The Effect of Liquidity on the Spoofability of Financial Markets

Anri Gu
University of Michigan
United States of America
anrigu@umich.edu

Yongzhao Wang
University of Liverpool
United Kingdom
The Alan Turing Institute
United Kingdom
yongzhao.wang@liverpool.ac.uk

Chris Mascioli
University of Michigan
United States of America
cmasciol@umich.edu

Mithun Chakraborty
University of Michigan
United States of America
dcsmc@umich.edu

Rahul Savani
University of Liverpool
United Kingdom
The Alan Turing Institute
United Kingdom
rahul.savani@liverpool.ac.uk

Theodore L. Turocy
University of East Anglia
United Kingdom
The Alan Turing Institute
United Kingdom
t.turocy@uea.ac.uk

Michael P. Wellman
University of Michigan
United States of America
wellman@umich.edu

Abstract

We investigate the relationship between market liquidity and spoofing, a manipulative practice involving the submission of deceptive orders aimed at misleading other traders. Utilizing an agent-based market simulator, we model markets with varying levels of liquidity, adjusting the spread and intervals of a market maker's orders to control liquidity. Within these simulated markets, we evaluate the effectiveness of two novel spoofing strategies against a benchmark approach. Our experiments show that in high-liquidity markets, spoofing is substantially less profitable and less detrimental to other traders compared to their low-liquidity counterparts. Additionally, we identify two distinct spoofing behavior regimes based on liquidity, each of which employ drastically different profit-making strategies. Finally, building on our quantitative findings, we identify and expound upon the mechanisms through which liquidity mitigates market manipulation.

Keywords

Market Manipulation, Spoofing, Liquidity, Agent-Based Simulation

ACM Reference Format:

Anri Gu, Yongzhao Wang, Chris Mascioli, Mithun Chakraborty, Rahul Savani, Theodore L. Turocy, and Michael P. Wellman. 2024. The Effect of Liquidity on the Spoofability of Financial Markets. In *5th ACM International Conference on AI in Finance (ICAIF '24), November 14–17, 2024, Brooklyn, NY, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3677052.3698634

ICAIF '24, November 14–17, 2024, Brooklyn, NY, USA

ACM ISBN 979-8-4007-1081-0/24/11
https://doi.org/10.1145/3677052.3698634

1 Introduction

Electronic trading platforms have reshaped the financial market landscape, enabling the automation of trading and significantly increasing transaction volume and speed. Automated traders now possess an extraordinary ability to gather and exploit market information from a plethora of sources, including transactions and order book data provided by multiple market mechanisms. While these advancements can improve price discovery and market efficiency, they may also enable advanced versions of disruptive practices such as market manipulation. For example, automation could enhance implementations of *spoofing*: placing spurious orders to create apparent order book imbalances. This generates false impressions of relative demand and supply, influencing the behavior of traders who track the order book as part of their decision-making processes. Spoofers then exploit the market movements for profit before canceling their spurious orders.

One of the most notorious cases of spoofing was that of Navinder Singh Sarao, an independent high-frequency trader who placed $200 million worth of orders betting that the market would fall and then swiftly modified and/or canceled these orders 19,000 times [22]. This manipulative tactic distorted market prices and created artificial price fluctuations, allowing Sarao to profit at the expense of other traders, ultimately netting him $40 million in profit.

Though spoofing has been ruled illegal in the U.S. and many other countries, it continues to be widespread in practice due to its profitability and the inherent difficulties of determining *manipulative intent* behind a trader's actions [12]. Therefore, thoroughly understanding spoofing mechanisms and behavior under various market conditions is essential for aiding regulatory bodies develop more effective manipulation detection and prevention strategies.

In this study, we investigate the effects of market liquidity on spoofing within an agent-based simulator, aiming to identify the types of markets most vulnerable to spoofing and the behaviors of spoofers in these environments. We control market liquidity by introducing a *market maker* (MM) who continually maintains offers to buy and sell. By adjusting the spread and intervals of the MM's orders, we can modulate market liquidity. Simulations include two types of background traders: zero intelligence (ZI) agents, who disregard the order book, and heuristic belief learning (HBL) agents, who leverage the order book to predict price movements and can be affected by spoofing. We then examine the performance of three spoofing strategies under different liquidity levels: a baseline strategy, a reinforcement learning-derived (*R-Learned*) strategy, and a static, tuned strategy with optimized parameters. The R-Learned and tuned strategies, which we developed, show strong profitability and have a more detrimental impact on the market, whereas the baseline strategy, as described by Wang et al. [25], only exhibits moderate profitability.

