Many learning agents interacting with an agent-based market model

Matthew Dicks[a], Andrew Paskaramoorthy[a], Tim Gebbie[a]

[a]*Department of Statistical Sciences, University of Cape Town, Rondebosch 7701, South Africa*

Abstract

We consider the dynamics and the interactions of multiple reinforcement learning optimal execution trading agents interacting with a reactive agent-based model of a financial market in event time for a single traded instrument. The model represents a market ecology with 3-trophic levels represented by: optimal execution learning agents, minimally intelligent liquidity takers, and fast electronic liquidity providers. The liquidity takers cannot learn to switch strategies. Learning is restricted to an optimal execution agent class that includes buying and selling agents. These agents can use combinations of limit orders and market orders. Their reward function explicitly balances trade execution slippage against the penalty of not executing the order timeously. This demonstrates how multiple competing learning agents impact a minimally intelligent market simulation as functions of the number of agents, the size of agents' initial orders, and the state spaces used for learning. Phase space plots are used to examine the dynamics of the various model configurations. Including learning agents in chartist-fundamentalist-noise models for a single traded instrument improves conformity with the empirical stylised facts but is insufficient to recover the complexity observed in empirical data.

Keywords: agent-based model, reinforcement learning, price impact, complex systems

1. Introduction

At high-frequency time scales, we can investigate how prices change at the level of individual trades revealing several remarkable micro-structural stylised facts [41, 28, 14, 4, 3, 45, 10, 8]. The most important of these stylised facts include: the long-memory of order flows and absolute returns, the distribution of rare events, and the power-law of price impact. Taken together, these stylised facts motivate a model for price dynamics based on order flow and liquidity provision, arising from the strategic behaviour of different classes of heterogeneous agents, operating at different time scales and under asymmetric information.

These agents conceal private information to prevent adverse selection, which limits the available liquidity and has given rise to the notion of latent supply and demand for liquidity not reflected in the visible order-book. These micro-structural stylised facts are plausibly thought to arise from the interaction of latent supply and demand with revealed liquidity, facilitated by optimal execution agents. However, by definition, data pertaining to latent demand and supply is not publicly available — it is hidden.

Agent-Based Models (ABMs) have been extensively used to explain subsets of various low-frequency stylised facts arising from the interactions of heterogeneous agents; most commonly chartists and fundamentalists within a minority game setting[11, 12, 38]. However, some micro-structural stylised facts can be directly attributed to the behaviour of optimal execution agents and isolating these features has received far less attention within the ABM literature. The execution of any parent order arising from latent demand incurs trading costs in the form of price impact. Optimal execution agents will try to minimise this impact, which arises from limited liquidity and incomplete information, through the use of strategic order-splitting.

Here, a large parent order is split into smaller child orders to be sequentially executed. Empirically, order-splitting appears to be widespread in real markets, and is used to explain the observed persistence of order flow and the long-memory of the size of returns. Furthermore, it is now well-established that realised price impact is concave with a power-law relationship, which has been attributed to the transfer of latent liquidity to the visible limit order book in response to a change in price [18, 22]. In light of this, how should optimal execution agents be defined in order to minimise costs and simultaneously reproduce these stylised facts?

Firstly, it appears that realistic execution agents must also be able to use limit orders (LOs) to execute parent orders, to facilitate the transfer of latent demand and supply to the limit order book, and hence recover the concave price impact function. Next, we can distinguish between static and dynamic execution strategies. Markets are not automatically efficient in the sense of price predictability,

suggesting there is some form of tâtonnement.

Static strategies typically require market efficiency or asymmetric information access to be optimal, *e.g.* consistency with linear price impact [35, 31, 22]. In contrast, the observed concavity of price impact [3, 26] provides both a normative and a positive argument for dynamic execution strategies that are aware of market conditions. Since a correct *a priori* specification of the data-generating process is unrealistic, it seems that some form of learning is necessary for dynamic strategies to be operational in practice. However, in the presence of many learning agents acting competitively, is learning still possible? This seems to depend on how market dynamics increase in complexity with the number of additional agents and the correlation between their payoffs [24, 46, 42].

In summary, the process that seems to dominate high-frequency phenomena appears to be the coupling between the low-frequency latent demand with the high-frequency mechanics, *i.e.* the market-microstructure of continuous-time double auction markets in the presence of limited liquidity. We speculate that the role of learning and the choice of order types in optimal execution are fundamental to this process.

To explore this, we simulate how latent demand is revealed in a single stock financial market using an ABM to capture the market environment into which we introduce both single and many optimal execution agents that engage in learning. The execution agents recieve their parent orders from agents generating the latent demand, and these agents each execute a single parent order in a market environment that consists of (minimally intelligent) chartists, fundamentalists, and high-frequency market makers. Model-free learning is incorporated using a simple (Multi-Agent) Reinforcement Learning specification. From these simulations, we examine how different model specifications affect stylised facts.

We investigate whether the addition of the new agent-type — the execution agents with learning — to the traditional agent classes used to define the learning environment, can provide a more complete description of market ecology both in terms of stylised facts, but also by trying to recover the empirically measured market complexity. To do this, we view the ABMs as nonlinear dynamical systems and compare their complexity as measured by Grassberger-Procaccia correlation dimension plots (See Figure 8a).

However, our main contribution is to demonstrate how different model specifications affect stylised facts. This yields several key findings. Firstly, we find that *learning decreases the persistence in order flow*, with some evidence that learning can also decrease the memory in the absolute returns. Second, we find that *the persistence of order flow is largely determined by the difference in the number of buying and selling agents*. Third, we find that *increasing the number of agents increases the persistence of order flow*. Fourth, the ability to use LOs to execute a parent order results in lower price impact and faster decay in the size

of absolute returns. This suggests that *a good approach to optimal execution will always be a judicious combination of limit-orders and market-orders*[1]. Surprisingly, we did not find conclusive evidence that learning reduced price impact. Lastly, the inclusion of many execution agents endowed with learning, trading a single stock, is not able to recover the complexity observed in real-world data.

The missing complexity demonstrates that our agent-based models are incomplete and we speculate this is because of the exclusion of intra-order book network effects; that market models will be incomplete unless they include the cross-order book trading. The missing complexity is substantial. Modelling order-books in isolation from other order-books and strategic agents may well be a first-order effect and is probably the most important observation arising from this project even when the model itself does reasonably recover the appropriate dynamics.

The rest of this paper is organised as follows. In Section 2, we briefly review the literature to motivate our investigation into optimal execution with learning. In Section 3, we specify a novel learning agent that can post market orders (MOs) and limit orders (LOs) to execute a parent order, and examine its learning dynamics (section 3.3). The underlying model for the environment is that provided by [17]. In section 4, we present the remaining results of our study, which includes the analysis of stylised facts in Section 4.1, and an investigation into the market dynamics and complexity in Section 4.5. Finally, in Section 5, we conclude by summarising our study and indicating possible future directions of research.

2. Background and Motivation

2.1. Reinforcement Learning for Optimal Execution

An important question in formal models of learning is determining what conditions allow for successful learning. Learning is said to be possible when, given a sufficient amount of data, the errors of the learner's outputs can be made arbitrarily small. When it is difficult to show analytically that an algorithm can learn, the ability to learn is intimated from measuring performance on test data. In sequential decision problems, asymptotic convergence guarantees have been derived for many reinforcement learning algorithms, such as Q-learning, under the assumption that the agent's environment is stationary [36]. Here, stationarity refers to state transition probabilities and the reward function. In financial markets, where the dynamics of the environment are changing, is learning possible?

Intuitively, we may guess that under a changing environment, a learning agent would have to continually relearn its policy to be optimal under the prevailing conditions. Thus, it would seem that asymptotic convergence

---

[1]Here limit-orders are resting orders passively inserted into the order-book with a price and volume requirement, while market orders are orders for immediate execution that demand a particular volume irrespective of the prevailing price.

may not be possible, but this need not be catastrophic to learning. Learning algorithms may still be able to outperform rule-based counterparts, but inferring performance on static data may be misleading.

Dicks et al. [17] show that a simple learning agent could outperform a Time Weighted Average Price (TWAP) agent even without policy convergence. Here the learning dynamics where of a simple RL execution agent in an ABM where non-stationary dynamics in the environment are primarily driven by the interactions of minimally intelligent agents. However, the difference in policy over consecutive training periods did appeared to converge.

This work extends [17] by studying the learning and market dynamics that arise from many learning agents interacting within an environment that includes minimally intelligent agents. Interactions between minimally intelligent agents can produce nonlinear dynamics, but agents' decision rules do not change over time, limiting the variation in the environmental dynamics. However, in the presence of multiple learning agents, the best policy of an agent changes in response to the changing environmental dynamics, which in turn changes with the other agent's policies [7]. Studying optimal execution in a MARL framework within an ABM can lead to a better understanding of how increasing the number of RL agents affects the agents' ability to learn, as well as how these affect the underlying ABM.

In multi-agent reinforcement learning, an important problem is the appropriate specification of learning goals, which typically involves a trade-off between stability and adaption [7]. The former is desirable for inferential purposes and establishing generalisation, but the latter is desirable for continual performance improvements. For example, learning stability can be established by specifying convergence to an optimal Nash equilibrium as a common learning goal. The combination of the two broad learning goals can be seen as having each agent converge to a stationary policy where each agent's policy is the best response to all the other agents' policies. Convergence to a Nash equilibria is easier to establish when agents are fully cooperative (agents have identical reward functions) or competitive (agents' rewards are zero-sum), in comparison to when agents are trained independently without a common goal [5]. In the latter case, asymptotic performance guarantees of (single-agent) Q-learning may fail [47], but training agents independently still can work well in practice [50, 23].

