	RDSA.L	0.3 (0.57)	0.27 (0.23)	2331.HK	0.38 (0.07)	0.42 (0.09)
NVDA	0.27 (0.08)	0.3 (0.07)	REL.L	0.35 (0.05)	0.37 (0.05)	2333.HK	0.4 (0.07)	0.44 (0.08)
SOFI	0.3 (0.08)	0.36 (0.1)	RIO.L	0.27 (0.24)	0.26 (0.11)	3690.HK	0.34 (0.08)	0.38 (0.07)
TLRY	0.29 (0.08)	0.34 (0.09)	RR.L	0.25 (0.08)	0.28 (0.09)	3968.HK	0.39 (0.06)	0.42 (0.07)
TSLA	0.36 (0.2)	0.28 (0.08)	SMT.L	0.23 (0.05)	0.26 (0.05)	9618.HK	0.35 (0.07)	0.38 (0.07)
UAL	0.33 (0.07)	0.37 (0.09)	STAN.L	0.27 (0.05)	0.27 (0.06)	9888.HK	0.38 (0.09)	0.4 (0.07)
UBER	0.29 (0.07)	0.33 (0.08)	ULVR.L	0.23 (0.08)	0.23 (0.07)	9988.HK	0.33 (0.06)	0.36 (0.07)
WISH	0.38 (0.14)	0.29 (0.07)	VOD.L	0.24 (0.07)	0.25 (0.12)	9999.HK	0.36 (0.07)	0.4 (0.07)

stocks, errors between the predicted distance and the actual distance are less than 10 percent. It is also shown that the accuracy of the XGBoost surrogate model prediction is similar across the three exchanges, which indicates the robustness of the XGB-Chiarella methodology.

Apart from the prediction accuracy at the predicted optimal point, we also examine the prediction accuracy around the optimal point as a sensitivity analysis. Figure 4 shows the comparison between the XGBoost surrogate model prediction and the actual simulated stylized facts value. For each sub-graph, one model parameter is assigned a series of values across the given range during calibration. Other model parameters are fixed to the optimal value. In this way a set of parameter combinations is obtained, with different values for that particular model parameter. Then, predicted and actual stylized facts distance are calculated and compared, as shown in Figure 4. The gray lines are the actual stylized facts distances, while the blue lines represent the XGBoost surrogate model prediction for the stylized facts distances. It is shown that the XGBoost prediction line fits the actual simulated stylized facts distance line quite well, especially for parameter “sigma N”. For each parameter, there is a unique minimal value in the graph, which corresponds to the optimal point predicted by the surrogate model. The results here indicate that the surrogate model approximates the real stylized facts distance quite accurately and is capable of finding the true optimal model parameter combination that is able to generate realistic financial market simulations.

4.3 Autocorrelations

One unique advantage of agent-based financial market simulation is the capability of reproducing autocorrelation patterns of returns. Currently, most literature in financial econometrics model financial price series has it as Geometric Brownian Motion (GBM). For example, the Black-Scholes model for option pricing explicitly assumes the stock price follows a GBM. However, the case is not true in real financial market. One obvious anomaly is the autocorrelation patterns in return series, such as the volatility clustering phenomenon shown in Figure 3. Our XGB-Chiarella method has an advantage over the GBM model, in that, it can successfully reproduce the autocorrelation patterns in financial return time series. Figure 5

>

Table 4: Percentage of trading days when XGB-Chiarella calibration outperforms the Expectation-Maximization method.