Through comprehensive experiments, we find that market liquidity significantly influences both the performance and behavior of spoofers. In highly liquid markets, the environment is more resilient and background traders are less susceptible to spoofing, making it difficult for spoofers to profit. Consequently, spoofers tend to lower their profit expectations and behave more like regular traders. In contrast, in less liquid markets, spoofers can fully exploit background traders and achieve significant profits. Additionally, we observe that the spoofer's arrival rate is crucial for successful spoofing, especially in high liquidity markets. A slow arrival rate increases the risk of unintended execution of spoofing orders and reduces the ability to promptly adjust to market conditions. Accordingly, we demonstrate how market liquidity affects these factors and show that the impact of arrival frequency on spoofer performance varies with market liquidity. Finally, we utilize an analytical framework to elaborate upon the mechanisms through which market liquidity inherently mitigates the effects of spoofing.

In summary, the contributions of this work include:

(1) Developing advanced spoofing strategies using RL and parameter optimization, which surpass existing baseline strategies in both profitability and market impact;
(2) Demonstrating that market liquidity significantly affects the performance and behavior of spoofers, with highly liquid markets being more resistant to spoofing while less liquid markets experience more severe impacts;
(3) Examining the effect of a spoofer's arrival rate on the market and the spoofer itself, revealing that frequent arrivals are critical in highly liquid markets but less significant in low liquidity markets;
(4) Providing a comprehensive analysis of the mechanisms through which spoofing affects the market and how market liquidity inherently mitigates the effects of spoofing.

2 Related Work

2.1 Spoofing in Financial Markets

The literature on spoofing and its impact on financial markets is relatively limited, with the majority of existing research relying on historical market data analysis [21]. Lee et al. [12] conducted one of the pioneering spoofing studies, modeling spoofing using data from the Korea Exchange. Their work provided insights into how spoofing influences market price movements and the scale of the order-book imbalances that spoofing creates. Wang [27] expanded upon their work and investigated spoofing in an index futures market in Taiwan, identifying characteristics of the strategy, its profitability, and its real-time impact. Cartea et al. [4] utilized a mathematical framework to derive the optimal spoofing strategy given a set of historical NASDAQ data. The strategy trades off the benefits from spoofing and the potential penalties the manipulator may receive from being caught in executing the nefarious practice. Cartea et al. [3] further utilized historical data to analyze the conditions under which learning market makers unintentionally adopt manipulative behavioral patterns; in contrast, in our model, we define our spoofing strategies within a predefined manipulative action space, enabling a direct comparison and characterization of spoofing behavior across different market conditions.

A few studies have also utilized machine learning to model spoofing [2, 15, 26]. Notably, Martínez-Miranda et al. [13] examined spoofing through the framework of reinforcement learning, modeling the manipulative practice as a Markov Decision Process (MDP). They determined liquidity to be a favorable condition for spoofing because it facilitates more rapid trades and thus higher transition probabilities in the MDP. In contrast, our study investigates the influence of liquidity on manipulation across all aspects of market dynamics.

Our work is most closely related to a line of investigation by Wang et al. [25], which uses an *agent-based model* (ABM) populated with diverse trading agents to simulate spoofing and its general impact on financial markets.

2.2 Other Types of Market Manipulation

Various forms of market manipulation beyond spoofing have been investigated in existing research. One notable example is *benchmark manipulation*, the practice of intentionally distorting financial reference rates or indices used to price various summary statistics [1, 7, 8, 11, 17, 23].

In particular, Shearer et al. [19] employed an augmentation of our ABM framework to study benchmark manipulation and illustrate the potential for deep RL agents to find manipulative practices when given the sole goal of profit-maximization.

Automating market manipulation at scale has also been the subject of a body of work; for example, Yagemann et al. [28] demonstrated the feasibility of stock market manipulation via botnet hijacking of brokerage accounts using an ABM that incorporates SEC data. Stenfors et al. [20] investigated the practice of *cross-market spoofing*, the act of exploiting the interconnected nature of different markets. They demonstrated the viability of this manipulative strategy in some paired currency exchange markets and the significant challenges related to its detection due to its complex, multi-market nature.

3 Agent-Based Simulation Environment

3.1 Market Model

Our market model extends those used in previous studies [19, 25]. It employs a continuous limit order book (LOB) mechanism over a single security. Time progresses in integer timesteps, $t \in [1, T]$. Agents in the model submit buy (sell) limit orders, specifying the maximum (minimum) price they are willing to pay (accept) and the number of units they wish to trade.

3.1.1 Fundamental Value Process. The dynamic intrinsic value of the security is determined by a mean-reverting stochastic process:

$$r_t = max\{0, \kappa\bar{r} + (1-\kappa)r_{t-1} + u_t\}; r_0 = \bar{r},$$

where r_t is the *fundamental value* at time t; the mean-reversion rate $\kappa \in [0,1]$ captures the tendency of the fundamental to revert to its mean $\bar{r}$; $u_t \sim \mathcal{N}(0, \sigma_s^2)$ represents a random shock applied to the fundamental at each step, σ_s^2 being the shock variance. The *realized* value of the security at the end of the finite horizon T is given by r_T.