2.2. Learning in Complex Multi-player Games

This work is further motivated by the study of learning dynamics in complex games [24, 46, 42], where complexity is measured in terms of the number of possible actions. In particular, Galla and Farmer [24] considers a two-player game, and show how the learning dynamics of an RL algorithm called experience-weighted attraction (EWA) are affected by the correlation between the player's payoffs and the memory parameter of the learning algorithm. Galla and Farmer [24] finds that the learning dynamics of players' strategies can be separated into three different regimes. Namely, *i)* strategies can converge to a unique fixed point if agents have short memory, and the correlation between the players' payoffs becomes increasingly negative (ie. competitive), or *ii)* if correlations are negative but players have long memory, then learning dynamics can be chaotic or converge to limit cycles, and *iii)* if players have long memory and payoffs are positively correlated (ie. cooperative), then learning has a multiplicity of fixed points. In the second regime in particular, strategies are essentially random and learning is not possible. In two-player normal-form games, a possible reason for this is that as games get more complicated and/or more competitive this causes best reply cycles to become dominant, which [42] shows to be a good predictor of the non-convergence of several learning algorithms. In games involving more than two players, Sanders et al. [46] show that the parameter range in which learning converges to a fixed point becomes smaller as the number players increases. This implies that chaotic behaviour is characteristic of many-player games.

In summary, as games become more complex, and agents payoffs become increasingly independent, and the number of players increases, equilibrium becomes more unlikely. So, if normal-form games are a good model for the games market participants play, and if these participants can be modelled reasonably well by learning algorithms, this body of work calls into question the common assumption of equilibrium in economics and finance. However, a key constraint related to the carrying capacity of real-world markets is that of liquidity: without sufficient liquidity, trading decisions can not be made, compromising both the ability to learn and the profitability of the corresponding policy. Hence, to understand how the scaling insights of many player games can impact financial markets, one needs to include a broader ecosystem that captures these salient constraints: liquidity and the indirect cost of trading.

2.3. Market Ecology

The interactions of market participants can be viewed from an ecological perspective, forming a useful conceptual framework to develop ABMs to investigate disequilibrium price dynamics. Agent classes are specified in terms of strategies, analagous to "species" in ecological systems, which are rules describing how agents make trading decisions and hence govern their interactions [21]. Whilst market ecology is far from a formal theory of market functioning, two classification systems seem to have emerged in the literature: i.) Chartist-Fundamentalist-Noise (CFN) models and ii.) Liquidity-Provider (LP) and Liquidity-Taker (LT) models [2]. In both classification systems, agent interactions are intermediated by price. Many of these types of models allow strategy switching [15, 6, 9] as a form of adaption and naive learning.

There is a key nuance relating to where and how learning can operationally take place in real financial markets

with these types of agent representations. In real financial markets, traders and money managers do not easily change their trading strategies because of the regulatory environment in which they reside *e.g.* risk capital allocations on trading desks are made based on strategy definitions and trading simulations, while asset management mandates to money managers are most usually stated *a priori* as part of agreements between providers of capital and those trading and investing, even in CTA funds. For this reason, we favour a hierarchical approach where there is strategy die-out and the emergence of new traders with particular *a priori* strategies for the design of the fundamentalists and chartists. This is part of the key design decisions made in this work and in prior work that is used to define the learning environment [17].

It should be noted that some of the early prior literature bundle learning feedbacks and strategy definitions into a single model layer and thus risk conflating the impact of learning and adaption with the impact of die-out and feedback without learning (by allowing in-layer strategy switching) [6, 13]. Similar care is prudent when considering the emergence of herd behaviours [15, 37] without carefully considering the driving force behind herding itself, which is most often driven by higher order or adaptive behaviour of predators. In our work, the reinforcement learning component relates entirely to optimal execution in the presence of low-order strategies that are either fundamentalist (and contrarian) or chartists (and trend following) and learning-based adaption is separated into an isolated model structure. In part, this is why in this work we place some weight on the descriptive narrative surrounding the trophic level structure. These structures are incredibly important in real ecosystems for system stability for the same reason that we expect them to be important in financial markets [1].

The choice we have made here is to follow the approach of Wilcox and Gebbie [48] where it was argued that what has been learnt about more general complex systems [1, 19, 20] should probably be considered, and transferred into how models are built in financial market ecosystems. Particularly if the most important feature of real stable complex systems is a hierarchy of causal complexity [1] — that the causal classes create system robustness through network effects that are not entirely bottom-up.

In our model setting, the reinforcement learning agent would be an example of Top-Down Causation class IV (TDC4) engaging in feedback control with adaptive goals [19, 20, 48] while the lower level in which the fundamentalist and chartist reside are Top-Down Causation class II (TDC2) as agents engaging in non-adaptive information control. It should be noted that the market-makers facilitating liquidity provision in actual markets would have an objective function and engage in risk management, or rather money management portfolio control, and would appear to be more like the Top-Down Causation Class III (TDC3) agents engaging in adaptive selection. Our implementation is different because we feel that we can correctly capture the high-frequency traders being mechanistic liquidity providers at first order, and as such, in our implementation appear more like a TDC2 actor, than a TDC3 actor.

The reason for this is that we have excluded intra-order book traders and the inventory management component for real risk traders and market makers. Here, we have chosen the simplest model that we felt captures the model design narrative we are interested in while separating learning as a higher-order function separate from the niche being created that learning can exploit (the environment created by the fundamentalists and chartists in the presence of liquidity providers). We now consider the learning environment more carefully [17].

Chartists determine trading behaviour purely on past performance, whereas fundamentalists determine trading behaviour based on current price relative to some subjective valuation. Chartists and fundamentalists have opposing effects on price dynamics, and their interaction is able to produce many stylised facts observed in intra-day and lower frequencies (mesoscale), including clustered volatility and some additional fattening of the tails in returns. In contrast, the liquidity provider and liquidity-taker classification is useful to explain additional microstructural stylised facts in terms of how information is processed into prices by agents operating at different timescales, whilst keeping prices unpredictable *e.g* some order-flow memory.

In particular, liquidity-takers operating at low frequencies (particularly the fundamentalists) with large liquidity demands, can only trade incrementally owing to the limited available liquidity provided by higher frequency market makers. In between the large low-frequency liquidity takers and high-frequency market markers, are chartists acting at shorter time intervals acting on information contained in price changes. An additional feature of our model formulation is that we consider the fundamentalists and chartists to be minimally intelligent in the sense that the price levels that they trade relative to are randomly generated, and they are operationally trading noise in an environment with limited liquidity. From trading the noise, structure emerges that the learning agents can respond to.

To capture an ecology with limited liquidity and reasonably realistic market impact, we postulate a model in terms of 3-trophic levels defined respectively by three agent classes: learning agents, liquidity takers, and liquidity providers. In any given epoch, the liquidity takers can have increasing and decreasing profits and face gamblers ruin, but they cannot switch strategies. This is similar to the thinking used in a 3-trophic level model of carnivore-herbivore-plant systems. Here, we have learning agents as the "carnivores", or predators that engage in opportunistic behaviours which necessarily require adaption and hence, learning. We have the liquidity takers, both the fundamentalists and the trend-followers, as "foraging herbivores" since their activity is rule-based and hence "passive" in their consumption of liquidity. Liquidity is the "food" provided by "plants", which are uninformed High-Frequency

Traders (HFTs), or rather Electronic Liquidity Providers (ELPs). This suggests not only a financial market equivalent of a "food web" but also the inherent nonlinearity and feedbacks. Again, we include this because we believe that financial market model builders can learn from other domains that have become adept at model building and testing in complex systems — in particular, those working in Ecosystem model building and simulation.

How trophic levels are defined in financial markets is necessarily contested, and can be fluid as markets change and adapt because financial markets are reflexive [44]. However, this metaphor frames an the narrative around the role of learning agents within our financial market ecosystem — specifically the nature of strategic order-splitting in an environment with a highly constrained carrying capacity. The point of using such a narrative this way is to try to move the model building away from mechanistically implementing this or that convenient mathematical feature, and then coming up with a narrative that uses the mathematical trick, and then testing against the stylised facts; to probing how we think about the agents and the environment itself that the model aims to represent. Here, we aspire to create a link between the narrative, the models, and the observations on an equal footing.

We apparently do not need learning for the first two trophic levels of our system since static mechanistic rule-based responses to the market states seem to be sufficient to recover almost all of the necessary stylised facts of the environment [17]. Learning seems to only become important as a coupling mechanism between the visible market (the learning environment) and the latent order book. This provides a mechanism to drive strategic order splitting that can be used to fine-tune the observed balance between order-flow persistence, price volatility and clustering, and the observed extreme events in the complete market. This is particularly interesting because we know from the prior work that further tuning the parameters describing the environment cannot do so [17].

Whilst there are feedbacks between each class of agents in the different trophic levels, our framework suggests a hierarchy based on dependency. Although not a functioning market, the presence of ELPs is sufficient for the existence of a persistent order-book, hence providing an environment for the other agent classes, and thus existing at the bottom of the hierarchy. In contrast, learning is situated in higher trophic levels [1], because the higher levels necessarily include adaption via feedback control[2]. However, from simulation work and the sensitivity analysis we know

that learning agents cannot adapt to ELPs alone because that would merely create noise – this should be explored more carefully in future work and is outside of the scope of this project. This indicates the need for minimally intelligent agents at an intermediary level whose actions create learning opportunities by using the noise signal to generate exploitable but transient structure via emergence. This picture, in turn, would suggest that a market made entirely of learning agents would not be feasible because of the extremely limited liquidity and that in simulation the algorithms synchronise and via tight-coupling lead to system failure – this should be explored more carefully in future work.

Concretely, we find that if only learning agents interact with noise traders learning is not possible unless the noise is synchronous across agents. We speculate that this implies that the carrying capacity of a market is a crucial feature. Any model that simulates the herding of learning agents presupposes sufficient liquidity to profitably support trading as well as learning and that the underlying data-generating processes are sufficiently synchronised to trigger coordination. Here the carrying capacity of the environment would be too low to support both the cost of learning, and then using it successfully, because other agents adapted similarly to the single learning agent with insufficient liquidity to take the necessary learning actions. This, in turn, points to the importance of the liquidity multiplier effect introduced by the minimally intelligent liquidity takers and potentially why a hierarchy of causality may well be a crucial model feature in realistic financial market models and simulations.

2.4. Limitations

One question surrounding the study of ABMs is determining whether simulated paths represent general agent and market behaviour or are unique to modelling choices. In our implementation, we address this by separating various sources of variation and important model features in the modelling process. From the modelling side, we have ensured that learning only takes place at the highest level in the hierarchy of complexity in our model and that there is no strategic adaption in the middle level where our minimally intelligent liquidity takers reside. At the lowest level, we have zero-intelligence noise traders providing liquidity.