NASDAQ	Better	Total	Percentage	LSEG	Better	Total	Percentage	HKEX	Better	Total	Percentage
AAL	46	50	92.00%	AAL.L	43	52	82.69%	0005.HK	37	47	78.72%
AAPL	41	50	82.00%	AV.L	37	52	71.15%	0027.HK	37	47	78.72%
AFRM	46	50	92.00%	BATS.L	38	52	73.08%	0175.HK	34	47	72.34%
AMD	39	50	78.00%	BHP.L	39	52	75.00%	0388.HK	39	47	82.98%
CMCSA	46	50	92.00%	BP.L	40	52	76.92%	0700.HK	46	47	97.87%
CSCO	42	50	84.00%	BRBY.L	48	51	94.12%	0836.HK	45	47	95.74%
CSX	44	50	88.00%	DGE.L	33	52	63.46%	0916.HK	41	47	87.23%
DKNG	47	50	94.00%	EZJ.L	49	52	94.23%	1024.HK	45	47	95.74%
FB	41	50	82.00%	GLEN.L	34	52	65.38%	1171.HK	38	47	80.85%
HBAN	14	50	28.00%	GSK.L	30	52	57.69%	1211.HK	45	47	95.74%
HON	44	50	88.00%	HSBA.L	31	52	59.62%	1299.HK	42	47	89.36%
HUT	44	50	88.00%	IAG.L	39	52	75.00%	1772.HK	45	47	95.74%
INTC	44	50	88.00%	JET.L	37	52	71.15%	1918.HK	39	47	82.98%
JD	44	50	88.00%	LGEN.L	37	52	71.15%	1919.HK	40	47	85.11%
LCID	47	50	94.00%	LLOY.L	33	52	63.46%	2020.HK	42	47	89.36%
MRNA	47	50	94.00%	NG.L	31	52	59.62%	2269.HK	42	47	89.36%
MSFT	42	50	84.00%	PRU.L	45	52	86.54%	2318.HK	43	47	91.49%
MU	45	50	90.00%	RDSA.L	36	52	69.23%	2331.HK	46	47	97.87%
NVDA	43	50	86.00%	REL.L	44	51	86.27%	2333.HK	42	47	89.36%
SOFI	44	50	88.00%	RIO.L	38	52	73.08%	3690.HK	45	47	95.74%
TLRY	45	50	90.00%	RR.L	40	52	76.92%	3968.HK	43	47	91.49%
TSLA	19	50	38.00%	SMT.L	45	52	86.54%	9618.HK	44	47	93.62%
UAL	46	50	92.00%	STAN.L	38	52	73.08%	9888.HK	43	47	91.49%
UBER	47	50	94.00%	ULVR.L	29	52	55.77%	9988.HK	44	47	93.62%
WISH	14	50	28.00%	VOD.L	29	52	55.77%	9999.HK	44	47	93.62%

Figure 4. Sensitivity analysis around the optimal point predicted by the XGBoost surrogate model. This surrogate model corresponds to the trading day on September 29, 2021, for stock FB. For each sub-graph, the model parameters are fixed to the optimal value, except for the parameter in the x-axis. The gray lines represent the actual stylized facts distances while the blue lines represent the XGBoost surrogate model prediction for stylized facts distances.

