A network simulation of OTC markets with multiple agents

1st James T. Wilkinson
McCormick School of Engineering
Northwestern University
Evanston, U.S.A
jameswilkinson2022@u.northwestern.edu

***Abstract*—We present a model of a free-market in which dynamics, such as changes in price, result from agent-level interactions. The model is unique in its consideration of value investors, artificially intelligent trend investors and market-makers whose interactions are constrained to edges on a pre-defined network. This contribution extends the range of possible investigations to consider the effect of various market structures on price movements. First, we present the model, including an overview of the agents and the rules that govern the agent-level interactions. Second, we demonstrate that the model exhibits a number of similarities to the real world: a characteristic fat-tailed distribution of price changes whose ranks follow a power law; a skew negatively correlated to market-maker positioning; the emergence of characteristic price patterns (or "technicals") and more. Finally, we briefly explore the impact of market structure and network topology on the markets functionality, finding that markets organised as random networks have a critical point of fragmentation beyond which arbitrage becomes rapidly possible between the prices of different market makers. A discussion is provided on future work that would be beneficial.**

I. Introduction

Agent based modelling (ABM) has become a popular method for simulating financial markets since its initial use in the late 1980s [1]–[4]. Financial risk modelling intuitively lends itself to this approach. Firstly, real world markets are naturally defined by the interaction of multiple agents, each with a diverse set of functions and behaviours. Secondly, systems with many interacting sub-units often exhibit scale free phenomena, as is the fractal nature of market prices [5]. Emergent phenomena within markets is often difficult to model analytically, and whilst such models exist [6]–[8], they are inherently limited by their high-level assumptions of a predefined stochastic process governing price movements. The complexity and nature of the financial environment has encouraged the use of ABM, which typically makes few assumptions about pricing dynamics. Now a staple component of financial risk analysis [9], the multi-agent approach allows a better understanding of how certain phenomena emerges from agent-level interactions.

Agents within a financial market are able to interact both directly through trades and indirectly through observation. Prior works into agent based financial modelling have used a variety of agents and mechanisms to define the system's behavior [2]–[4], [10]–[18]. Often, these models are specialised for specific investigations, such as exploring the effect of insurance methods [1] on market crashes or increased leverage on price change kurtosis [18]. Typically, these investigations have neglected over-the-counter (OTC) systems in favor of exchange traded markets or auctions [14], [16], [17]. In part, this is because the additional requirement to model market makers, whose reactions to trades are difficult to quantify. However, market makers play a vital role in the financial system by providing additional liquidity to investors. Their function is central to OTC markets in particular, especially in low liquidity environments. We present a model with three agent types - value investors, trend investors and market makers. Value investors are assigned heterogenous and static views and uncertainties on risk. Trend investors observe the historical price information that is available to them, using a reinforcement learning approach to select actions. Market makers adjust their prices based on the trades they see happening within the market, with a parameterised sensitivity, and a functional increase in commission with inventory size.

We introduce the novel consideration of a network topology. Given that transactions within a market occur as spatial interactions, it is natural to extend an ABM market model onto a network topology. By limiting the interactions of agents to occur only along the edges of a network, we can perform investigations into the effect of market structure on pricing dynamics. We focus on modelling a random network, parameterised by the probability of a link forming between an agent, and a market maker.

In this report, we explore the ability of the proposed model to reproduce the emergent phenomena observed in real financial markets. In particular, we demonstrate that the model produces fat-tailed distributions of price changes that follow a power-law rank distribution [19] [20], that prices converge to the mean views of value investors, that the model exhibits realistic and reproducible price patterns (often referred to as "technicals"). In addition, we use the network functionality of the model to demonstrate that markets constrained to random networks exhibit a critical point beyond which market function deteriorates and arbitrage becomes rapidly possible between the prices of different market makers.

II. The Model

We use the NetLogo platform [21], [22] to simulate our agent based model. NetLogo is a multi-agent programming language and modeling environment that has seen widespread use in simulating complex phenomena across a wide range of disciplines [23]–[25]. Netlogo provides an ideal interface that allows users to intuitively adjust parameters, run experiments and visualise the phenomena being modelled. An illustration of the NetLogo layout of this model is provided in Appendix VI.

A. Model mechanics

The model comprises of three types of agent: market makers, value investors and trend investors (AI agents).

At initialisation, each value investor is given an expectation of the benchmark's value drawn from a distribution centered at a price of 100. These expectations are static, and never change. Value investors are also initialised with an uncertainty of their expectation, which we define here as being drawn from a uniform distribution between 5 and 10. The simulation begins with market-makers making prices around the initial price of 100 with a fixed bid-offer that is a model-parameter. Market-makers are also assigned a soft position limit, determined by the user. This soft limit represents the size of position a market-maker can comfortably hold without adjusting their prices to compensate for their risk.