We retain the three-step approach for the development and calibration of the learning environment [33, 17]: 1. sensitivity analysis, 2. calibration with a minimal set of known and re-used seeds using the relationships between model and parameter variations found from the sensitivity analysis, and 3. simulation using the same restricted number of seeds using the calibrated parameters for the training environment.

The number of random seeds is chosen to crudely optimise the compute times while providing relative stability of the parameters and a learning environment, to faithfully capture the empirically measured environment. Many sample paths are simulated from each random seed to facilitate

---

[2]Auletta et al. [1] use the term *information control* to make clear the importance of: i.) information selection, and ii.) feedback control. We draw the interested reader to their Figure 7 [1]. The key idea is that top-down causation by information control occurs when an equivalence class is established, and the information selection defining the operations of the class is conserved because of modularity, despite the variability of lower-level variables. This is part of why learning and adaption are typically found in higher levels, and not lower levels, in real-world systems.

learning since including learning within an agent-based model imposes computational considerations that are at least an order of magnitude more onerous than a typical sensitivity analysis.

One of the drawbacks of ABMs is that they are computationally expensive to run, necessarily forcing pragmatic choices on the modeller. In this work, key sub-cases of the agent-based model were prudently chosen to explore the impact of both model and sample variations, whilst trying to prevent model and path variations that could confound our results. Since generating all possible combinations of parameters and paths is not computationally tractable, a subset must be pre-selected for simulation, which substantially restricts this type of simulation work.

Consequently, it may appear there is the possible danger of selective bias in the presented results, where possible sub-cases, parameters, and paths are chosen *a-priori* to support a particular conclusion. However, owing to the nonlinearity of the system and the inclusion of various sources of noise, we do not know with any certainty the possible results corresponding to different parameter combinations, precluding the possibility of selective bias. However, conclusions from restricted results are never as robust as one would like, particularly when critiqued through the lens of linear statistical modelling techniques aimed at inference. Rather, our results are indicative and aimed to guide incremental model refinement to explain specific model features. One key lesson is that, although we can capture most of the key stylised facts, we are not able to capture the observed market complexity, an experimental outcome that we think is very important; however, many of the narratives that emerge do make physical (or rather financial) sense. Because of the missing complexity we are very cautious about further model refinement and conclusions.

3. Agent Specification and Learning

We investigate the interaction of different types of optimal execution agents within a pre-calibrated event-based minimally intelligent ABM, which forms a training environment. The minimally intelligent agents consist of chartists, fundamentalists, and liquidity providers. The first two act as minimally intelligent agents without strategy switching, the last class is a zero-intelligence mechanistic noise trader. Their full specification, as well as details related to their learning dynamics, can be found in Dicks et al. [17], which we do not restate here for the sake brevity. We study different cases, described in Table 3, where each case is defined by the number, type, and side (*i.e.* buying or selling) of the agents, to determine how each of these characteristics affects the stylised facts.

The state space used for both agent classes corresponds to the smaller state space from Dicks et al. [17], which is described in Table 1. Consequently, the results of this study are directly comparable with the results of prior work as part of the model design. Similarly, the agent actions for MOs as based on multiples of a TWAP strategy to provide an additional dimension to compare with prior work. As such, this study is a direct extension of Dicks et al. [17].

3.1. Actions

Following Dicks et al. [17] the volume of an outstanding parent order is X_0 and an order will have a volume trajectory $\{x_i\}_{i=1}^N$ so that $X_0 = \sum_{i=1}^N x_i$. Then the ith decision point in a trading schedule gives a trade with volume x_i. Here all the agents use trading schedules that are multiples of a TWAP strategy for the original parent order. At each trade decision point there is an action a_i. For market orders, the action is a multiplier such that the submitted order has volume $a_i x_i$ from the volume trajectory that defines the TWAP trading schedule. For limit orders, the actions are the placement depth a_δ and rate of trading a_ν. This suggests three basic types of execution agents: the benchmark TWAP trading schedule, a market order only trading schedule, and a trading schedule that combines market orders with limit orders. These three agent types are defined in Table 2.

The first type of agent is denoted "Type S" and is a minimally intelligent execution agent characterised by a fixed TWAP schedule consisting of MOs only. The minimally intelligent type of execution agent is a benchmark agent and should be differentiated from the two learning agents.

We denote the first class of learning agents as "Type I" agents [17]. Type I agents have explicit order-splitting but only use MOs to interact with the market. Example actions as they relate to states for Type I agents are given in the heat map legend on the left of Figure 2a and Figure 2b. This means that Type I agents cannot directly interact with each other in the model but are inter-mediated only by the liquidity providers.

Furthermore, in this study, we introduce a new class of agents, denoted "Type II" agents, which can trade with both MOs and LOs. However, Type II agents can interact directly with each other and other agents as they use both LOs and MOs to trade. Example actions as they relate to states for Type II agents are given in the heat map legend on the left in Figure 3.

Both type I and type II agents use trading schedules that are multiples of TWAP and order sizes following Dicks et al. [17]. The action set of the Type I agents are given by $a_X = [0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2]$ giving a grid of actions from small orders to large orders as multiples of the TWAP strategy. For the Type II agents, MOs are placed into the market at some rate of trading ν as a function of machine time, while the LOs will be placed into the orderbook at a depth δ from the mid-price. The order sizes will be as before. Here the actions will be the placement depth at an activation is $a_\delta = [0.01, 1]$ (shallow or deep) and the rate of trading $a_\nu = \frac{1}{T}[100, 10, 1]$ (fast, moderate or slow) as a multiple of the total session length.

Table 1: The best bid/ask volume (v_k) states are given, and the spread (s_k) states. Here the all provide the approximate probability of being in each states are based on the observed data at event k and the simulated historical distributions. Explain how the bid and ask volume states works. Let $\tau = T_0/n_T$, where T_0 is the total trading time for the RL agent and n_T is the number of time states. Let $i = X_0/n_V$, where X_0 is the total amount of inventory to be traded and n_V is the total number of volume states. The actions associated with these states are visualised in Table 2 and Table 3.

State	Inventory states		Time states		Volume states		Spread states	
	Range	Prob.	Range	Prob.	Range	Prob.	Range	Prob.
	$\cdot < x_k \leq \cdot$		$\cdot < t_k \leq \cdot$		$\cdot < v_k \leq \cdot$		$\cdot < s_k \leq \cdot$	
1	0　i	0.2	0　τ	0.2	31	0.2	0　1	0.6
2	i　2i	0.2	τ　2τ	0.2	31　266	0.2	1　2	0.123
3	2i　3i	0.2	2τ　3τ	0.2	266　1453	0.2	2　3	0.062
4	3i　4i	0.2	3τ　4τ	0.2	1453　5209	0.2	3　7	0.091
5	4i	0.2	4τ	0.2	5209　$\cdot$	0.2	7　$\cdot$	0.124

3.2. Rewards

As in prior work [29, 17], we considered using Perold's implementation shortfall [43], where the buying agent would try minimise the difference between the Volume Weighted Average Price (VWAP) achieved using its strategy and the hypothetical VWAP it would receive if it traded the entire parent order at the initial price, with no price impact using an Immediate Execution (IE) strategy. However, in this case, the minimum implementation shortfall is negative, which can incentivise buying at higher prices. Conversely, if you aim to maximise the implementation shortfall, to try to incentivise buying at lower prices, then you have the same problem arising from minimising the total cost because the agent learns not to trade.

For the Type II agent, we specify a reward function that is symmetric between buying and selling agents, without resulting in agents learning degenerate policies. It is important to note that the simple reward function of the agent specified in Dicks et al. [17] is suitable for profit maximisation on the sell-side only. Attempting to construct a symmetric reward function applicable to buying agents through cost minimisation results in a degenerate policy which minimises trading costs by not trading. Thus, some form of penalty needs to be included in a suitable reward function to encourage trading. This is summarised in Table 2.

The aim is to formulate a reward function that is intuitive and minimises the cost on the buy side, maximises the profit on the sell side, and creates an incentive to trading. This can be achieved by a combination of the trader's slippage and a penalty for not trading.

We start with $p_{\text{VWAP}}(\mathcal{X})$, the VWAP price received from the set $\mathcal{X} = \{x\}_{i=1}^n$ of all trades including the nth trade submitted by the ℓth RL agent. The VWAP price found from all trades, excluding all the trades submitted by the ℓth RL agent, is $p_{\text{VWAP}}(\mathcal{X} \setminus \mathcal{X}_\ell)$. The difference between these two quantities measures the slippage between a given RL agent's trades, and the trades made by the rest of the market (potentially including other RL agents).

The total time past in the simulation is t, the remaining inventory is $x_{\ell,n}$, and the total volume matched for the nth order is v_n. We propose a penalty that increases ex-

ponentially as time increases, forcing the agents to learn to trade before the end of the simulation. The penalty is proportional to the inventory remaining and inversely proportional to the amount of volume matched for the last trade, ensuring that the penalty increases as the remaining inventory increases, and decreases as the amount traded by the n_{th} order increases. The agent will want to trade more when there is a lot of inventory remaining. The penalty has two parameters: λ_r controls how much effect the penalty term has on the reward function, and γ_r controls the sensitivity to time.

Now, for the nth order in an episode the *reward function* for the ℓth learning agent can be given as combination of the slippage and penalty:

$$R_{\ell,n} = \pm \underbrace{\ln\left(\frac{p_{\text{VWAP}}(\mathcal{X})}{p_{\text{VWAP}}(\mathcal{X} \setminus \mathcal{X}_\ell)}\right)}_{slippage} - \underbrace{\left(\frac{x_{\ell,n}}{v_n}\right)\lambda_r e^{\gamma_r t}}_{penalty}. \quad (1)$$

The first term is the *slippage*, and is positive for selling agents and negative for buying agents. By maximising this reward, we aim to get higher prices when we sell, and lower prices when we buy. The second term is the *penalty* term for not trading, which creates an incentive to trade, whether on the buy or sell side.

Together, the slippage and penalty mean that the agents aim to minimise slippage and get the best prices relative to the rest of the market while being incentivised to trade. The return function for the ℓth RL agent is the accumulated reward: $\sum_n R_{\ell,n}$. This will be maximised by all the RL agents.