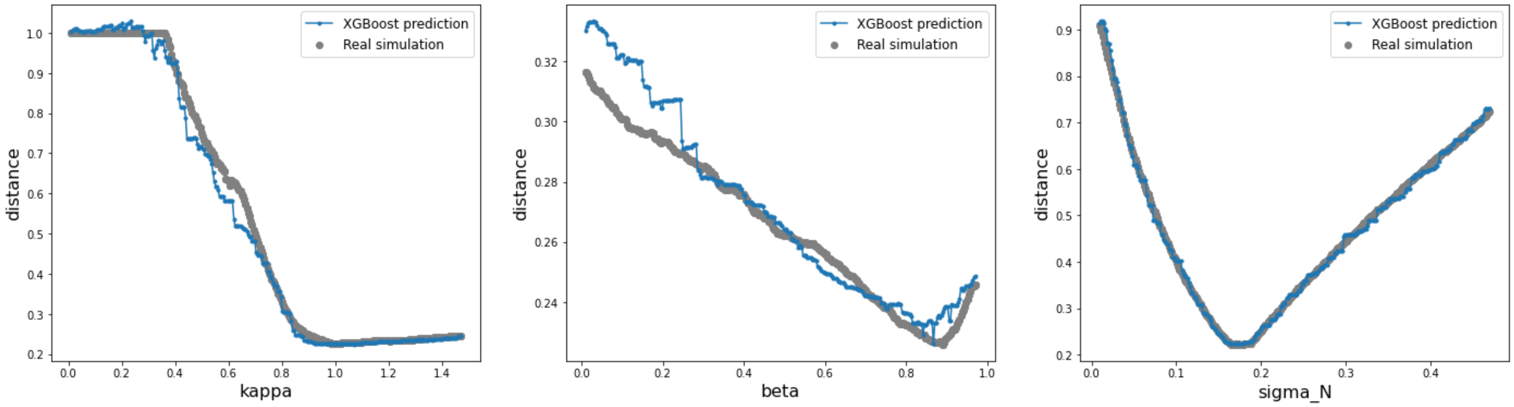
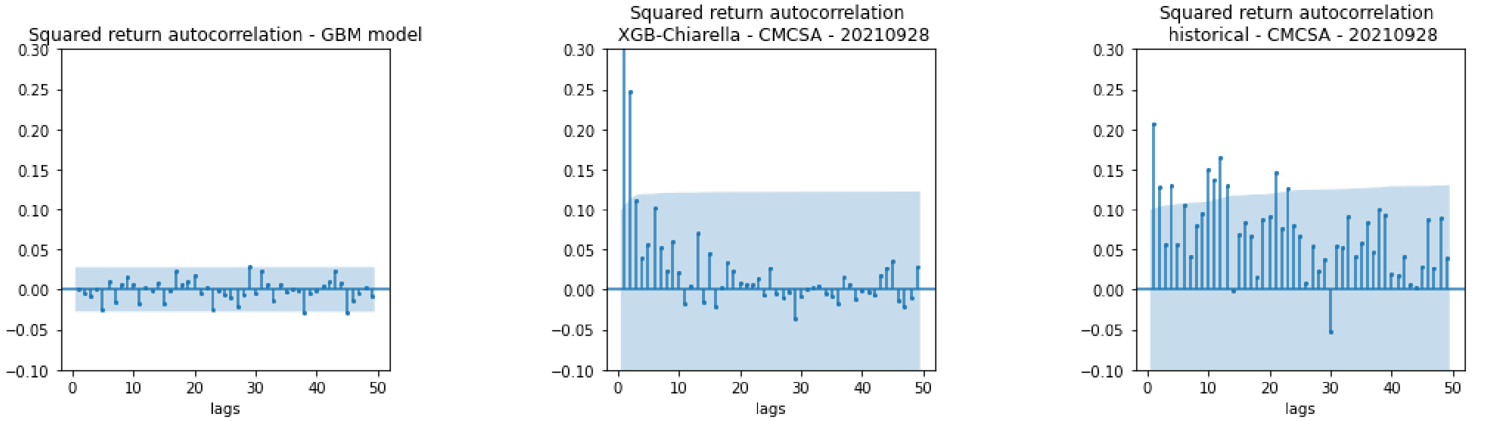


Table 5: Comparison between the surrogate model predicted and the actual stylized facts distance at the surrogate predicted optimal point. The results shown here are for one trading day (November 2, 2021), while results for other trading days are very similar.