Market-makers always advertise two prices - a bid price at which they are willing to buy, and an offer price at which they are willing to sell. The difference between these prices is referred to as the bid-offer, and is a fixed parameter determined by the user. Once a trade occurs, the price is "published", and all market makers that are connected via the network to the transacting market maker move their bid and offer prices towards the published price. If a market maker with bid-offer Δ and a mid price of P_{old} sees that a new trade occurred at price P_{trade}, then they update their price according to Eq. 1.

$$P = P_{trade}\gamma + P_{old}(1-\gamma) \tag{1}$$

Where γ represents the sensitivity of market makers to price changes, and scales between zero and unity. In order to reflect a market maker's inventory in their price, the advertised price is shifted by an amount proportional to the inventory size of the market maker.

$$P_{adv} = P - \alpha i \tag{2}$$

Market makers in our model will allow transactions at the advertised price if their net position following the trade lies within their soft limits. However, if a trade would result in a breech of their risk limits, the market makers further increase their bid-offer linearly for each unit of trade beyond this limit. To ease the calculations, we divide any trade that will breech the market-maker's soft limit into two components - a portion that takes the market maker up to their risk limit, and a portion that exceeds the limit. The former portion is executed at the advertised price P_{adv}, whilst the excess portion is executed at an adjusted price that is more favorable for the market maker. For a market maker with inventory i executing a trade of size S, they formulate the price at which they transact the portion of a trade that breeches a risk limit as

$$P = \begin{cases} P_{adv} - \alpha(i+S-L) & \textit{when buying} \\ P_{adv} + \alpha(i+S-L) & \textit{when selling} \end{cases} \tag{3}$$

where P is the updated price, P_{adv} is the market maker's advertised price, L is the market maker's soft inventory limit, and α is some positive constant.

At each tick, market makers refresh their prices and one of the investors is selected and allowed to trade. The decision mechanism is different for a value investor and a trend investor.

A value investor's action is to compare the prices of market makers, and choose to buy if the price is lower than their expectation and to sell if the price is higher than their expectation. As is typical in a real financial market, the trade size is variable, and grows as the difference between the investor's expectation and the market price increases, and also as the investor's uncertainty decreases. Specifically, investors calculate how much they want to trade as

$$S = \frac{P - E_{vi}}{\sigma_{vi}} \tag{4}$$

Where P is the optimal transaction price provided by the market makers, E_{vi} is the investor's expectation, and σ_{vi} is the investor's uncertainty. The trade size is capped to a maximum value determined by the user - an intuitive restriction found within real financial markets. Equations Eq. 3 and Eq. 4 are simultaneous equations with an analytic solution for a trade price and size that satisfies both the value investor and the market makers behavior.

$$P = P_{adv} + \alpha\left(i + \frac{E_{vi}}{\sigma_{vi}} - L\right)\left(1 - \frac{\alpha}{\sigma_{vi}}\right)^{-1} \tag{5}$$

A trend investor's action is simple to explain, in that it is always a fixed size (determined as a parameter of the model), and in a direction determined by the AI algorithm. For simplicity, the AI algorithm only determines the trend investor's action when they have zero position. Once a trend investor has a non-zero position, they will close the position on their next turn. The AI is based loosely on a deep Q network approach, and is trained to optimise the investors profit under these constraints.

At the end of each iteration, the market makers are allowed to trade versus one another in order to distribute risk between them. Each market-maker that is beyond their position limits is allowed to become an investor, and asks the other market-makers to trade a size that would restore the market-maker to within their risk-limits. The logic here is similar to before, with the trade size determined by the amount required to buy/sell to restore the market maker within their risk limits, and the price proposed by each counter-party calculated with a linearly increasing commission with trade size.

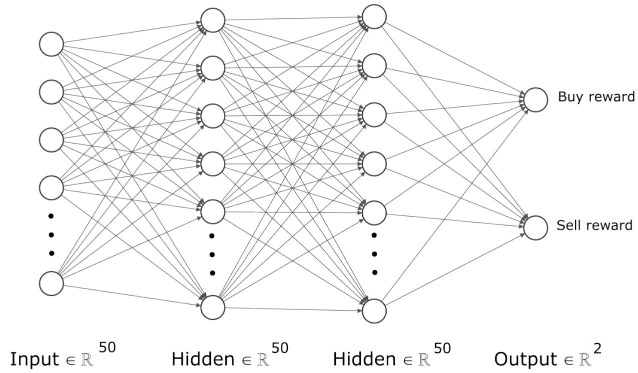

Fig. 1. Illustration of the neural network used, with two hidden layers of 50 nodes each, and two output nodes representing the expected profit for an agent buying and selling given the state.

B. Reinforcement Learning agent mechanics

The trend investors use a simplified AI approach to inform them whether to buy or sell, which is equivalent to deep Q learning with a Markov decision process of length 1. The state size is formed of the most recent price history, formatted as the mean of market-maker mid prices for each one-in-ten of the prior 500 time steps. The consideration of every 10th datapoint considerably reduces the size of the model and speeds up the training process. The model architecture is outlined in Fig. 1.