In addition, by observing the current fundamental, agents can compute $\tilde{r}_t$, their estimate of r_T:

$$\tilde{r}_t = (1 - (1-\kappa)^{T-t})\bar{r} + (1-\kappa)^{T-t} r_t. \tag{1}$$

3.1.2 Limit Order Book. The market tracks outstanding orders using a *limit order book*. The book has two sides organized as priority queues: the buy (sell) side prioritizes orders by higher (lower) limit prices and earlier arrival times t, further ties being broken uniformly at random. BID_t (ASK_t) denotes the highest buy offer (lowest sell) price at timestep t. When orders are transacted or withdrawn, the order book is updated instantly by appropriate order removals or quantity adjustments.

3.1.3 Arrival Process. Agents (re-)arrive to the market following a Poisson process with an agent-specific arrival rate λ_a. Upon each arrival, an agent first withdraws its previous outstanding orders from the order book and then submits new limit order(s) based on its observations and its trading strategy.

3.2 Trading Strategies

We consider four types of market participants: background traders further divided into *zero intelligence* (ZI) and *heuristic belief learning* (HBL) agents; a market maker (MM); and a spoofer.

3.2.1 Background Agents. The net position q (positive for long, negative for short) of a background trader is an integer representing the number of units of the security held and is constrained to be at most $q_{\max}$ in magnitude. Each background trader i has an individual *private value* expressed as a vector Θ_i of length $2q_{\max}$:

$$\Theta_i = [\theta_i^{-q_{\max}+1}, \ldots, \theta_i^{-1}, \theta_i^0, \theta_i^1, \ldots, \theta_i^{q_{\max}}],$$

where θ_i^q is the incremental private benefit forgone by *selling* one unit (alternatively, θ_i^{q+1} is the marginal private gain from *buying* one additional unit) given the current position q. We construct Θ_i by generating $2q_{\max}$ samples independently from the distribution $\mathcal{N}(0, \sigma_{PV}^2)$ and sorting them in descending order, thus ensuring diminishing marginal returns. That is, $\theta^{q'} \leq \theta^q$ for all $q' \geq q$.

Moreover, upon every arrival, a background trader is designated to be a buyer or seller with equal probability and places a new order of unit quantity. ZI and HBL traders differ in their strategy of determining the limit price.

ZI Agents. The ZI trading strategy was originally introduced by Gode and Sunder [10]. Upon each arrival, our ZI agent observes the current fundamental r_t and uses it to compute an estimate $\tilde{r}_t$ of the final fundamental via Equation (1). Its limit price is a combination of this estimate, its private value, and a random offset $o \sim \mathcal{U}(R_{\min}, R_{\max})$ where $R_{\min}$ and $R_{\max}$ are lower and upper bounds on the ZI agent's demanded surplus. For agent i with current position q, this price is given by

$$p_i = \tilde{r}_t + \begin{cases} \theta_i^{q+1} - o & \text{if buying,} \\ \theta_i^q + o & \text{if selling.} \end{cases}$$

HBL Agents. Unlike ZI agents, an HBL agent examines historical order and transaction patterns to estimate the probability of execution for various limit prices [9]. Upon each arrival, it then updates its observations and computes the limit price that maximizes the expected surplus it will earn from a unit-quantity order. Specifically, based on data collected from the latest $\mathcal{L}$ transactions, where $\mathcal{L}$ represents an HBL agent's memory capacity, the estimated probability that an order placed at timestep t at price p will result in a successful transaction is expressed as a belief function $f_t(p)$ as follows:

$$f_t(p) = \begin{cases} \frac{TBL_t(p)+AL_t(p)}{TBL_t(p)+AL_t(p)+RBG_t(p)} & \text{if buying,} \\ \frac{TAG_t(p)+BG_t(p)}{TAG_t(p)+BG_t(p)+RAL_t(p)} & \text{if selling.} \end{cases} \tag{2}$$

In Equation (2), each term in the numerator and denominator represents the number of orders of a specific type in memory, and we use abbreviations to denote these types. T and R stand for transacted and rejected orders; A and B represent sell and buy orders; L and G indicate orders with prices at most and at least p, respectively. The rejected value of an order is equal to $min(alive \cdot \lambda_b, 1)$, where λ_b is the agent's arrival rate. The *alive* time is measured as the difference between the time of order submission and one of 3 endpoints depending on the corresponding event: the order transaction time (if transacted), the order withdrawal time (if inactive), or the current time (if active). For further details, we refer the reader to Wang et al. [25].

The HBL agent i at step t then selects a surplus-maximizing limit price as given by:

$$p_i^*(t) = \begin{cases} \arg\max_p (\tilde{r}_t + \theta_i^{q+1} - p) f_t(p) & \text{if buying,} \\ \arg\max_p (p - \theta_i^q - \tilde{r}_t) f_t(p) & \text{if selling.} \end{cases}$$

If upon arrival an HBL agent finds that fewer than $\mathcal{L}$ transactions have occurred or if either side of the order book (buy or sell) is empty, it defaults to the ZI strategy. However, these cases are infrequent, so the change in strategy does not materially impact the HBL agent's overall performance.