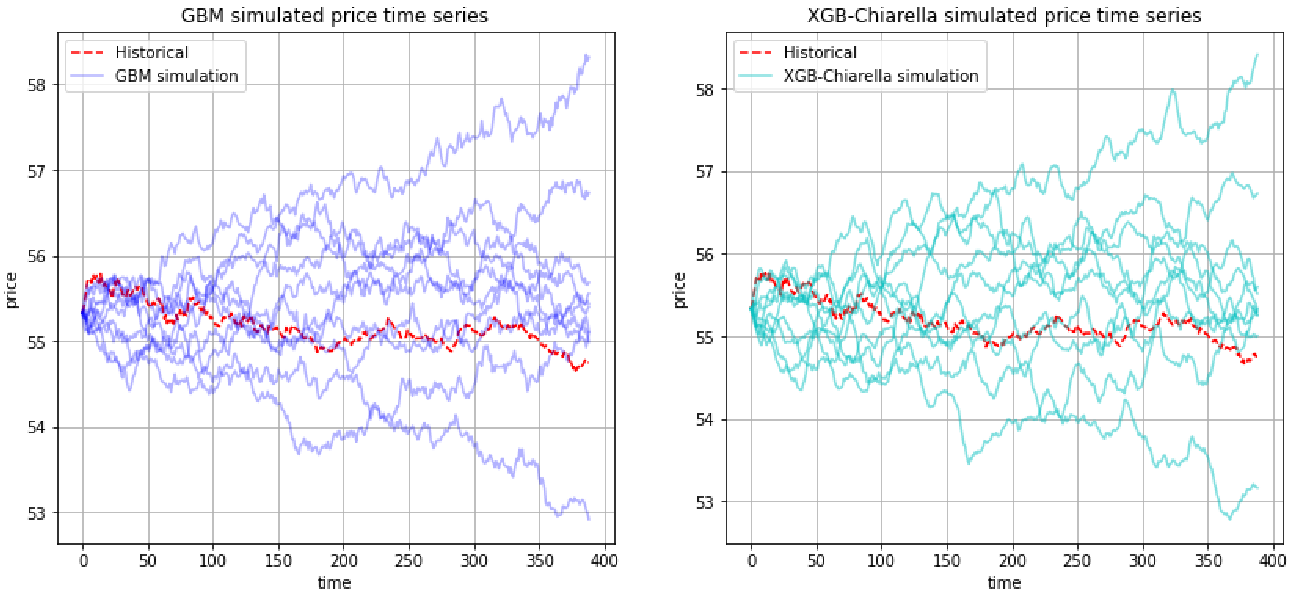
NASDAQ	Predict	Actual	Error	LSEG	Predict	Actual	Error	HKEX	Predict	Actual	Error
AAL	0.33	0.37	10.81%	AAL.L	0.28	0.27	3.70%	0005.HK	0.39	0.42	7.14%
AAPL	0.25	0.27	7.41%	AV.L	0.35	0.36	2.78%	0027.HK	0.42	0.43	2.33%
AFRM	0.48	0.49	2.04%	BATS.L	0.25	0.24	4.17%	0175.HK	0.37	0.4	7.50%
AMD	0.39	0.4	2.50%	BHP.L	0.26	0.26	0.00%	0388.HK	0.36	0.36	0.00%
CMCSA	0.45	0.46	2.17%	BP.L	0.16	0.16	0.00%	0700.HK	0.27	0.26	3.85%
CSCO	0.39	0.4	2.50%	BRBY.L	0.3	0.29	3.45%	0836.HK	0.29	0.31	6.45%
CSX	0.31	0.31	0.00%	DGE.L	0.17	0.18	5.56%	0916.HK	0.53	0.55	3.64%
DKNG	0.26	0.27	3.70%	EZJ.L	0.22	0.21	4.76%	1024.HK	0.28	0.26	7.69%
FB	0.27	0.27	0.00%	GLEN.L	0.27	0.27	0.00%	1171.HK	0.37	0.36	2.78%
HBAN	0.4	0.38	5.26%	GSK.L	0.4	0.39	2.56%	1211.HK	0.26	0.26	0.00%
HON	0.25	0.25	0.00%	HSBA.L	0.18	0.19	5.26%	1299.HK	0.33	0.35	5.71%
HUT	0.3	0.31	3.23%	IAG.L	0.26	0.27	3.70%	1772.HK	0.35	0.35	0.00%
INTC	0.36	0.35	2.86%	JET.L	0.37	0.37	0.00%	1918.HK	0.3	0.34	11.76%
JD	0.33	0.33	0.00%	LGEN.L	0.25	0.25	0.00%	1919.HK	0.33	0.36	8.33%
LCID	0.28	0.26	7.69%	LLOY.L	0.25	0.27	7.41%	2020.HK	0.35	0.35	0.00%
MRNA	0.32	0.33	3.03%	NG.L	0.2	0.2	0.00%	2269.HK	0.32	0.34	5.88%
MSFT	0.29	0.29	0.00%	PRU.L	0.33	0.35	5.71%	2318.HK	0.31	0.31	0.00%
MU	0.42	0.43	2.33%	RDSA.L	0.23	0.23	0.00%	2331.HK	0.29	0.28	3.57%
NVDA	0.2	0.2	0.00%	REL.L	0.3	0.29	3.45%	2333.HK	0.32	0.35	8.57%
SOFI	0.32	0.32	0.00%	RIO.L	0.25	0.25	0.00%	3690.HK	0.25	0.23	8.70%
TLRY	0.27	0.28	3.57%	RR.L	0.27	0.32	15.63%	3968.HK	0.29	0.31	6.45%
TSLA	0.44	0.44	0.00%	SMT.L	0.21	0.2	5.00%	9618.HK	0.25	0.24	4.17%
UAL	0.24	0.24	0.00%	STAN.L	0.23	0.23	0.00%	9888.HK	0.34	0.33	3.03%
UBER	0.27	0.27	0.00%	ULVR.L	0.21	0.21	0.00%	9988.HK	0.28	0.28	0.00%
WISH	0.52	0.48	8.33%	VOD.L	0.15	0.21	28.57%	9999.HK	0.37	0.37	0.00%