As in deep Q learning, in order to address the exploration-vs-exploitation trade off, the agents hold a parameter ϵ which represents the probability that an agent chooses an action at random. ϵ decreases proportionally at each iteration, as the agents learn. We truncate the MDP at a length of one primarily to boost the networks training speed. This is enforced by ensuring that trend investors who have accumulated a prior position will predictably close out their inventory on their next go, whilst trend investors who have no prior position will use the AI in order to make a decision to buy or sell. The state and action is cached at the trade entry point, allowing the realised reward to be calculated at the trade exit as the profit from that trade. The neural network architecture consists of two output nodes, which are trained to predict the rewards from either buying or selling given the provided state.

In certain instances with a small number of agents, the trend investors can be chosen to act twice in short spaces of time. Due to the bid-offer commission within the system, this can make a profitable strategy unattainable if investments time-frames are small. In order to mitigate this, and to standardize the duration for which a trend investor invests for, we create a user-definable model parameter which limits the minimum number of iterations a trend investor can hold their position for.

C. Network formation

Within the model, interactions are constrained to occur along the edges of a network. We define this network as random, with the constraint that every agent must be connected to at least one other market maker. Since agents can only participate in the system through links with other market-makers, this constraint has no net effect on nature of the network's structure itself. The effect of the network is to inhibit the transfer of information throughout the market. As such, market makers will not update their prices unless a new trade occurs versus a neighboring market maker on the network. Similarly, value investors will only consider directly linked market makers to obtain prices and execute trades. In addition, the state used by trend investors is uniquely defined for each instance as the mean market-maker price when considering only market-makers that it is directly connected to.

D. Expectation generation of value investors

As has been discussed, value investors are initialised with a static expectation of the market price. We distribute these value investor expectations according to a mixture model of two equivalent normal distributions with means separated by a model parameter $\Delta\mu$, which extends the model to consider a market which has split expectations of the price. This is implemented by assigning value investors an expectation calculated from Eq. 6. We assign the investor uncertainty according to a uniform distribution between 5 and 10, which represents a realistic range of uncertainty in the context of our setup (an initial price of 100, and the means of value investor expectations typically diverging between 0-20 units of 100 as is parameterised in the model).

$$E = \begin{cases} \mathcal{N}(100 + \Delta\mu, \sigma) & \text{with probability } 0.5 \\ \mathcal{N}(100 + \Delta\mu, \sigma) & \text{with probability } 0.5 \end{cases} \tag{6}$$

Such bi-modal situations are common in the real market, especially around binary events, and as such, are an important consideration of this model.

E. User-defined parameters

The model has the following list of parameters that can be determined by the user.

nValueInvestors (default 50) determines the number of value investor agents that will initialised in the model.

nSmartInvestors (default 50) determines the number of AI trend investor agents that will initialised in the model.

nMarketMakers (default 10) determines the number of market maker agents that will initialised in the model.

bid-offer (default 1) specifies the fixed interval between each market-maker's bid and offer prices.

price-sensitivity (default 100) specifies the rate at which market makers move their price when they learn of a new trade as a percentage. This is equivalent to the γ parameter in Eq. 1

mm-positionlimit (default 7) represents a soft limit to market maker inventory before adjustments are made to prices from their advertised levels. This is the parameter L in equation Eq. 5.

prob-of-link (default 100) is the probability of a link forming in our random network as a percentage.

trade-size-cap (default 3) represents the maximum trade size a value investor can make.

smartsize (default 3) represents the fixed trade size of a trend investor.

smart-holdingtime (default 250) determines the minimum trade-time for a trend investor.

market-disparity (default 20) determines the difference between the peaks of the bi-modal distributions, from which the value investors expectations are drawn, and 100. This is equivalent to the parameter $\Delta\mu$ in Eq. 6.

mm-limit-toggle (default ON) triggers the behavior whereby market makers will transact versus one another if their soft limit is breached. If this parameter is turned off, market makers do not transact against each other.

verbose (default OFF) is a debugging toggle that prints each transaction in real time.

AI-investor-model allows the user to choose between a pre-trained set of trend investors, or whether to reinitialise the trend investors from scratch and to commence the learning process when the model is run. The pre-trained models are trained using the default settings outlined above, and performance could be sub optimal if using these pre-trained models in a market with different parameters.

The model also includes the following additional functionalities.

Force Market Makers Short is a button which instantaneously positions all market makers to be beyond their soft limit. This can be used to demonstrate certain technicals within the market.

Market Crash is a button which instantaneously recalibrates all value investor expectations to be 20% lower than they were previously.

Remove Value Investors is a button which removes all value investors from the system, even mid-simulation. This allows for training of the RL algorithm in the trend investors before removing them entirely, should the user want to explore these possibilities.