3.3. Optimal Policies and Convergence

To visualise the convergence of the training of learning agents we plot the agent policy returns as a function of the training episodes in Figure 1. Here we have used 1000 training episodes. Figure 1b shows the return rewards for agent type I, where both buyers (+) and sellers (-) are shown. Agents are taken from different model configuration sets *e.g.* the blue line is a lone buying agent (I^+) using only MOs and has similar dynamics to the lone selling agent (I^-) given in pink. This shows that the rewards appear to converge under training. Similarly, Figure 1b has

Table 2: Agents using strategic order splitting. Those engaged in learning their respective rewards and actions are given with the state-space size on (n_t, n_I, n_S, n_v). All the agents below engage in learning except for the minimally intelligent agent S

| Agent type | Reward | State-space | | Actions | |
		Size	$\pm$	Parent order	Order-type
Hendricks and Wilcox [29]	short-fall	5	Sell	multiple of AC	MO's
Dicks et al. [17]	trading profit	5,10	Sell	multiple of TWAP	MO's
Type $S^{\pm}$	-	-	Buy/Sell	TWAP	MO's
Type $I^{\pm}$	slippage - penalty	5	Buy/Sell	multiple of TWAP	MO's
Type $II^{\pm}$	slippage - penalty	5	Buy/Sell	multiple of TWAP	MO's & LO's

Case #	Types	#Agents	Parent order size
0	-	0	-
1	S^-	1	6% ADV
2	$5S^+$	5	$\frac{1}{5}$(6 % ADV)
3	I^+	1	6% ADV
4	I^-	1	6% ADV
5	I^+,I^-	2	3 % ADV
6	II^+	1	6 % ADV
7	II^+,II^-	2	3 % ADV
8	II^+,I^-	2	3 % ADV
9	$5I^-$	5	$\frac{1}{5}$(6% ADV)
10	$5II^+$	5	$\frac{1}{5}$(6% ADV)
11	$5I^+, 5I^-$	10	$\frac{1}{10}$ (6% ADV)
12	$5II^+, 5II^-$	10	$\frac{1}{10}$ (6% ADV)

Table 3: Different combinations of agents engaging in strategic order splitting. The first case is the agent-based model without any learning agents – type 0. The minimally intelligent agents engaging in naive order splitting using a TWAP strategy are denoted as agents of type S. The agent types are either an acquisition agent ($+$) or a liquidation agent ($-$). The number of acquisition agents and liquidation agents is given for each case. The overall order size is in multiples of $X_0 = 6\%$ of Average Daily Volume (ADV) for all agent classes to ensure that the market has similar liquidity across all the cases considered. The learning agents are then either of class I or II. Here class I only use market-orders and class II use both market-orders and limit-orders (see Table 2). The state-space is of size 5 for all the learning agents.

the same plot, but for agents of type II — agents that use both MOs and LOs. Again, we note reasonable evidence of convergence behaviour in the reward function over the 1000 training episodes for the ten different learning agent configurations. The benchmark TWAP agents ($S^{\pm}$) are included for comparison.

In Figure 2 we show the final buying and selling learning agents greedy policy after 1000 training episodes in the training environment [17] for Case 5 (the case which includes both type I^+ and I^- agents together in the environment). The five discrete inventory states increase from bottom to top, and the temporal states increase from left to right across the episode. Within each inventory and time state there are five spread and volume states, represented as heat maps. The specific actions are labelled as colour legend on the left. Action "-1" represents the state has not been reached, and increasing aggression, via

the size label, moving from the bottom to top. Here only MOs are used. The learning agent with a mixture of MOs and LOs is given in the action space example for Case 6 in Figure 3.

In Figure 3 we have the final greedy policy for Case 6 (type II^-) learnt over 1000 training episodes in the ABM training environment. The five inventory states increase from bottom to top, and the five temporal states from left to right, from the first fifth of the training episode to the last fifth. Within each inventory and time combination a 5×5 heat map for the spread and volume state combinations is plot. The legend on the left provides the mapping from the number of actions taken for a particular combination to the action and represents the greedy actions taken in a particular state. The Case 4 agent is a combination of MOs and LOs actions with varying aggression.

4. Exploratory Data Analysis

The measured trade-and-quote (TAQ) data is a single 8-hour day of trading, on 08-07-2019, for a single Johannesburg Stock Exchange (JSE) dual-listed equity, Naspers (NPN.J) [34]. The data excludes all auctions. The simulated data similarly represents a single period comparable to 8 hour of trading on the basis of the Average Daily Volume (ADV) where the ABM component was calibrated to the estimated moments. The method of simulated moments was used to calibrate the model for the base case (Case 0) and is described in prior work [33, 17].

4.1. Comparing Stylised Facts

Table 4 compares the moments for key configurations. The sample paths used to capture the model variations are comparable to those found in the calibration. The cases are ranked, left to right, on the micro-price fluctuation volatilities and the Geweke and Porter-Hudak (GPH) statistic [25] from Table 4 — this is a measure of the long-range dependency. We find that all the models have more extreme events than found in the real-world data and that none of the classes suggests evidence of a unit-root.

Multiple balanced learning agents (Case 5) increase the long-range dependencies (GPH) and mean-reversion (as measured by the Hurst exponent [32]), possibly because of interactions. Conversely, Type II agents decrease long-range dependency whilst reducing volatility. We believe
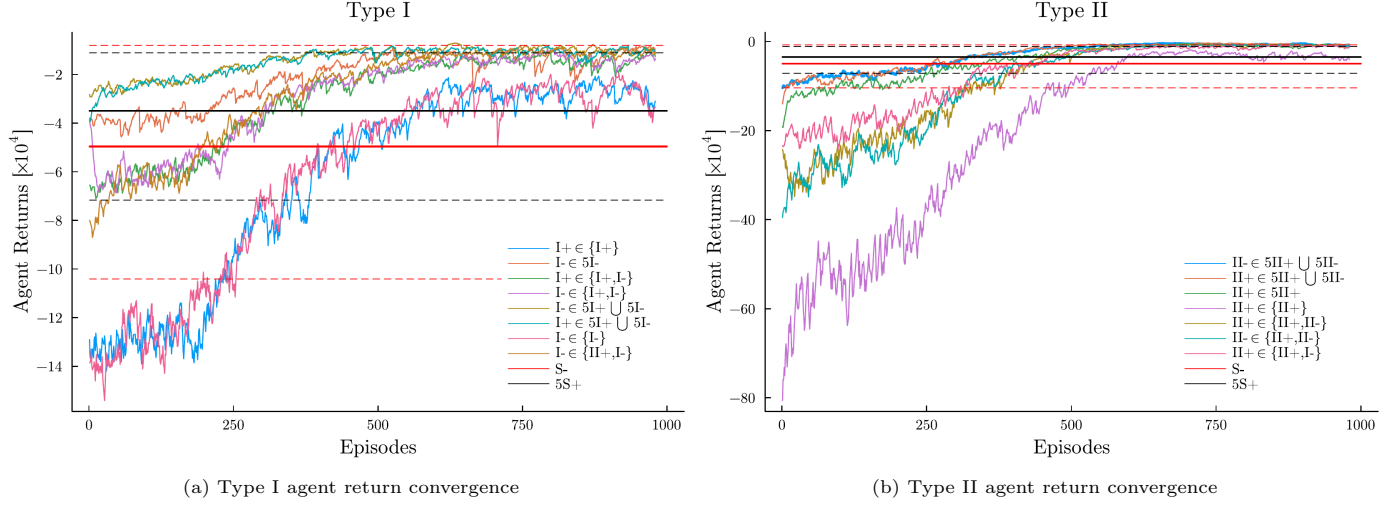
(a) Type I agent return convergence

(b) Type II agent return convergence

Figure 1: The agent returns are given as a function of the training episodes. This demonstrates how the different agent's rewards converge under training. In Figure 1a, the return rewards for type I agent, both buyers (+) and sellers (-) are shown. Type I agents only trade using market orders.1b has the same plot, but for agents of type II — agents that use both market orders and limit orders.
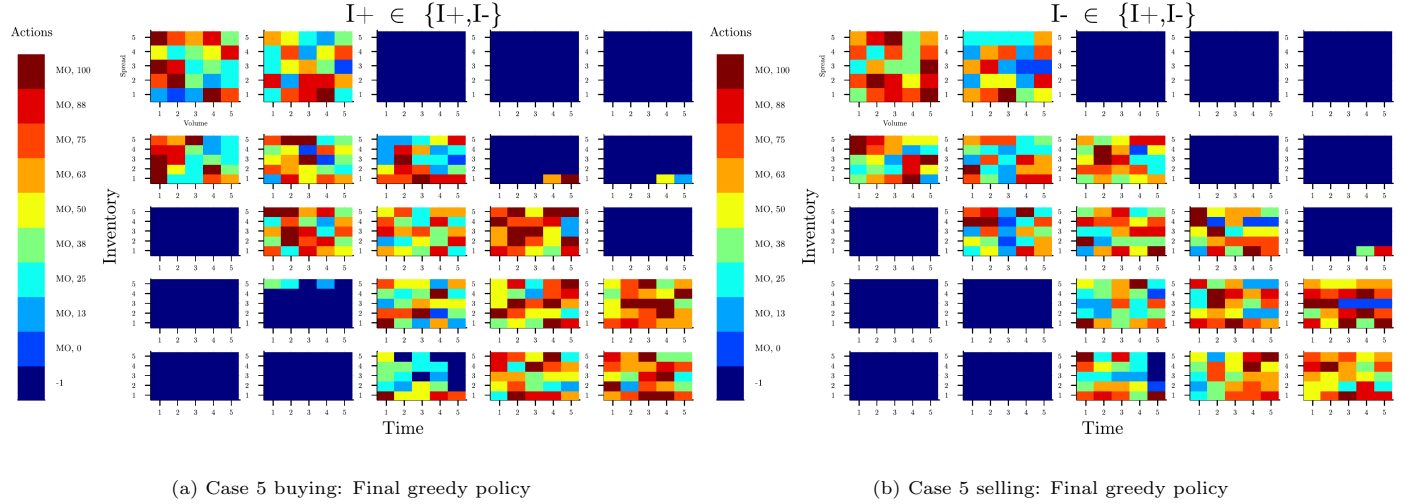

(a) Case 5 buying: Final greedy policy

(b) Case 5 selling: Final greedy policy

Figure 2: The final ϵ-greedy policy is shown for a type I agent example as a heat map. Here for Case 3 (see Table 3) with the buying agent in Figure 2a and the selling agent in 2b.

this occurs since allowing execution agents to post LOs introduces additional heterogeneity in the order flow and liquidity provision processes, reducing long range dependence, whilst the additional liquidity provided by these agents reduces the occurrence of liquidity shocks, decreasing volatility. From the Hill estimators [40], we see that Case 5 has less extreme events (a slower decay in the tail distribution exponent) than Case 1, which may suggest that balanced execution agents (*i.e.* an equal number of buying and selling agents) more generally, reduces the tail effects.