Figure 5: (a) presents the autocorrelation of squared returns generated by a GBM; (b) is the autocorrelation of squared returns generated by the XGB-Chiarella method for CMCSA on September 28, 2021; and (c) is the historical autocorrelation of squared returns for CMCSA on September 28, 2021.



(a) GBM **(b) XGB-Chiarella** **(c) Historical**

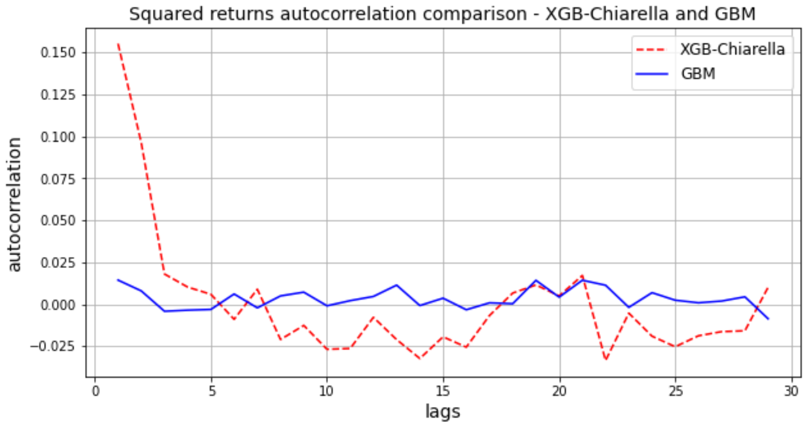
Figure 6: Simulated price scenarios generated by GBM and XGB-Chiarella.



shows autocorrelations of squared returns of a GBM model and that of a simulation under the XGB-Chiarella methodology for one stock. The real historical autocorrelation of squared returns on the same day for that stock is also presented. It is obvious that our XGB-Chiarella method outperforms the traditional GBM model in terms of the replication of realistic autocorrelation patterns, especially for small time lags. For small time lags, significant positive autocorrelations of squared returns exist in historical financial market data. This stylized fact is successfully reproduced in XGB-Chiarella simulation, while the autocorrelations of squared returns generated by a GBM is basically close to zero, regardless of time lags. In fact, there is no specific patterns in autocorrelations generated by the GBM model. Notice that simulated results in other stocks for other trading days are similar to what is shown here.