III. Validation of the Model

We examine the effectiveness of the model through a number of statistical tests that are prevalent throughout the literature [19], [20]. Unless specified otherwise, we use the default parameters for the model outlined in II-E, training the trend investors AI from scratch and only considering results from where all AI investors are fully trained. Specifically, we look for quantitative features in the price distribution that reflects the fat-tailed distributions observed in nature, such as a kurtosis above 3 and a Zipf power-law between price movement sizes and ranks. We additionally illustrate a number of qualitative features exhibited by the model that highlight further similarities to reality.

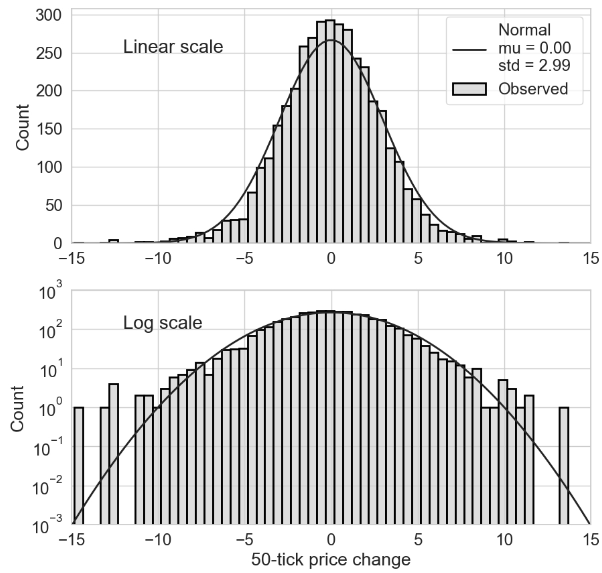

Fig. 2. Price changes, measured every 50 iterations of the model, illustrated with a fitted normal distribution. Shown both on a linear scale (upper) and a logarithmic scale (lower).

Firstly, the distribution of price changes within the model can be shown to follow fat-tailed distributions, as illustrated in Fig 2.

The model produces price returns that are inadequate to be described with a normal distribution. In particular, the tails of the distribution have higher populations, which we can describe with the kurtosis statistic. Kurtosis above 3 implies a distribution with higher population in the tails - our model consistently produces price distributions with kurtoses above 3. It should be noted that normal distributions are entirely plausible in the financial market, as they are in our model implementation should we limit the simulation to only include value investors, high market maker soft limits, and no trend investors. We also note the nature of the distribution tails, which becomes clear in the lower plot of Fig. 2. As the tails deviate from the normal distribution, they become approximately linear in the logarithmic scale (a power law). This behaviour has been noted to exist in the observed price movements of the Dow Jones [20].

We can also show that the price changes follow a power law, also known as Zipf's law [26]. Although power laws are found througout nature, particularly in real multi agent systems, this is a comparison which has not been included in the previous literature on agent-based models. Zipf's law has been demonstrated to apply in practice to stock price movements [19], and it is intuitive that our model should replicate this phenomena.

We also find that our model exhibits some qualitative behaviour that is desirable. For example, the skew within the pricing distribution is negatively proportional to the position-

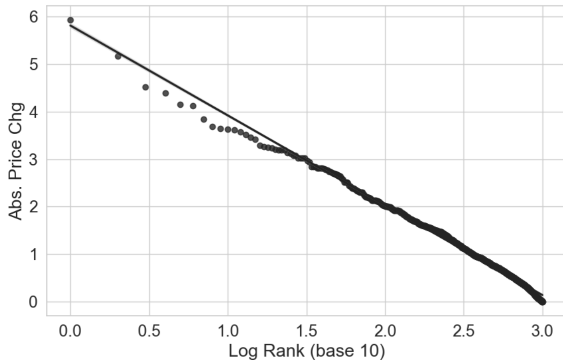

Fig. 3. A typical rank distribution of price changes within the model, illustrated on a logarithmic axis. The linear line implies a power law distribution (Zipf's law).

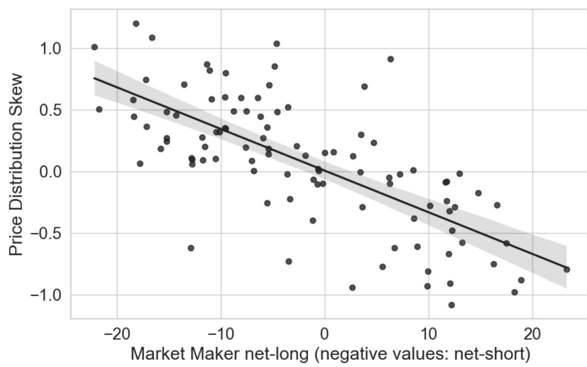

Fig. 4. Market maker net positioning (positive equates to long) compared to the skew of the resulting price distribution. The market maker positioning is calculated as an average across the duration of the simulation, and the pricing distribution comprises the cumulative changes over each successive 50 iterations of the model.