3.2.2 Market Maker. An MM submits limit orders on both sides of the market simultaneously. Our MM uses a parameterized *ladder* strategy [5, 24] described as follows. Upon (re-)arriving and withdrawing all of its previous orders, the MM submits a new series of unit-quantity orders arranged in a ladder structure with K rungs spaced ξ ticks apart. The ladder starts at limit price $B_t = \tilde{r}_t - \omega/2$ on the buy side and $S_t = \tilde{r}_t + \omega/2$ on the sell side, where ω is the offset parameter set by the MM. If needed, the ladder is truncated to ensure that none of the MM orders immediately executes with the best bid or ask. If truncated, the ladder is given by

$$\begin{cases} [B_t - K\xi, \ldots, B_t - (K-x+1)\xi, B_t - (K-x)\xi] & \text{if } B_t > ASK_t \\ [S_t + (K-x)\xi, S_t + (K-x+1)\xi, \ldots, S_t + K\xi] & \text{if } S_t < BID_t, \end{cases}$$

where the integer $x > 0$ represents the rung immediately above BID_t for sell orders and below ASK_t for buy orders. That is, x satisfies $B_t - (K - x)\xi < ASK_t < B_t - (K - x - 1)$ for buy orders and $S_t + (K - x - 1)\xi < BID_t < S_t + (K - x)\xi$ for sell orders.

3.2.3 Spoofer. Our model includes a spoofer that attempts to manipulate the buy side but can be trivially extended to symmetrically manipulate the sell side. At each entry, the spoofer places two orders: a large-quantity limit buy and a unit-quantity limit sell. The large-quantity bid, which we call the *spoofing order*, creates the appearance of high demand in the market. As a result, HBL agents place bids at higher prices, driving the market price up (see Section 8.1 for analysis on this mechanism). The sell order then takes advantage of this artificial interest. In our experiments, we fix the spoofing order quantity at 200 and its limit price (denoted p_{SP}) at one tick below the best bid (i.e., $BID_t - 1$), but we investigate three spoofing strategies that differ in how they determine the price of the spoofer's sell order.

Baseline Spoofer. This strategy, following the spoofing policy specified by Wang et al. [25], sets a fixed limit price $\tilde{r}_t + 1$ for the sell order.

Tuned Spoofer. In this strategy, the sell price is $\tilde{r}_t + o$, where the offset is found through a grid search of values $o \in [1, 200]$ in multiples of 10. Each offset performance is evaluated as the average spoofer surplus over 4×10^4 simulations in the experimental test environment. The best performing offset for each liquidity environment is then chosen as the optimal.

R-Learned Spoofer. This spoofer uses an RL-derived policy (trained over 2.5×10^3 simulations) to decide the optimal offset $o \in [1, 200]$ from the estimated fundamental $\tilde{r}_t$ given current market observations. The underlying RL algorithm utilized was Proximal Policy Optimization (PPO) [18] with default parameters[1] as specified in Stable-Baseline3 [16]. The observation space for the spoofer consists of:

- Time left in simulation: $(T - t)$
- Best bid at current timestep: BID_t
- Best ask at current timestep: ASK_t
- Estimated final fundamental at time t: $\tilde{r}_t$
- Midprice movement from $t - 1$ to t: Indication of direction of market price movement
- Volume Imbalance: The ratio of the difference in quantity of the buy and sell sides of the order book
- Queue Imbalance: The ratio of the difference between the number of buy and sell orders
- Volatility: Variation in asset market value determined by historical returns
- Relative Strength Index (RSI): A technical indicator used in momentum trading that measures the speed of a security's recent price changes to assess whether the security is overvalued or undervalued

As the spoofer's surplus is only realized at the end of the simulation, we utilize a period-by-period reward function that calculates rewards as changes in valuation between two consecutive spoofer orders at timesteps t and $t + k$. In essence, this reward represents the incremental impact (in hindsight) on final spoofer surplus of the action taken at time $t + k$:

$$u_{t+k} = (r_T \cdot q_{t+k} + cash_{t+k}) - (r_T \cdot q_t + cash_t).$$

3.3 Realized Surplus

At the end of each simulation, the surplus for each background agent i is computed as the sum of their net cash ($cash_i$) and the valuation $final\_val_i$ of their final net position q_i^T:

$$surplus_i = final\_val_i + cash_i,$$

where the final valuation is determined by the final fundamental, r_T, and the agent-specific cumulative private value of the position:

$$final\_val_i = r_T \cdot q_i^T + \begin{cases} \sum_{k=1}^{k=q_i^T} \theta_i^k & \text{if } q_i^T > 0, \\ - \sum_{k=q_i^T+1}^{k=0} \theta_i^k & \text{if } q_i^T < 0. \end{cases}$$

Since spoofers and MMs do not have preferences for holding a certain position on the security, they do not have private values. Therefore, their surplus is calculated as $(r_T \cdot q_i^T + cash_i)$.

4 Environment Settings

4.1 Simulation Setup

In our experiments, we use PyMarketSim, an efficient Python implementation [14] of the agent-based market simulator described in Section 3.[2] Each simulation runs for $T = 1 \times 10^4$ timesteps and includes 26 agents: 24 background traders (12 ZI and 12 HBL), 1 MM, and 1 spoofer.