4.2. Persistence of Orderflow

The ACF of tradesigns reflects the persistence in the direction of the MO flow[3]. Due to potentially large sam-

pling variation, analysis of a single path is not indicative of general behaviour. On the other hand, averaging sample ACFs tends to conceal interesting differences in the distribution of sample paths between the different cases. Thus, we consider average sample ACFs and the individual sample paths to explain the ABM behaviour, and discuss broad phenomena observed from their plots.

Firstly, as seen in Figure 6 (left), the average level of ACF reflects the number and (net) direction of the execution agents. The ABM was calibrated by design in an upward-trending market, and hence the environment base case produces sample paths that are slightly more biased towards buying (Case 0). Adding execution agents increases the persistence of the trade signs, and hence the ACF, simply by increasing the proportion of buys in the order flow. Conversely, including selling agents in the ABM results in an increase in activity in opposition to the orig-

---

[3]The ACFs are not on demeaned data because the data is ordinal.

9

| Moment | Simulated | | | | | | | | Estimated | |
| | Case 5:ABM + I$^{\pm}$ | | Case 1:ABM + S^{-} | | Case 0:ABM | | Case 6:ABM + II^{+} | | JSE:NPN.J | |
	$\boldsymbol{m}^s$	[97.5% CI]	$\boldsymbol{m}^s$	[97.5% CI]	$\boldsymbol{m}^s$	[97.5% CI]	$\boldsymbol{m}^s$	[97.5% CI]	$\boldsymbol{m}^e$	[97.5% CI]
Mean	0	-	0	-	0	-	0	-	0	-
Std×10^{-4}	4.60	[3.90,5.30]	4.04	[3.36,4.75]	2.31	[1.61,3.02]	1.48	[0.78,2.18]	1.39	[1.19,1.59]
KS	0.22	[0.16,0.28]	0.16	[0.16,0.27]	0.18	[0.12,0.23]	0.27	[0.21,0.32]	0.00	[-0.01,0.01]
Hurst	0.29	[0.22,0.35]	0.33	[0.26,0.40]	0.40	[0.33,0.46]	0.38	[0.31,0.44]	0.47	[0.41,0.52]
GPH	0.69	[0.57,0.82]	0.62	[0.49,0.74]	0.51	[0.39,0.63]	0.42	[0.29,0.54]	0.44	[0.30,0.59]
ADF	-154	[-158,-151]	-158	[-161,-154]	-148	[-152,-145]	-167	[-171,-164]	-136	[-140,-133]
GARCH	1.31	[1.24,1.38]	0.95	[0.87,1.02]	0.99	[0.91,1.05]	1.06	[0.99,1.13]	1,00	[0.96,1.03]
Hill	0.83	[0.41,1.25]	1.39	[0.97,1.81]	1.24	[0.82,1.66]	1.25	[0.83,1.67]	1.99	[1.72,2.26]

Table 4: Simulated moments (using the calibrated model for the environment) and estimated moments. These are the same used in prior work [17]. The estimated moments are from the market data [17]. The selected simulated moment set is ordered on decreasing micro-price volatility (Std.) left to right. The mean and standard deviation (Std) are the usual sample moments, the remaining are as follows, Kolmogorov-Smirnov (KS) statistic [39], Hurst exponent [32], Geweke and Porter-Hudak (GPH) statistic [25], Augmented Dickey-Fuller (ADF) statistic [16], the sum of the parameters of a GARCH(1,1) model [49], and the improved Hill estimator [40]. The cases left to right from Table 3: Case 5 is the ABM environment with both a buying and selling agent trading in MO's, Case 1 has the ABM combined with the benchmark TWAP trader, Case 0 is the ABM environment alone, and Case 6, that ABM combined with a single buying agent trading in both LO's and MO's.
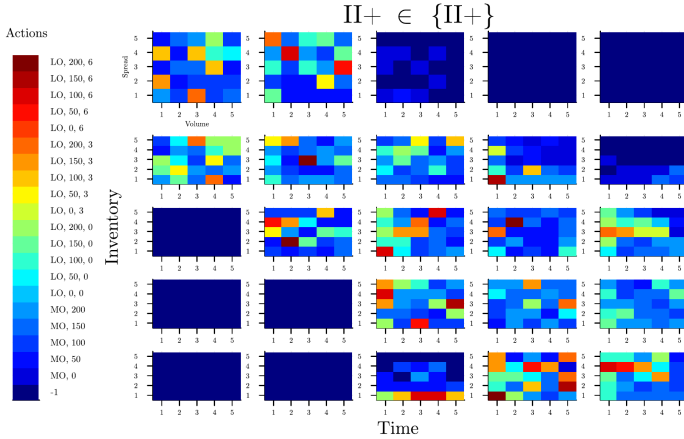

Figure 3: The final greedy policy example for a type II agent as shown for Case 4 (see Table 3) as a heat map.

inal orderflow, thus decreasing the ACFs. Including both selling and buying agents produces ACFs that tend to lie between the ACFs of the buying and selling agents, respectively. The tradesign ACF plot is ordered by colours on the tradesign to make this more apparent.

However, as seen in Figure 6 (right), we believe the observed increases or decreases in demeaned ACF, largely reflect the bias of direction of order flow in the ABM. If the ABM was biased towards selling, we would observe the effects in the opposite direction observed here. Nonetheless, we observe that the change in level from including agents of a given type becomes more pronounced with the number of agents, as these execution agents start to dominate the order flow over the other classes. To show this we ordered the plot colours on the number of execution agents interacting with the environment.

Secondly, as seen in Figure 6, following the inclusion of execution agents, the ABM produces, with greater frequency, sample paths where the cumulative direction of the order flow persists for very long periods. The corresponding sample ACFs are linear with very slow decay. This is especially evident in the minimally intelligent cases (TWAPs), where ACFs become increasingly linear with greater frequency as the number of agents increases. Surprisingly, the presence of sample paths with persistent cumulative order flow is also evident (although to a lesser degree) when the number of buying and selling agents is equal. However, the remaining sample paths may be nonlinear, with several changes in the direction of cumulative order flow with corresponding nonlinear ACFs, which can produce nonlinear behaviour in the average ACFs.

Furthermore, when comparing a single selling TWAP agent (Case 1) with, for example, five buying TWAP agents (Case 2) competing for liquidity, it is noted that as the number of optimal execution agents increases from one to five, the cumulative order flow becomes increasingly uniform, leading to slowly decaying monotonic ACFs.

We observe that increasing the number of agents tends to decrease the rate of change of the ACF and is most evident at low lags (less than 1000 events). In the base case, the orderflow arises from the interaction of fundamentalists and chartists, which tends to have higher autocorrelations at shorter horizons, most likely due to minority game dynamics. In contrast, the order flow increasingly reflects the activity of execution agents as their numbers increase, which tends to result in linear slow decaying ACFs, and decrease the decay rate of ACF.

4.3. Price Impact

The buyer and seller initiated price impact curves are shown in Figure 4. Price impact is defined as the instantaneous change to the mid-price following a trade, which depends on the shape of the orderbook at the time of trade. The greater the amount of liquidity, particularly at prices at and close to best quotes, the lower the price impact. Conversely, reducing the available liquidity will increase price impact.
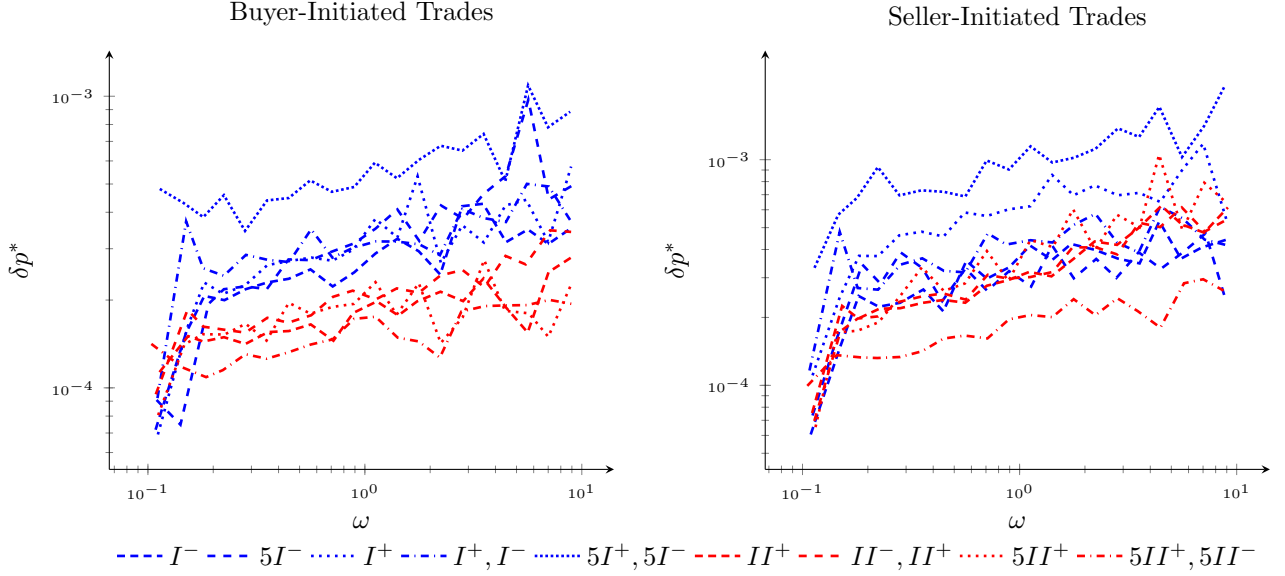
Figure 4: Price impact plots for the buyer (left) and seller (right) initiated trades for different cases of learning agents. Type I (blue) agents use only market orders to execute parent orders, whilst Type II (red) agents use market and limit orders. The lower price impact of Type II agents suggests that limit orders can be used to take advantage of opportunities created by market flow and changes in the spread.
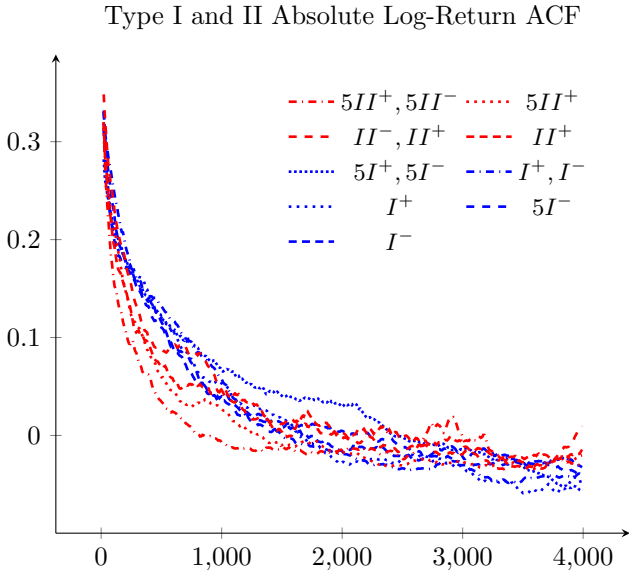