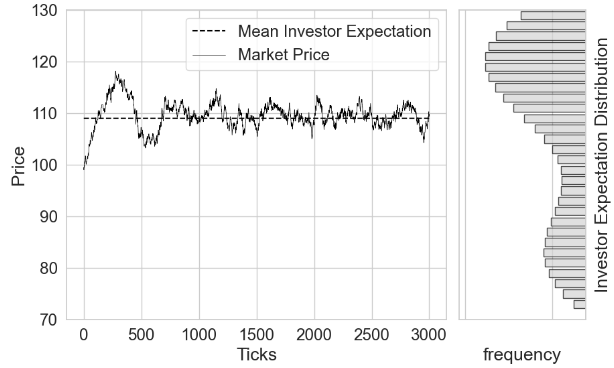

Fig. 5. (Left) Price history over 3000 iterations (solid) compared to the mean value investor expectation (dashed). (Right) Twin-peaked distribution representing value investor expectations.

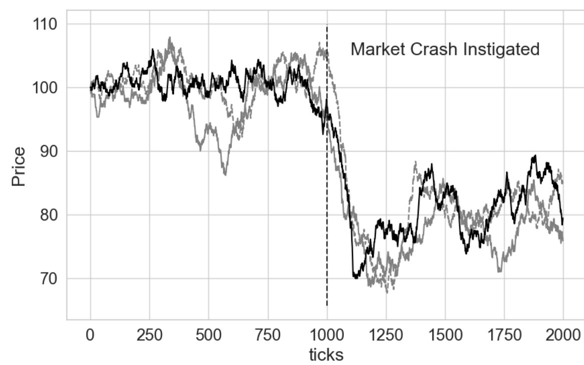

Fig. 6. Price history when a market crash is instigated after 1000 iterations of the simulation (vertical dashed) shown in four instances. Notice the characteristic overshoot followed a rebound, and the increased volatility following the crash compared to before.

ing of market makers. Whilst data on market maker positioning is highly confidential, it is impossible to make quantitative comparisons to data, however, this relationship is both intuitive and is found to anecdotally exist amongst professionals.

Systems where market-makers are positioned long will produce price distributions with a negative skew, and positive-skew distributions result when market-makers are generally positioned short. We can see this when measuring the skew of a distribution after a large number of time-steps versus the average market-maker position during the episode, as is shown in Fig. 4.

We also note the tendency of the market price to settle at-or-close-to the numerical mean of value investor expectations. This feature is often observed in real financial markets, with market participants routinely interpreting price levels between binary outcomes as an indicator of probability. This feature is a necessary component of the efficient market hypothesis, and in this regard, the market acts as an ensemble model with a mean estimator across its participants. As in real financial markets, this property holds the majority of the time, although there are situations where the market price can diverge largely from this mean expectation and the behaviour is lost.

It is noteworthy that the price within the system converges to the mean regardless of the distribution of value investor expectations. As is illustrated in Fig. 5, the price quickly converges to the mean view in a system where value investor expectations are drawn from a bimodal mixture distribution with two peaks.

We also note the presence of certain repeatable patterns that predictably emerge in the model. For example, during large price movements, the market prices typically overshoot the equilibrium level before retracing a smaller amount.

We induce a market crash by instantaneously reducing the expectations of value investors by 20%. As can be seen in Fig. 6, the model exhibits a characteristic rebound following the initial overshoot of the price. The overshoot occurs entirely through a combination of market maker positioning as they overshoot the equilibrium price in their plight to recycle inventory, along with trend investor strategy. However, as is typically the case, value investors once again restore the mar-

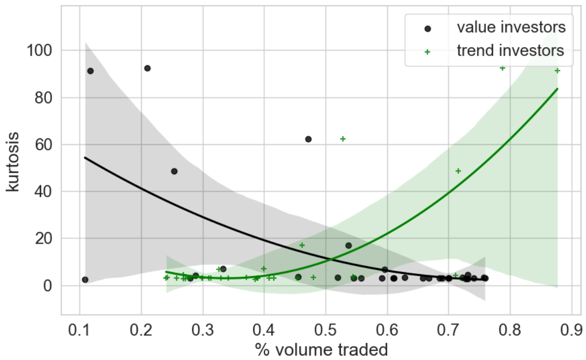

Fig. 7. Whilst broadly robust, a number of anomalies represented by extremely high kurtosis exist. The anomalies exist in environments where value investor trades make up a very small proportion of the system's total volume, and where trend investor trades make up a very large proportion of the system's total volume.