The following market parameters remain constant across all simulations: the fundamental mean $\bar{r} = 1 \times 10^5$, the mean-reversion parameter $\kappa = 5 \times 10^{-2}$, and the fundamental shock u_t at each timestep is drawn independently from distribution $\mathcal{N}(0, 1 \times 10^4)$.

The Poisson arrival rate for background traders is $\lambda_b = 0.002$, that is, once every 500 timesteps on average, resulting in *some* background agent arriving roughly every 20 timesteps.

Background traders have a maximum position of $q_{\max} = 10$, surplus offset bounds of $[250, 500]$, and private values are drawn from a zero-mean normal distribution with variance $\sigma_{PV}^2 = 5 \times 10^6$. The HBL memory window is $\mathcal{L} = 4$. The MM enters the market at a rate of $\lambda_{MM} = 0.035$, corresponding to an arrival approximately every 29 timesteps. The spoofer has an arrival rate of $\lambda_{SP} = 0.02$, which translates to arrivals approximately every 50 timesteps. The spoofer arrives in the market only after timestep $t = 1000$ to allow for market warm-up. To account for the stochastic nature of the simulations, including fluctuations in market fundamentals, variations in agent arrival rates, and diverse private valuations, we average the results over 40,000 simulations.

4.2 Market Liquidity Configurations

According to Chordia et al. [6], *high market liquidity* refers to "the ability to buy or sell large quantities of an asset quickly and at low cost". In our setting, we define liquidity in terms of parameters

[1] Hyperparameter tuning could potentially improve the performance of the RL-derived spoofer, but we kept the default.

[2] The code for our experiments is publicly available through the PyMarketSim repository.

	A1	**A2**	**A3**	**B1**	**B2**	**B3**	**C1**	**C2**	**C3**
ξ	10	50	100	10	50	100	10	50	100
ω	16	16	16	64	64	64	256	256	256
K	8	8	8	8	8	8	8	8	8

Table 1: Market maker configurations for each environment.

governing our liquidity provider: the MM. Specifically, we control liquidity by setting MM rung size ξ and spread ω. We consider nine distinct configurations as shown in Table 1.

- **Low ξ, Low ω**: Highest liquidity (e.g., A1). Price ladders are densely concentrated near $\tilde{r}_t$, creating a thick market around the fundamental, enabling larger volumes to be traded with minimal price movement.
- **High ξ, Low ω**: High/mid liquidity (e.g., A3). Price ladders start close to $\tilde{r}_t$ but have large gaps between the rungs. Some of the MM orders offer attractive prices, but the sparse rungs increase the probability of large market price movements.
- **Low ξ, High ω**: Mid/low liquidity (e.g., C1). Price ladders are dense but placed further from $\tilde{r}_t$. Depending on how large ω is, the ladder may begin so far from $\tilde{r}_t$ that even dense rungs have little effect in increasing market liquidity.
- **High ξ, High ω**: Lowest liquidity (e.g., C3). Price ladders start far from $\tilde{r}_t$ and have large gaps between rungs, reducing trading activity and increasing the market impact of large orders.

Note that these scenarios outline key parameter combinations; the rest of the simulated market configurations denote incremental changes within this spectrum.

By fixing one of the two MM parameters, we can easily compare the liquidities of the markets. For example, A1 is more liquid than B1, which is more liquid than C1. However, when altering both parameters, such as in B3 and C1, the relative liquidities can be less obvious.

To characterize the joint effect of ξ and ω, we plot the *bid-ask spread* ($ASK_t - BID_t$). In Figure 1, we see that in high-liquidity markets, the bid-ask spread is smaller compared to markets with low liquidity (e.g., compare A1, B1, and C1 or A1, A2, and A3).

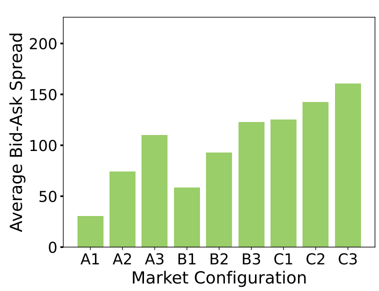


Figure 1: Average bid-ask spread in each market configuration (with no spoofer). Error bars are omitted, as they are significantly smaller than marker size.

5 Experimental Results

We quantify the performance and impact of our spoofing strategies along four dimensions: the spoofer's realized profit (i.e., surplus), and the effects of spoofing on the ZI agents' surplus, the HBL agents' surplus, and the market midprice.