Figure 5: Auto-Correlation Function (ACF) plots of the absolute value of the mid-price returns comparing Type I (blue) and Type II (red) cases. Type I agents have non-trivial auto-correlations, while Type II agents suppress autocorrelations. This is indicative that the more complex Type II agents reduce regularity in potentially both the order flow and the liquidity provision processes.

Liquidity supply is dynamic and is reduced by liquidity-taking agents (and order cancellations), and is increased by liquidity-providing agents. Thus, differences between cases can be attributed to the differences in the trading and liquidity provision behaviour of the different agent classes. As with the analysis of tradesigns, the trading and liquidity provision behaviour is path-dependent and hence has a large degree of sampling variation, which we

try to eliminate by analysing average price impact curves.

The clearest pattern that we observe in Figure 4 is that cases with Type II agents have lower price impact than Type I agents. Type II agents can use LOs in place of MOs to execute their parent orders, thereby reducing liquidity-taking behaviour while increasing liquidity provision, both of which result in greater average available liquidity. The difference in the relative level of the buyer and seller initiated price impact can be explained to be due to the overall trend in the market – which is upward.

4.4. Memory in Absolute Returns

Figure 5 shows the ACF of absolute returns. This measures persistence in the size of returns calculated from micro-prices [4], reflecting the dynamics of the top of the order book. By definition, a change in the micro-price arises from a change in the top-of-book price and/or top-of-book volume. Changes in these quantities are due to events: a trade, a new limit order or a cancellation, and how these quantities change depends on the shape of the order book. The different event-types are mutually exclusive and have their own processes defined by volume, price, and relative frequency.

Thus, the ACF of absolute returns encapsulates the behaviour of a wider array of market variables, in comparison to the ACF of tradesigns and price impact curves, making any observed patterns remarkable and worthy of attention, but difficult to interpret.

In Figure 5 we observe two patterns which appear to support the hypothesis that the variation in liquidity demand in excess of liquidity determines the decay in the

---

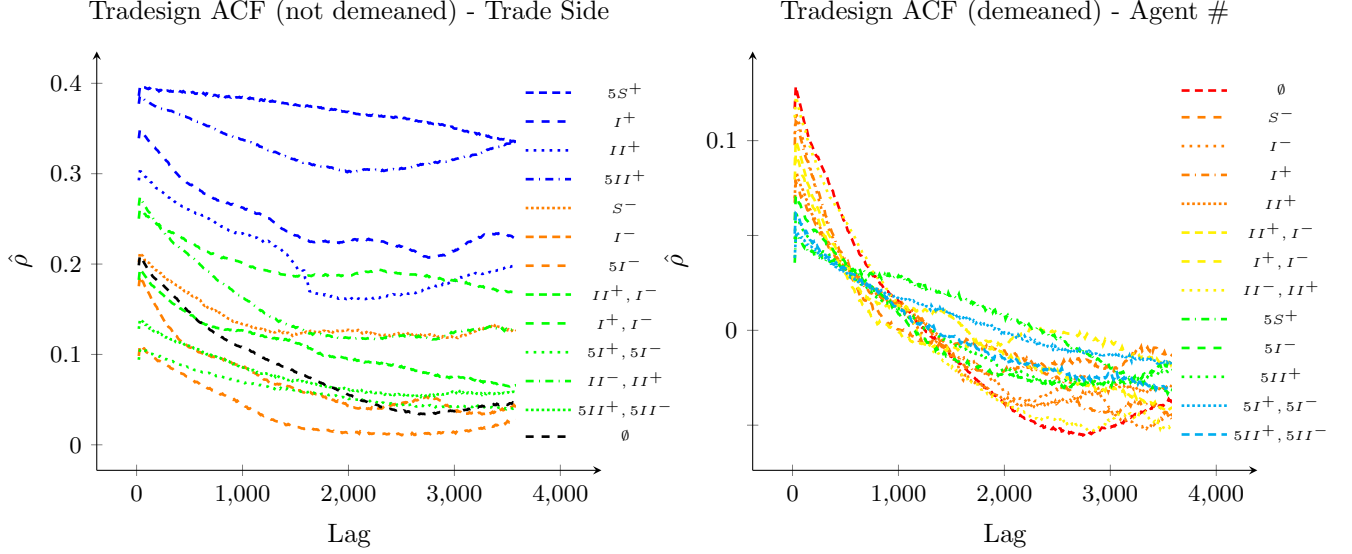[4]The ACF of the absolute returns uses demeaned data.

Figure 6: Auto-Correlation Functions (ACF) of tradesigns where the tradesigns are not demeaned (left) and demeaned (right), respectively. On the left we observe that the level of the ACF reflects the imbalance in the number of execution agents that are buying vs. selling. To demonstrate this we order the plot colours by the trade signs - buying (+), to selling (-), to increasing mixtures of buying and selling (±). This shows that the level of the ACF increases with more buying agents (blue), is moderated by including both buying and selling agents (green), and decreases by including selling agents only (orange). The directional changes in the ACF most likely reflect that of the environment ABM (black) which was by design calibrated in an upward-trending market. On the right, we demean the ACF so that the imbalance in the number of agents does not visually dominate the plot. We then order the plot colour by the number of agents. We find that increasing the number of agents (red is lowest, cyan is greatest) decreases the rate of decay of the ACF.

ACF of absolute returns. Firstly, the ACF decays at the slowest rate when five buying TWAP agents (Case 2) are added to the base ABM, since this reduces variation in the volume and frequency of MOs. Furthermore, we see more generally that the ACFs of Type II agents decay faster than Type I agents, because the ability to post LOs of varying size and depth introduces further variation in order flow and liquidity processes, which is temporally uncorrelated due to changes in the market's state. However, limitations in the data prevented convincing support or falsification of this hypothesis.

4.5. The missing complexity

Although we are able to recover many of the stylised facts (See Table 4 and figures 4, 5 and 6) we are not able to fully recover sufficient model complexity relative to the measure real-world data. This is shown in Figure 8 where the empirical data from the JSE test data is given with confidence intervals (red). The training environment is shown in black with confidence intervals. The learning agent configurations use the type labelling from Table 3.

We notice that the none of the model configurations are able to capture the full complexity of the real-world data, and all have dimensions less than at least 2. The single agents tend to have dimensions slightly greater than that of the ABM describing the learning environment, and the many agent configurations are slightly less. This is elaborated in Table 8b. Table 8b gives the relative differences of the model configurations. The agent case (first column), RL agents types (second column) and ΔD, the difference

between fractal dimensions of the different configurations, and the ABM's fractal dimensions, as averaged across the embedding dimension (see Figure 8a) are sorted on the number of RL agents interacting with the ABM.

We estimate the correlation dimensions using the method of Grassberger and Procaccia [27]. This gives us a reasonable bound on the required phase space from the micro-price data. The resulting correlation dimension will be used as a proxy for the representation dimension of a particular model configuration *i.e* as a measure of the model configuration's relative complexity. We select the correlation time in machine-time by selecting the first minimum in the micro-price autocorrelations [30].