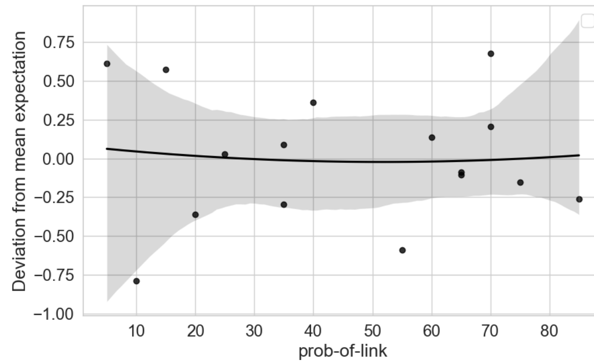

Fig. 8. The accuracy of the market in converging to the mean value investor expectation shows little dependence on the sparsity of the random network, even when the probability of a link between an agent and a market maker is extremely low.

ket price to be range bound around the new mean expectation. It is also noteworthy that the scale of price movements is visibly increased following the induced market crash. This demonstrates the auto-correlation and clustering of volatility that the model exhibits. This is also a well known feature present in the real financial markets.

IV. The Effect of Market Structure on Price Action

We have the necessary setup to explore the effect of various market structures on the resulting pricing distribution. We are able to explore the impact of bid-offer spread, different proportions of value investors to trend investors, degrees of network fragmentation and more.

Firstly, we explore the impact of trade imbalance between value and trend investors. We have already seen the effect of the intuitive result that value investors apply a normalising force to prices within the market that effectively bound the price level a range. However, our implementation (reflecting real markets) have a maximum trade size, which limits the normalising effect of value investors. It is therefore entirely possible that the effect of high-volume transacted by trend investors could outweigh the normalising effect of the value investors, resulting in a divergence of stock prices and a reduction of market efficiency.

Finally, we utilise the model's network functionality to test the effectiveness of the market for varying levels of fragmentation. Intuition dictates that the ability of the market to converge to the mean expectation of value investors, as demonstrated in Fig. 5, would deteriorate as the market becomes more fragmented.

It is therefore surprising that the average market maker price remains extremely efficient in finding the mean expectation, even in extremely sparse random networks with few connections. . 8 illustrates this behaviour, measuring the mean absolute difference between all market maker mid-prices and the mean value investor expectation. However, whilst

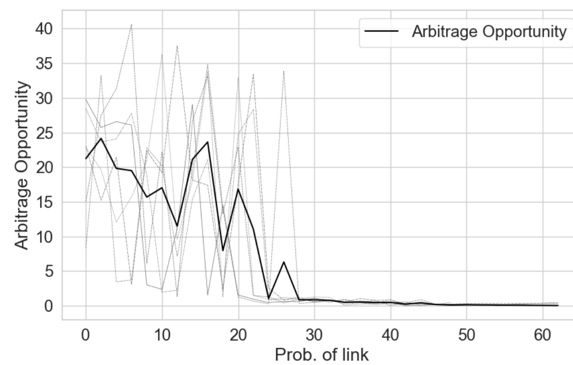

Fig. 9. Below a link probability of approximately 25% in the market's network, arbitrage opportunities rapidly grow in size.

not visible in Fig. 8, for lower link probabilities, market maker prices diverge significantly, and thus, while the average market maker price consistently reflects the mean expectation of value investors, the market functionality is impaired by the fragmentation.

To illustrate this, we focus on the arbitrage opportunity within the market for varying levels of network density. The relationship displays a critical point within the system, which for the default parameters outlined in II-E, is when the random network has links formed with a probability of below 25%.

V. Discussion

We have presented a novel agent based model for simulating the financial markets, that is unique in its consideration of network constraints, and its implementation of value investors, trend investors and market makers. The models mechanics have been outlined, and demonstrated to replicate a wide range of features that are observed in the real financial markets. For example, price changes follow fat-tailed distributions that follow Zipf's law when ranked [19], and a power distribution attenuation in the tails [20]. The model also reproduces a number of qualitatively desirable features, such as a negative correlation between market maker positioning and the skew of the price change distribution, a tendency to converge to the mean view of investors, and the exhibition of characteristic and repeatable price patterns. Evidence for volatility clustering has also been shown.

We then progressed to examine the effect of network topology on the market function. This is a particularly relevant line of examination in OTC markets which often have lower visibility and liquidity than exchange traded environments. OTC markets can also be characterised by the prevalence of market makers as is in our model.

We also illustrated that markets constrained to random networks exhibit a critical point of fragmentation, beyond which market function quickly deteriorates. We illustrate this by considering the arbitrage opportunity that arises from different market maker prices, and found that networks generated with fewer than one-in-four of the possible connections between agents and market makers have significant arbitrage opportunity that rapidly rises as the network becomes increasingly sparse.

There is room for further work, particularly in the exploration of various network topologies. It is most intuitive to model the market using a small-world network, which can be generated directly [27], and allows further analysis of various critical points that exist within the small-world literature and their effect on financial prices [28], [29].