Successful reproduction of autocorrelation patterns gives the XGB-Chiarella method an advantage in realistic simulation of multiple price series, which can be used in scenario simulations for risk management. When calculating simulation-based risk metrics, such as value at risk (VaR), lots of institutions are still using GBM to simulate future scenarios. As shown in Figure 5, the return series simulated by the GBM model lack realistic autocorrelation pattern, which would undermine the credibility of the risk metric calculation. Our proposed method is a better alternative. The proposed XGB-Chiarella method extracts fundamental values and calibrate the model parameters using historical data. With the same set of calibrated model parameters, the agent-based model is capable of generating multiple different scenarios by changing the input fundamental value series. For example, if GBM scenarios are used as fundamental value, the agent-based model is able to reproduce similar price series as the GBM model, but with realistic autocorrelation patterns of returns. Figure 6 shows the price scenarios generated by the GBM and XGB-Chiarella methods, respectively. Here the fundamental values in the XGB-Chiarella model are the price scenarios generated by the GBM. Figure 7 shows the corresponding average autocorrelation of squared returns for the two cases. It is shown that the two models can generate similar price scenarios, but the XGB-Chiarella model is able to reproduce much more realistic autocorrelation patterns. Specifically, for small time lags, the XGB-Chiarella model reproduces significant positive autocorrelation of squared returns, which is consistent with empirical data. In contrast, there are no significant autocorrelation patterns for GBM scenarios. It is believed that such agent-based price series simulation could provide a richer environment than the GBM model for risk management practice such as VaR calculation. Since the focus of this paper is on intra-day price formation process, we will address this topic in a separate paper in the future.

Figure 7: Average autocorrelation of squared returns associated with the simulated price scenarios in Figure 6. Blue line represents the GBM simulated scenarios, while the red dashed line represents the XGB-Chiarella simulated scenarios.



5 Conclusion and future work

5.1 Summary of achievements

In this paper, a new approach called XGB-Chiarella is proposed to generate realistic intra-day artificial financial price data in order to provide insight into the intra-day price formation process. To the best of our knowledge, this is the first extension of the Chiarella model to generate minute-level intra-day financial market simulation. The approach utilizes agent-based modeling techniques. The underlying simulation model has only three agents: one for fundamental trader, one for momentum trader, and one for noise trader. The model is simulated, and model parameters are calibrated by an XGBoost machine learning surrogate. The proposed methodology is tested on 75 stocks from three exchanges: Nasdaq, the London Stock Exchange, and the Hong Kong Stock Exchange. In terms of stylized facts distance, the proposed XGB-Chiarella method is able to generate more realistic financial market simulations than the original Expectation-Maximization estimation algorithm. This is true in nearly all the stocks from the three exchanges. Despite the fact that the methodology is based on a model with only three agents, the XGB-Chiarella methodology successfully generates very realistic financial market simulations. This indicates that one agent per category seems to be sufficient to capture the intra-day price formation process for the time scale (minutes) chosen in this paper. The very simple model structure not only accelerates the simulation process in terms of computational cost, but also enables us to scrutinize the intra-day price formation process, such as the trend and value effects. The results provide support for the existence of a universal intra-day price formation mechanism. The realistic simulated intra-day financial market indicates that trend and value effects, as well as noise trading, are indispensable to the intra-day price formation process.

We also show that in the process of calibration, the XGBoost surrogate model is an accurate approximation of the true model. At the predicted optimal point for each stock on each trading day, the surrogate model prediction error is mostly smaller than 10 percent. The machine learning surrogate is capable of intelligently directing the exploration of model parameter space. The exploitation-exploration mechanism is also introduced in the model calibration process. A practical application of the proposed methodology is also presented.

5.2 Future work

This work can be extended in several aspects. Firstly, in modern financial markets a large number of transactions can happen in fractions of a second, raising interest in the price formation process at higher frequency. Therefore, it would be interesting to test whether the proposed methodology would work at higher frequency, for example, in microseconds or even nanoseconds level. Another interesting extension is about agent heterogeneity. For example, the momentum traders can be divided into two groups: one group of traders that focuses on lower frequency price momentum and the other group that acts according to the value of higher frequency price momentum. It is expected that the introduction of further agent heterogeneity would improve the realism of the model since real-world traders are obviously heterogeneous. In addition, it would also be interesting to understand the price behaviors if we relax the assumption of λ-approximation in the underlying Chiarella model. With the λ-approximation, it is assumed that price change is linearly proportional to the cumulative demand of all traders. One extension to relax this assumption is to introduce full exchange protocols and limit order books to the simulated financial market and investigate the corresponding market dynamics. In this circumstance, hundreds of agents can be included in the model and interact with one another through limit order books, which is exactly the mechanism existing in real financial markets. The last aspects of future work involve extending the Chiarella model to multiple stocks. For example, how to simulate the correlation structure across multiple stocks and how to create correlated demands for related stocks during simulation.