5.1 Spoofer's Surplus

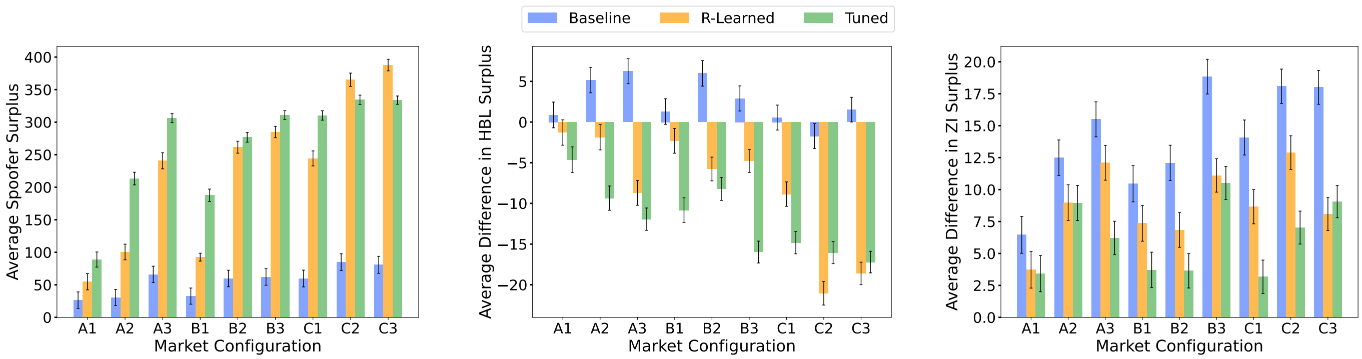
In Figure 2a, we compare our three different spoofing strategies with respect to spoofer surplus. We observe that all strategies achieve positive surplus in all environments. However, the baseline spoofer achieved significantly lower surplus compared to the other strategies, especially in low liquidity markets. This underperformance is attributed to the low-priced sell order submitted by the spoofer, placed at a price of only one tick above $\tilde{r}_t$, resulting in extremely low profit per transaction. In contrast, both the R-Learned and tuned spoofers performed significantly better as their sell limit prices were strictly higher than one tick above $\tilde{r}_t$ for all environments and thus, were either learned or optimized to be more effective at generating profits. We delve deeper into the importance of the sell price in Section 6.

5.2 Spoofing Effects on HBL Agents

In Figure 2b, we illustrate the impact of each spoofer on the average surplus of HBL agents across market configurations. Each vertical bar represents the difference between the average realized surplus of HBL agents in an environment with a spoofer (of one of the three types indicated in the legend) relative to that achieved in an otherwise identical environment with no spoofer; individual surpluses are computed as described in Section 3.3. We observed that HBL agents generally experienced losses in the presence of spoofers except in markets with the baseline spoofer. The baseline spoofer places extremely low-priced sell orders, allowing HBL agents to accumulate inventory at a lower cost. Although the manipulation occasionally leads HBL agents to trade at higher prices and earn lower profits, the gains from transactions with the spoofer's low-priced sell orders still outweigh these losses, resulting in a relatively strong net positive effect on HBL agent profits. However, in low liquidity markets (e.g., the C markets), the spoofing effect is more powerful and pushes HBL agents to buy at even higher prices, neutralizing the high profits from the baseline spoofer's low-priced sell orders. Moreover, a more powerful spoofer does not provide low sell orders (see the tuned offsets in Table 2) and significantly manipulates the HBL agents' beliefs, resulting in substantially lower surplus for the HBL agents relative to achieved surplus in equivalent non-spoofed environments.

5.3 Spoofing Effects on ZI agents

In Figure 2c, we demonstrate the effects of spoofing strategies on the average surplus of ZI agents. Each vertical bar represents the difference between the average realized surplus of ZI agents in an environment with a spoofer (of one of the three types indicated in the legend) relative to that achieved in an otherwise identical environment with no spoofer. We observed that the spoofer increased average ZI agent surplus across all market settings. This was primarily due to the spoofer's manipulation causing HBL agents to place higher-priced buy orders, which in turn increased the likelihood



(a) Spoofer surplus

(b) Impact of spoofing on HBL agent surplus

(c) Impact of spoofing on ZI agent surplus

Figure 2: Spoofer surplus and effect of spoofing on background traders. For each background trader type, bars represent average surplus difference between spoofed and non-spoofed markets that otherwise have identical experimental parameters. Error bars represent standard errors.

of high-priced ZI agent sell orders transacting. Additionally, the baseline spoofer strategy yielded the highest increase in ZI agent surplus among the three strategies. This was due to the baseline spoofer's low-priced sell orders, which enabled ZI agents to make high profit transactions, similar to the baseline strategy's effect on HBL agents. As sell prices were set to be higher in the R-Learned and tuned spoofing strategies, the increase in surplus for ZI agents was significantly lower under those strategies. Furthermore, in less liquid markets, spoofing orders had a more pronounced effect on HBL agents, inducing higher bid prices (e.g., compare midprices of C3 and A1 markets in Figures 3b and 3c). This further increased the probability of higher-priced ZI agent sell orders executing, resulting in higher profit transactions and greater overall surplus relative to more liquid markets.