Advancements could also be made to the algorithm governing trend investors. Currently, the implementation is based on deep Q learning with the simplification of truncating the MDP to a length of 1. This dramatically improves the learning speed, however, limits the scope of the trend investors' behaviour. This could be altered to a complete deep Q learning implementation that also extends to allow the trend investors to transact in various sizes as opposed to the fixed trade size that is currently implemented. Investigations could be carried out into the diversity of behaviours that each individual trend investor agent learns - it is possible that different agents are learning different strategies - such as momentum or mean reversion trading. It is also feasible that these agents could learn these strategies without the presence of value investors within the system - an exploration that has not yet been undertaken. Finally, the market makers could also be adapted to be artificially intelligent. It is particularly relevant that a market maker's bid-offer spread (the difference between their buy and sell prices that represents the market commission) is not always constant in the real world, but varies particularly with volatility. AI market makers could have the ability to choose their bid offer spread independently, with the goal of maximising their profit in a competitive environment.

References

[1] G.-r. Kim and H. M. Markowitz, "Investment rules, margin, and market volatility," *The Journal of Portfolio Management*, vol. 16, no. 1, pp. 45–52, 1989.

[2] R. Palmer, W. Brian Arthur, J. H. Holland, B. LeBaron, and P. Tayler, "Artificial economic life: a simple model of a stockmarket," *Physica D: Nonlinear Phenomena*, vol. 75, no. 1-3, pp. 264–274, 1994.

[3] E. Samanidou, E. Zschischang, D. Stauffer, and T. Lux, "Agent-based models of financial markets," *Reports on Progress in Physics*, vol. 70, no. 3, pp. 409–450, 2007.

[4] M. Levy, "Stock market crashes as social phase transitions," *Journal of Economic Dynamics and Control*, vol. 32, no. 1, pp. 137–155, 2008.

[5] Mandelbrot, Fisher, and Calvet, "A multifractal model of asset returns," *Cowles Foundation discussion paper*, 1997.

[6] P. S. Hagan, D. Kumar, A. Lesniewski, and D. Woodward, "Arbitrage-free sabr," *Wilmott*, vol. 2014, no. 69, pp. 60–75, 2014.

[7] P. S. Hagan, D. Kumar, and A. S. Lesniewski, "Smile risk,"

[8] B. Bartlett, "Hedging under sabr model," *Wilmott magazine*, vol. 4, pp. 2–4, 2006.

[9] M. H. Hassan, S. A. Mostafa, A. Mustapha, M. H. Abd Wahab, and D. M. Nor, "A survey of multi-agent system approach in risk assessment," in *2018 International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 1–6, IEEE, 2018.

[10] S. Vanfossan, C. H. Dagli, and B. Kwasa, "An agent-based approach to artificial stock market modeling," *Procedia Computer Science*, vol. 168, pp. 161–169, 2020.

[11] M. Amine Souissi, K. Bensaid, and R. Ellaia, "Multi-agent modeling and simulation of a stock market," *Investment Management and Financial Innovations*, vol. 15, no. 4, pp. 123–134, 2018.

[12] K. Kizaki, T. Saito, and A. Takahashi, "Equilibrium multi-agent model with heterogeneous views on fundamental risks," *SSRN Electronic Journal*, 2021.

[13] R. Grothmann, "Multi-agent modelling based on artificial neural networks," *PhD Thesis*.

[14] B. Tth, E. Scalas, J. Huber, and M. Kirchler, "The value of information in a multi-agent market model," *The European Physical Journal B*, vol. 55, no. 1, pp. 115–120, 2007.

[15] N. F. Johnson, D. Lamper, P. Jefferies, M. L. Hart, and S. Howison, "Application of multi-agent games to the prediction of financial time series," *Physica A: Statistical Mechanics and its Applications*, vol. 299, no. 1-2, pp. 222–227, 2001.

[16] T. Preis, S. Golke, W. Paul, and J. J. Schneider, "Multi-agent-based order book model of financial markets," *Europhysics Letters (EPL)*, vol. 75, no. 3, pp. 510–516, 2006.

[17] P. Jefferies, M. Hart, P. Hui, and N. Johnson, "From market games to real-world markets," *The European Physical Journal B*, vol. 20, no. 4, pp. 493–501, 2001.

[18] S. Thurner, J. D. Farmer, and J. Geanakoplos, "Leverage causes fat tails and clustered volatility," *SSRN Electronic Journal*, 2010.

[19] K. Kim, S. M. Yoon, and K. H. Chang, "Power law distributions for stock prices in financial markets," 2004.

[20] M. Warusawitharana, "Time-varying volatility and the power law distribution of stock returns," 2016.

[21] S. Tisue and U. Wilensky, "Netlogo: A simple environment for modeling complexity," in *International conference on complex systems*, vol. 21, pp. 16–21, Boston, MA, 2004.