From Table 8b, we can consider the relative differences in the model complexity measured at the higher embedding dimensions where there is the slowest increase in the estimated fractal dimension (the right of Figure 8a). In particular when focusing on the difference on the right of the inset of Figure 8a. Here we find that when single learning agents are combined with ABM, they tend to increase the fractal dimension, and hence, the complexity of the model configuration. Adding multiple learning agents tends to decrease the fractal dimension — possibly because these tend to interact with each other and reduce the overall impact of adaption and learning in the combined market. However, it is important to realise that the overall total outstanding initial parent order volume has been kept the same, at X_0=6% of ADV (see Table 3). This means that the combined outstanding orders, the volume exposure (Vol. Exp.) for the multi (many) learning agents
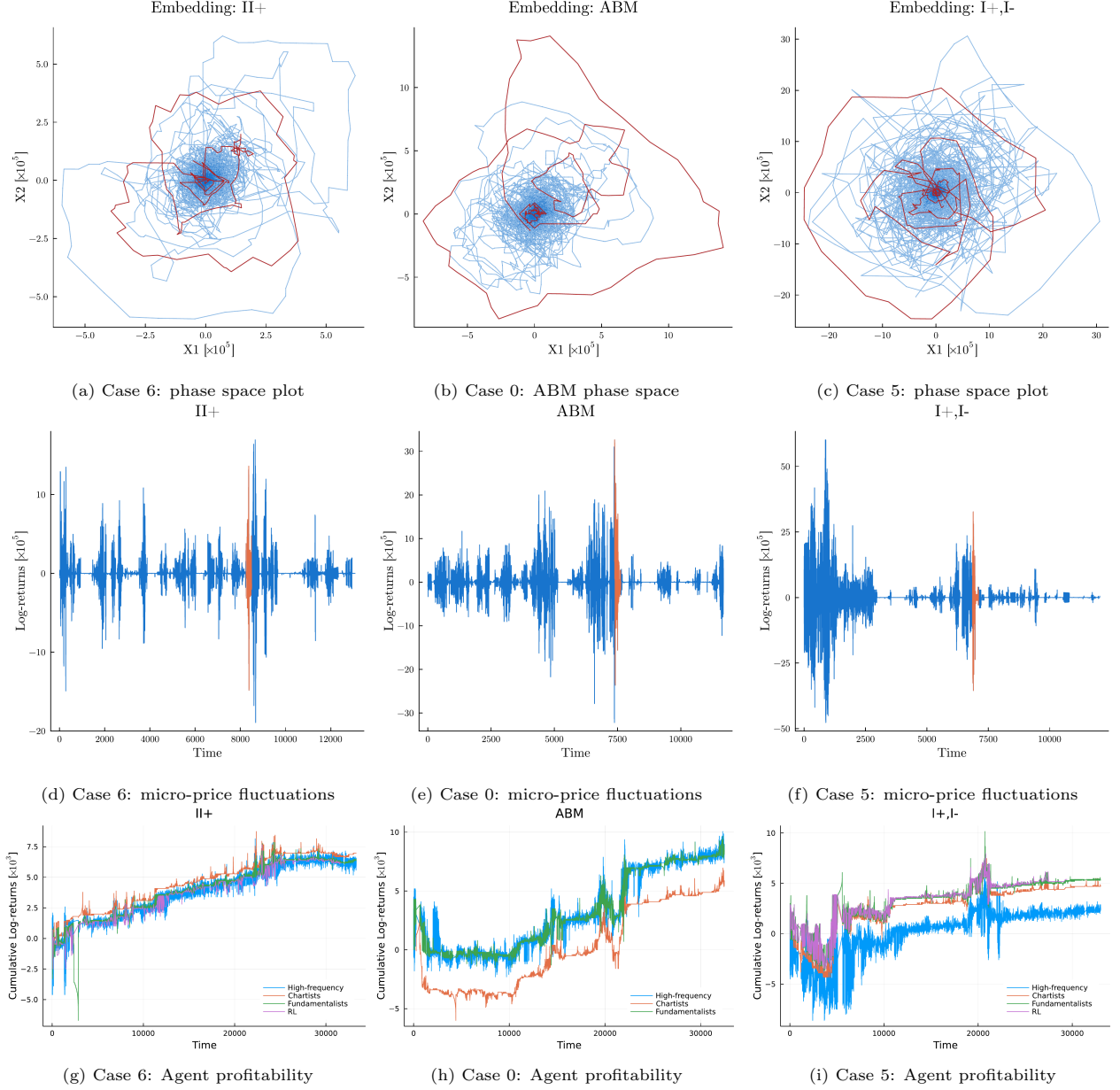
Figure 7: The phase space reconstruction plots using a 2-dimensional embedding, and a delay times $\tau = 10, 10$, and 6, respectively, for figures 7a, 7c and 7b. Segments, each of 250 points, are highlighted in red and show the dynamics associated with large micro-price movements, and these associated time-series segments are given in the second row of figures 7d, 7f and 7e. The lower row of plots are equivalent to "population" plots for the four different agent classes, the liquidity providers, the two classes of minimally intelligent liquidity takers, and the optimal execution agents where figures 7g, 7h and 7i plot the total running profit of the four agent classes. The plots are provided for Case 6, Case 0 and Case 5 as described in Table 3.

cases are *balanced* for cases 5,7,8,11 and 12, where the size of the buying position is the same as the selling position, zero. This is not the case for cases 1, 2, 3, 4, 6, 9 and 10 where there is non-zero volume exposure; the volume exposure is *one-sided*. This suggests that increases in the fractal dimension may be explained in this model to be due to asymmetric liquidity demand.

To provide context for the complexity measures we provide phase space reconstruction plots using a 2-dimensional embedding in Figure 7. Here with delay times $\tau = 10, 10$, and 6, respectively, for figures 7a, 7c and 7b. Segments,

each of 250 points, are highlighted in red and show the dynamics associated with large micro-price movements, and these associated time-series segments are given in the second row of figures 7d, 7f and 7e. We notice that there is non-random structure to the dynamics with indications of quasi-periodic orbits. The plots are noticeably distorted relative to each other. With the single learning agent (Case 6) having the most extended phase-space, and the situation with both a buying and selling learning agent (Case 5) having the most concentrated dynamics. The lower row of plots are equivalent to "population" plots for the four dif-

13
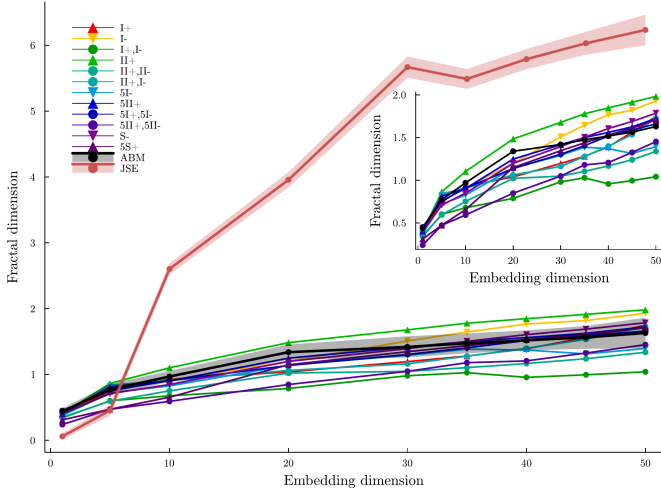
(a) The fractal (correlation) dimension as a function of the embedding dimension found using the Grassberger-Procaccio algorithm [27].

(b) Comparing the complexity of different model configurations. The single agents tend to increase the difference relative to the training environment, and the many agent configurations reduce the difference.

Case	Types	ΔD		# RL	Exp.	$\overline{\Delta D}$
6	II^+	0.319	single	1	X_0	0.130
4	I^-	0.213				
1	S^-	0.075				
3	I^+	-0.086				
9	$5II^+$	0.044	many agents	5	X_0	-0.040
2	$5S^+$	0.005				
10	$5I^-$	-0.170				
11	$5I^+, 5I^-$	-0.026		10	0	-0.152
12	$5II^+, 5II^-$	-0.278				
5	II^+,I^-	-0.101				
7	II^+,II^-	-0.341		2	0	-0.321
8	I^+,I^-	-0.520				

Figure 8: Comparing the complexity of the different configurations. There is complexity that is missing when compared to the real-world data (red) relative to the model distortions around the environment (black) introduced by the inclusion of learning agents. On the left (in Figure 8a), we have that triangles represent simulations with only either buy or sell side agents and the circles represent simulations with both buy and sell side agents, as well as the JSE and ABM.

ferent agent classes, the liquidity providers, the two classes of minimally intelligent liquidity takers, and the optimal execution agents where figures 7g, 7h and 7i plot the total running profit of the four agent classes.

5. Conclusion

Our expanded description of market ecology includes optimal execution agents, which are necessary to produce stylised facts associated with order flow and the cost of trading. We further argue that learning is necessary because markets fail when the volume traded by a given agent exceeds the carrying capacity of the environment. However, having many agents attempting to learn simultaneously can result in complex non-stationary market dynamics, preventing successful learning for any agent.

We explicitly exclude strategy switching between the fundamentalists and chartists as a model design decision in the environment in which learning and adaption take place. This is premised on the reality that in real financial markets traders, money managers and investors cannot easily change strategies, and do not in fact change strategies because of how both treasury functions in investment banks allocate risk capital to strategies and trading desk, as well as regulatory structure surrounding mandate design specification for asset management and market-making functions. Here, we only allow strategy die-out. However, the learning agents can exploit the relative die-out and emergence of particular strategies [17].

The inclusion of execution agents to a minimally intelligent ABM introduces additional heterogeneity into the observed order flow and liquidity provision processes. How these processes, and consequently the stylised facts, change

depends on the specification of the optimal execution agents. In particular, we find that: *i)* persistence in order flow increases with the number of execution agents trading on a single side, *ii)* the realised cost of trading decreases when agents can submit LOs, and *iii)* increasing the complexity of trading agents introduces an additional source of variation into the price process. These findings suggest the necessity of including optimal execution agents in ABMs to recover empirical high-frequency stylised facts.

Furthermore, we find that learning introduces further variation into the order flow and liquidity processes, as agents make state-based decisions using a decision rule that adapts over time. This is demonstrated by the decrease in the level and persistence of autocorrelations. Surprisingly, we did not find that learning decreased the average cost of trading as reflected by the realised price impact functions. However, we still find evidence that learning is feasible since execution agents increase their performance over successive training periods. Although learning agents added complexity to the market's dynamics, this was insufficient to recover the complexity observed in empirical data.

In spite of the necessity of incorporating execution agents for a realistic ABM framework, neither execution agents or learning appears to be the dominant source of financial market complexity, at least when considered for a single stock in isolation from the broader market. Thus, as future work, we think it is worthwhile to consider the interaction of at least two markets, and investigate the emergence of correlations as the driver of the missing financial market complexity. This would allow for the investigation of a new agent class: multi-asset portfolio optimising agents, and how they are situated within the existing market ecology.

14

In short, the bulk of the nonlinear dynamics, and complexity, in the single stock setting, seems to arise from the minimally intelligent agent dynamics, and it is this dynamic that provides the opportunity for learning.

Code and Data Availability

Acknowledgements

Author contributions statement

T.G. and M.D. conceived the experiments and models, M.D. implemented and conducted the experiments, A.P implemented and conducted the analysis, M.D, A.P. and T.G. analysed the results. All authors reviewed the manuscript.

Competing interests

There are no competing interest.