5.4 Spoofing Effects on Market Price

In our model, the spoofer aims to profit by creating the perception of high demand in the market, which is represented by an upward trend in market price, potentially followed by a dip if and when the spoofer's sell order is executed. With this in mind, we plot the volatility in the market midprice (i.e., midpoint between BID_t and ASK_t) under each spoofing strategy in Figure 3. As evident from Figure 3a, the baseline spoofer had the least manipulative impact. In all environments with a baseline spoofer, midprices barely fluctuate and, in particular, do not have the strong initial upward trends exhibited in the presence of the two more powerful spoofing strategies, as seen in Figures 3b and 3c. In fact, Figure 3a shows a sharp initial decrease in the midprice from the fundamental mean in the presence of the baseline spoofer. This is because the spoofer's sell order and spoofing buy order jointly influence the beliefs of HBL agents. We elaborate upon this mechanism in Section 8.1.

Furthermore, we found that liquidity levels significantly impact spoofing-induced market midprice movements. For instance, Figures 3b and 3c show insignificant upward trends in midprice in highly liquid markets (e.g., A1) compared to a substantial increase in less liquid markets (e.g., C3). These upward trends are then followed by declines due to the spoofers capitalizing on the artificial price increase and pushing the market price back down. A detailed analysis of the mechanisms behind the effects of market liquidity on spoofing is provided in Section 8.

6 Liquidity-Induced Spoofing Behavior Regimes

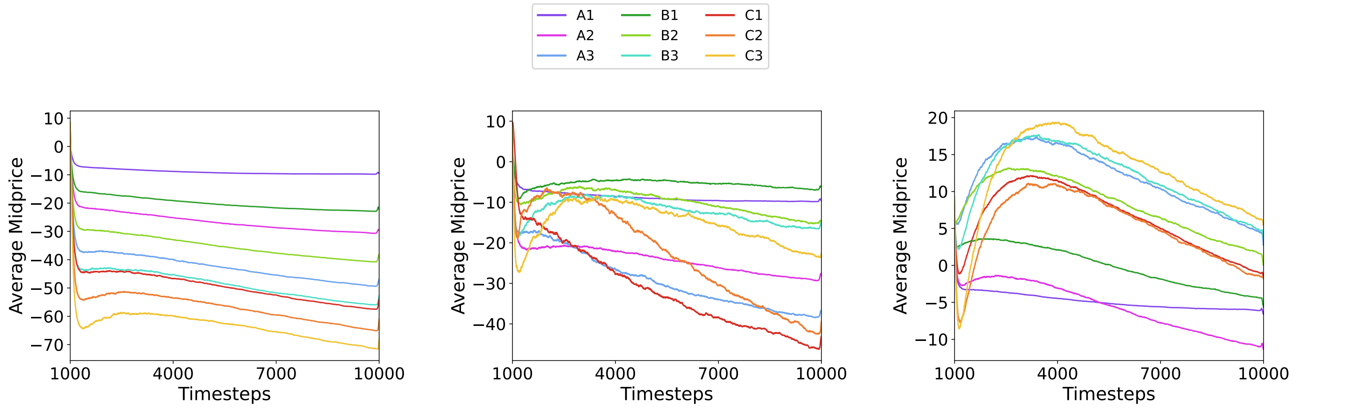
Experimental observations indicate that market liquidity influences a spoofer's ability to manipulate the market. Consequently, spoofers adjust their strategic approach based on market liquidity. Our study identified two distinct spoofing regimes in markets with varying liquidity levels. In highly liquid markets, spoofers placed sell orders that yielded lower profit-per-transaction but achieved a higher transaction rate. Conversely, in low liquidity markets, spoofers aimed for higher profit per sale with lower transaction rates.

	A1	A2	A3	B1	B2	B3	C1	C2	C3
Optimal Offset	10	40	100	40	80	110	100	110	140

Table 2: Experimentally-derived optimal sell offset o for the tuned spoofer in each market configuration. See Section 3.2.3 for details.

6.1 Low-Profit High-Frequency Regime

In high liquidity markets, a spoofer's ability to significantly influence market prices is limited. Consequently, their optimal strategy shifts towards prioritizing high transaction volume at the expense of profit-per-transaction. This behavior can be seen by examining the optimal sell offsets in Table 2. In the most liquid market, A1, the optimal tuned sell price for the spoofer was found to be 10 ticks above $\tilde{r}_t$, much lower than the offsets in less liquid markets (e.g., 140 ticks for C3). Consequently, the spoofer's sell orders are transacted more frequently, resulting in a lower (i.e., shorter) final position at the end of the simulation, as shown in Figure 4. We refer to this behavior as *low-profit, high-frequency trading*.



(a) Baseline Spoofer (b) R-Learned Spoofer (c) Tuned Spoofer

Figure 3: Average midprice evolution post-spoofer entry at $t = 1000$. Plotted values are an offset from the fundamental mean, 1×10^5.