[22] U. Wilensky and W. Rand, *An introduction to agent-based modeling: modeling natural, social, and engineered complex systems with NetLogo*. Mit Press, 2015.

[23] U. Wilensky, "Netlogo flocking model," *Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL*, 1998.

[24] U. Wilensky, "Netlogo wolf sheep predation model," 1997.

[25] U. Wilensky and K. Reisman, "Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories—an embodied modeling approach," *Cognition and instruction*, vol. 24, no. 2, pp. 171–209, 2006.

[26] G. K. Zipf, "Human behaviour and the principle of least effort,"

[27] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world'networks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[28] M. E. Newman and D. J. Watts, "Renormalization group analysis of the small-world network model," *Physics Letters A*, vol. 263, no. 4-6, pp. 341–346, 1999.

[29] A. Barrat and M. Weigt, "On the properties of small-world network models," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 13, no. 3, pp. 547–560, 2000.

VI. APPENDIX

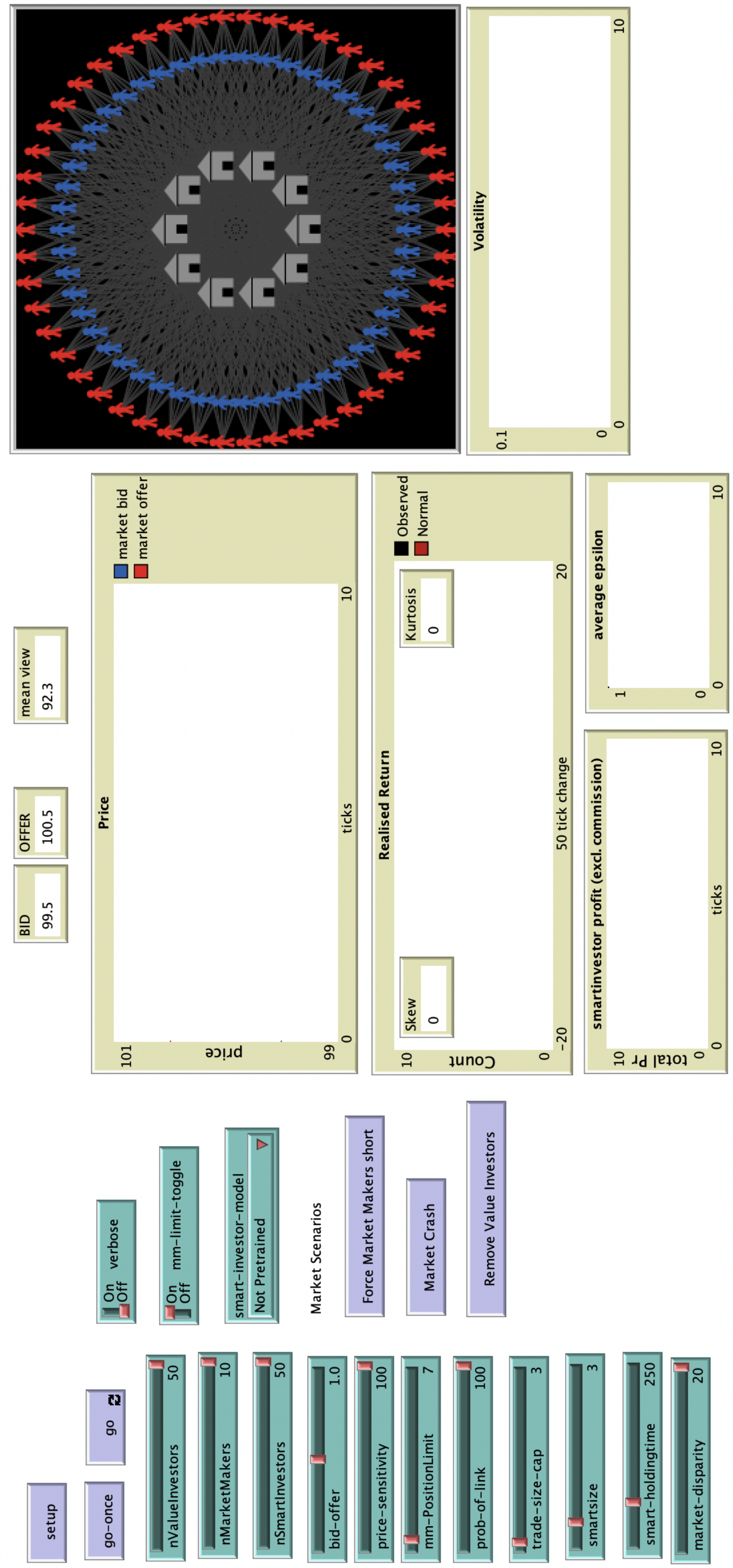

Fig. 10. An illustration of the NetLogo environment. The user parameters and their default settings used throughout this paper are visible in the green sliders and buttons.