Acknowledgements

Support of the UK EPSRC (Grant Nos. EP/L016796/1, EP/N031768/1, EP/P010040/1, EP/S030069/1, and EP/V028251/1), Xilinx, and Intel is gratefully acknowledged. Kang Gao holds a China Scholarship Council-Imperial Scholarship. The authors thank Kaveh Aasaraai and Javier A. Varela for their excellent suggestions.

Kang Gao is currently a PhD candidate in the Department of Computing at Imperial College London. His doctoral research covers the modeling and simulation techniques in financial services. He takes a multidisciplinary approach that encompasses the fields of artificial intelligence, agent-based modeling, and financial mathematics. He has a master's degree in machine learning from Imperial College London.

Perukrishnen Vytelingum is the Head of Quantitative Modeling at Simudyne, a startup specializing in agent-based modeling. His latest research focuses on market and credit risk modeling, market simulation, and agent-based modeling. He has over 10 years of experience in investment banking and was previously a senior research fellow at the University of Southampton. He has a master's degree in information systems engineering from Imperial College London and a PhD in agent-based modeling from the University of Southampton.

Wayne Luk is Professor of Computer Engineering at Imperial College London and Director of the EPSRC Centre for Doctoral Training in High Performance Embedded and Distributed Systems. His research focuses on theory and practice of customizing hardware and software for specific application domains, such as computational finance, climate modeling, and genomic data analysis. He pioneered the optimization of parametric hardware descriptions of neural network designs and the acceleration of financial simulation targeting field-programmable technology. He is a fellow of the Royal Academy of Engineering, IEEE, and BCS.

Stephen Weston is a Partner in the Risk Advisory practice at Deloitte, focusing on valuation and risk modeling, as well as AI. He has over 25 years of experience in investment banking, working at many of the largest investment banks, as well as hedge funds and start-ups. Immediately prior to joining Deloitte, he spent four years at Intel, focusing on modeling, fintech and HFT. His experience spans all areas of trading, risk management, and quantitative research. In addition, he is also a visiting Professor at Imperial College London in computational finance. Stephen has a PhD in mathematical finance from the London University. He also has a particular penchant for red trousers and bright socks.

Ce Guo is a Research Associate in custom computing and mass spectrometry in the Department of Computing and Department of Physics at Imperial College London. His research focuses on efficient computing systems for large-scale temporal data analytics, agent-based modeling, causal structural learning, and mass spectrometry data mining. He has a master's degree in artificial intelligence and a PhD in custom computing from Imperial College London.

>

Endnote

[1] XGBoost (Chen and Guestrin, 2016) is a highly flexible and efficient machine learning algorithm based on an ensemble of decision trees.

References

Beja, A. and Goldman, B. 1980. On the dynamic behavior of prices in disequilibrium. *The Journal of Finance*, 35(2). pp 235–248.

Bianchi, C., Cirillo, P., Gallegati, M., and Vagliasindi, P. 2007. Validating and calibrating agent-based models: A case study. *Computational Economics*, 30(3). pp 245–264.

Byron, M. Y., Shenoy, K. V., and Sahani, M. 2004. Derivation of kalman filtering and smoothing equations. *Technical report*, Stanford University.

Cason, T. N. and Friedman, D. 1996. Price formation in double auction markets. *Journal of Economic Dynamics and Control*, 20(8). pp 1307–1337.

Chen, T. and Guestrin, C. 2016. XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp 785–794.

Chiarella, C. 1992. The dynamics of speculative behaviour. *Annals of Operations Research*, 37(1). pp 101–123.