References

[1] Auletta, G., Ellis, G.F.R., Jaeger, L., 2008. Top-down causation by information control: from a philosophical problem to a scientific research program. J. R. Soc. Interface , 1159–1172doi:10.1098/rsif.2008.0018.

[2] Bouchaud, J.P., Farmer, J.D., Lillo, F., 2009. How markets slowly digest changes in supply and demand, in: Handbook of financial markets: dynamics and evolution. Elsevier, pp. 57–160.

[3] Bouchaud, J.P., Gefen, Y., Potters, M., Wyart, M., 2003. Fluctuations and response in financial markets: the subtle nature ofrandom'price changes. Quantitative finance 4, 176. doi:10.1080/14697680400000022.

[4] Bouchaud, J.P., Mézard, M., Potters, M., 2002. Statistical properties of stock order books: empirical results and models. Quantitative finance 2, 251–256. doi:10.1088/1469-7688/2/4/301.

[5] Bowling, M., 2000. Convergence problems of general-sum multiagent reinforcement learning, in: ICML, pp. 89–94.

[6] Brock, W.A., Hommes, C.H., 1998. Heterogeneous beliefs and routes to chaos in a simple asset pricing model. Journal of Economic Dynamics and Control 22, 1235–1274. URL: https://www.sciencedirect.com/science/article/pii/S0165188998000116, doi:https://doi.org/10.1016/S0165-1889(98)00011-6.

[7] Busoniu, L., Babuska, R., De Schutter, B., 2008. A comprehensive survey of multiagent reinforcement learning. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 38, 156–172.

[8] Cartea, Á., Jaimungal, S., Penalva, J., 2015. Algorithmic and high-frequency trading. Cambridge University Press.

[9] Chairella, C., Dieci, R., He, X., 2009. Heterogeneity, market mechanisms, and asset price dynamics, in: Hens, T., (Eds.), K.R.S.H. (Eds.), Handbook of Financial Markets: Dynamics and Evolution. Elsevier, pp. 277–344.

[10] Chakraborty, T., Kearns, M., 2011. Market making and mean reversion, in: Proceedings of the 12th ACM conference on Electronic commerce, pp. 307–314. doi:10.1145/1993574.1993622.

[11] Challet D, Z.Y.C., 1997. Emergence of cooperation and organization in an evolutionary game. Physica A 246.

[12] Challet D, Marsili M, Z.Y.C., 2000. Modeling market mechanisms with minority game. Physica A 276.

[13] Chiarella, C., He, X.Z., 2005. An asset pricing model with adaptive heterogeneous agents and wealth effects, in: Lux, T., Samanidou, E., Reitz, S. (Eds.), Nonlinear Dynamics and Heterogeneous Interacting Agents, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 269–285.

[14] Cont, R., 2001. Empirical properties of asset returns: stylized facts and statistical issues. Quantitative Finance 1, 223–236. doi:10.1080/713665670.

[15] Day, R.H., Huang, W., 1990. Bulls, bears and market sheep. Journal of Economic Behavior & Organization 14, 299–329. URL: https://www.sciencedirect.com/science/article/pii/016726819090061H, doi:https://doi.org/10.1016/0167-2681(90)90061-H.

[16] Dickey, D.A., Fuller, W.A., 1979. Distribution of the estimators for autoregressive time series with a unit root. Journal of the American Statistical Association 74, 427–431. doi:10.1080/01621459.1979.10482531.

[17] Dicks, M., Paskaramoorthy, A., Gebbie, T., 2024. A simple learning agent interacting with an agent-based market model. Physica A: Statistical Mechanics and its Applications 633, 129363. URL: https://www.sciencedirect.com/science/article/pii/S0378437123009184, doi:https://doi.org/10.1016/j.physa.2023.129363.

[18] Donier, J., Bonart, J., Mastromatteo, I., Bouchaud, J.P., 2015. A fully consistent, minimal model for non-linear market impact. Quantitative Finance 15, 1109–1121.

[19] Ellis, G.F., 2008. On the nature of causation in complex systems. Transactions of the Royal Society of South Africa 63, 69–84. URL: https://doi.org/10.1080/00359190809519211, doi:10.1080/00359190809519211, arXiv:https://doi.org/10.1080/00359190809519211.

[20] Ellis, G.F.R., 2012. Top-down causation and emergence: some comments on mechanisms. Interface Focus 2, 126–140. doi:10.1098/rsfs.2011.0062.

[21] Farmer, J.D., 2002. Market force, ecology and evolution. Industrial and Corporate Change 11, 895–953.

[22] Farmer, J.D., Gerig, A., Lillo, F., Waelbroeck, H., 2013. How efficiency shapes market impact. Quantitative Finance 13, 1743–1758.

[23] Foerster, J., Nardelli, N., Farquhar, G., Afouras, T., Torr, P.H., Kohli, P., Whiteson, S., 2017. Stabilising experience replay for deep multi-agent reinforcement learning, in: International conference on machine learning, PMLR. pp. 1146–1155.

[24] Galla, T., Farmer, J.D., 2013. Complex dynamics in learning complicated games. Proceedings of the National Academy of Sciences 110, 1232–1236.

[25] Geweke, J., Porter-Hudak, S., 1983. The estimation and application of long memory time series models. Journal of time series analysis 4, 221–238. doi:10.1111/j.1467-9892.1983.tb00371.x.

[26] Gould, M.D., Porter, M.A., Williams, S., McDonald, M., Fenn, D.J., Howison, S.D., 2013. Limit order books. Quantitative Finance 13, 1709–1742. doi:10.1080/14697688.2013.803148.

[27] Grassberger, P., Procaccia, I., 1983. Characterization of strange attractors. Phys. Rev. Lett. 50, 346–349. URL: https://link.aps.org/doi/10.1103/PhysRevLett.50.346, doi:10.1103/PhysRevLett.50.346.

[28] Hasbrouck, J., 1988. Trades, quotes, inventories, and information. Journal of financial economics 22, 229–252.

[29] Hendricks, D., Wilcox, D., 2014. A reinforcement learning extension to the almgren-chriss framework for optimal trade execution, in: 2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr), IEEE. pp. 457–464.

[30] Henry, B., Lovell, N., Camacho, F., 2000. Nonlinear Dynamics Time Series Analysis. John Wiley & Sons, Ltd. chapter 1. pp. 1–39. doi:https://doi.org/10.1109/9780470545379.

[31] Huberman, G., Stanzl, W., 2004. Price manipulation and quasi-arbitrage. Econometrica 72, 1247–1275. URL: http://www.jstor.org/stable/3598784.

[32] Hurst, H.E., 1951. Long-term storage capacity of reservoirs. Transactions of the American Society of Civil Engineers 116, 770–799. doi:10.1061/TACEAT.0006518.

[33] Jericevich, I., Chang, P., Gebbie, T., 2021. Simulation and estimation of an agent-based market-model with a matching engine. arXiv:2108.07806.

[34] Jericevich, I., Chang, P., Pillay, A., Gebbie, T., 2020. Supporting test data: Comparing the market microstructure between two south african exchanges. URL: https://zivahub.uct.ac.za/articles/dataset/Supporting_Test_Data_Comparing_the_Market_Microstructure_Between_two_South_African_Exchanges/13187591, doi:10.25375/uct.13187591.v1.

[35] Kyle, A.S., 1985. Continuous auctions and insider trading. Econometrica 53, 1315–35. doi:10.2307/1913210.

[36] Littman, M.L., 1994. Markov games as a framework for multi-agent reinforcement learning, in: Machine learning proceedings 1994. Elsevier, pp. 157–163.

[37] Lux, T., 1995. Herd Behaviour, Bubbles and Crashes. The Economic Journal 105, 881–896. URL: https://doi.org/10.2307/2235156, doi:10.2307/2235156, arXiv:https://academic.oup.com/ej/article-pdf/105/431/881/27041062/ej0881.pdf.

[38] Marsili, M., 2001. Market mechanism and expectations in minority and majority games. Physica A 299, 93–103.

[39] Massey, F.J.J., 1951. The kolmogorov-smirnov test for goodness of fit. Journal of the American Statistical Association 46, 68–78. doi:10.1080/01621459.1951.10500769.

[40] Nuyts, J., 2010. Inference about the tail of a distribution: Improvement on the hill estimator. International Journal of mathematics and mathematical sciences 2010. doi:10.1155/2010/924013.

[41] Pagan, A., 1996. The econometrics of financial markets. Journal of empirical finance 3, 15–102. URL: https://www.sciencedirect.com/science/article/pii/0927539895000208, doi:10.1016/0927-5398(95)00020-8.

[42] Pangallo, M., Heinrich, T., Farmer, J.D., 2019. Best reply structure and equilibrium convergence in generic games. Science advances 5, eaat1328.

[43] Perold, A.F., 1988. The implementation shortfall: Paper versus reality. Journal of Portfolio Management 14, 4.

[44] Polakow, D., Gebbie, T., Flint, E., (November 28, 2023). Epistemic limits of empirical finance: Causal reductionism and self-reference URL: https://ssrn.com/abstract=4646664, doi:10.2139/ssrn.4646664.

[45] Potters, M., Bouchaud, J.P., 2003. More statistical properties of order books and price impact. Physica A: Statistical Mechanics and its Applications 324, 133–140.

[46] Sanders, J.B., Farmer, J.D., Galla, T., 2018. The prevalence of chaotic dynamics in games with many players. Scientific reports 8, 1–13. doi:10.1038/s41598-018-22013-5.

[47] Tan, M., 1993. Multi-agent reinforcement learning: Independent vs. cooperative agents, in: Proceedings of the tenth international conference on machine learning, pp. 330–337.

[48] Wilcox, D., Gebbie, T., 2014. Hierarchical causality in financial economics. Available at SSRN: URL: https://ssrn.com/abstract=2544327, doi:http://dx.doi.org/10.2139/ssrn.2544327.

[49] Winker, P., Gilli, M., Jeleskovic, V., 2007. An objective function for simulation based inference on exchange rate data. Journal of Economic Interaction and Coordination 2, 125–145. doi:10.1007/s11403-007-0020-4.

[50] Zhang, K., Yang, Z., Başar, T., 2021. Multi-agent reinforcement learning: A selective overview of theories and algorithms. Handbook of Reinforcement Learning and Control , 321–384.