Cont, R. 2007. Volatility clustering in financial markets: Empirical facts and agent-based models. *Long memory in economics*. pp 289–309. Springer.

Dosi, G., Pereira, M., and Virgillito, M. E. 2018. On the robustness of the fat-tailed distribution of firm growth rates: A global sensitivity analysis. *Journal of Economic Interaction and Coordination*, 13(1). pp 173–193.

Faias, M., Herves-Beloso, C., and Moreno-Garcia, E. 2011. Equilibrium price formation in markets with differentially informed agents. *Economic Theory*, 48(1). pp 205–218.

Franke, R. 2009. Applying the method of simulated moments to estimate a small agent-based asset pricing model. *Journal of Empirical Finance*, 16(5). pp 804–815.

Franke, R. and Westerhoff, F. 2012. Structural stochastic volatility in asset pricing dynamics: Estimation and model contest. *Journal of Economic Dynamics and Control*, 36(8). pp 1193–1211.

Frankel, J. and Froot, K. 1986. Understanding the US dollar in the eighties: The expectations of chartists and fundamentalists. *Economic record*, 62(1). pp 24–38.

Gerety, M. S. and Mulherin, J. H. 1994. Price formation on stock exchanges: The evolution of trading within the day. *Review of Financial Studies*, 7(3). pp 609–629.

Grazzini, J. and Richiardi, M. 2015. Estimation of ergodic agent-based models by simulated minimum distance. *Journal of Economic Dynamics and Control*, 51. pp 148–165.

Jackson, M. O. 1991. Equilibrium, price formation, and the value of private information. *The Review of Financial Studies*, 4(1). pp 1–16.

Kucherenko, S., Albrecht, D., and Saltelli, A. 2015. Exploring multi-dimensional spaces: A comparison of Latin Hypercube and Quasi Monte Carlo Sampling Techniques. *arXiv preprint. arXiv:1505.02350*.

Kyle, A. 1985. Continuous auctions and insider trading. Econometrica: *Journal of the Econometric Society*. pp 1315–1335.

Lamperti, F., Roventini, A., and Sani, A. 2018. Agent-based model calibration using machine learning surrogates. *Journal of Economic Dynamics and Control*, 90. pp 366–389.

LeBaron, B. 2006. Agent-based computational finance. *Handbook of Computational Economics*, 2. pp 1187–1233.

Lo, A. W. 2017. *Adaptive Markets: Financial Evolution at the Speed of Thought*. Princeton University Press.

Majewski, A., Ciliberti, S., and Bouchaud, J.-P. 2020. Co-existence of Trend and Value in Financial Markets: Estimating an Extended Chiarella Model. *Journal of Economic Dynamics and Control*, 112. pp 103791.

McGroarty, F., Booth, A., Gerding, E., and Chinthalapati, V. R. 2019. High frequency trading strategies, market fragility and price spikes: An agent-based model perspective. *Annals of Operations Research*, 282(1). pp 217–244.

Morokoff, W. J. and Caflisch, R. E. 1994. Quasi-random sequences and their discrepancies. *SIAM Journal on Scientific Computing*, 15(6). pp 1251–1279.

Ralaivola, L. and d'Alche Buc, F. 2005. Time series filtering, smoothing and learning using the kernel Kalman filter. *Proceedings of 2005 IEEE International Joint Conference on Neural Networks*, 3. pp 1449–1454.

Rasmussen, C. E. 2003. Gaussian processes in machine learning. *Summer school on machine learning*. pp 63–71. Springer.

Sewell, M. 2011. Characterization of financial time series. Research Note, *Rn*/11/01. University College London.

Sirignano, J. and Cont, R. 2019. Universal features of price formation in financial markets: Perspectives from deep learning. *Quantitative Finance*, 19(9). pp 1449–1459.

Snoek, J., Larochelle, H., and Adams, R. P. 2012. Practical Bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25.

Zeeman, E. 1974. On the unstable behaviour of stock exchanges. *Journal of Mathematical Economics*, 1(1). pp 39–49.