Notably, the baseline spoofer is an extreme implementation of this behavior. Each unit-sell order is placed so as to achieve minimal profit but with an extremely high probability of execution. While it generates significantly lower surplus in non-liquid environments compared to the R-Learned and tuned spoofing strategies, the baseline spoofer achieves decent surplus in high liquidity markets relative to the other spoofing strategies because it utilizes the same low-profit high-frequency trading regime. In addition, in high liquidity markets, the spoofer's behavior (i.e., placing low sell limit prices) is less harmful to background traders as these low sell prices help counteract the effects of the spoofing orders. Therefore, high liquidity in a market is considered beneficial for both deterring spoofing and creating more stability in the market.

6.2 High-Profit Low-Frequency Regime

In contrast, in lower liquidity environments, the effect of spoofing orders is more significant in manipulating the market midprice. Therefore, the spoofer prefers to place sells at higher price levels, as shown in Table 2, because the sells maintain a reasonable probability of transaction. Although the transactions are less frequent with this behavior, the profit-per-transaction is much higher. We refer to this behavior as *high-profit, low-frequency trading*. When the market allows spoofers to exhibit this high-profit, low-frequency behavior, the midprice is manipulated to a much higher degree; in turn, learning traders (i.e., HBL agents) in the market suffer significant losses to their surplus. This can be seen through the R-Learned and tuned spoofer's negative effects on HBL surplus in the low liquidity markets shown in Figure 2b.

As illustrated in Figure 4, we observed a substantial difference in final positions for the tuned spoofer in high liquidity markets (A1, A2, and B1) and low liquidity environments because of the different spoofing regimes. In high liquidity markets, spoofers adopted a low-profit, high-frequency approach, selling significantly more units at lower prices. In contrast, spoofers in lower liquidity markets utilized a high-profit, low-frequency approach. This pattern and substantial difference in units sold ($\approx$ 1.5 units as shown in Figure 4) and tuned limit prices ($\geq$ 40 as detailed in Table 2) between markets above and below B1's liquidity level suggests the existence of a critical liquidity threshold. When market liquidity reaches this threshold, it seems to prompt spoofers to switch between these two behaviors. This observation suggests that maintaining adequate market liquidity could serve as an inherent defense mechanism against spoofing, potentially informing financial regulation approaches.

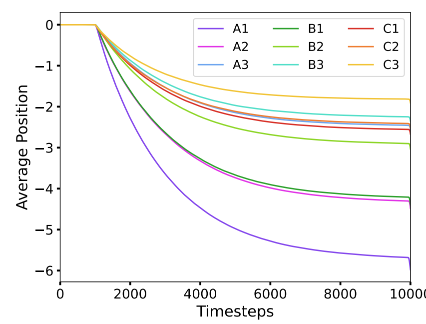
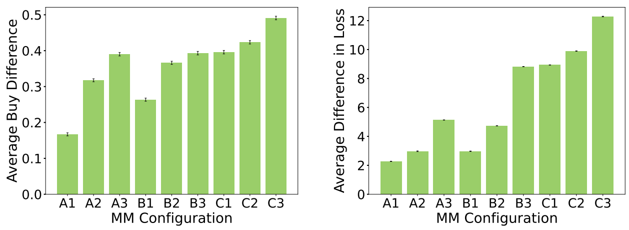


Figure 4: Average position of the tuned spoofer over the course of the simulation in each market configuration.

7 The Effects of Arrival Rate on Spoofing

A spoofer's rate of arrival to the market affects the frequency of modifying its previous sell and spoofing buy orders. A low arrival rate can increase the risk of unintended execution of spoofing orders and reduce the ability to adjust to current market conditions promptly, whereas a high arrival rate can help mitigate these risks. To understand the impact of the arrival rate on spoofing, we experimentally compare the performance of spoofing under two selected arrival rates $\lambda_{SP} = 0.02$ (fast) and $\lambda_{SP} = 0.006$ (slow).

In Figure 5, we illustrate the difference in the volume of buy spoofing orders transacted by the slow and fast tuned spoofers, along with the corresponding differences in surplus loss due to the buy spoofing orders transacting. From Figure 5a, as expected, we observed that, in all markets, the slower spoofer had more of



(a) Difference in spoofing order transaction volume.

(b) Difference in magnitude of surplus loss due to transacted volume of spoofing orders.

Figure 5: Average difference between slow ($\lambda_{SP} = 0.006$) spoofer and fast ($\lambda_{SP} = 0.02$) spoofer. Error bars represent standard errors.

	Δ Units Sold	% Surplus Increase	Δ Offset
A1	+1.47	+34.5	0
A2	+0.69	+26.5	+10
A3	+0.30	+20.6	+20
B1	+0.89	+32.7	0
B2	+0.58	+21.1	+10
B3	+0.54	+16.4	+10
C1	+0.58	+14.2	+10
C2	+0.58	+18.7	+10
C3	+0.53	+12.6	+10

Table 3: Increase in units sold, percent increase in surplus, and change in optimal offset values from the slow ($\lambda_{SP} = 0.006$) to the fast ($\lambda_{SP} = 0.02$) tuned spoofer.

its spoofing orders transact, with a higher number of units being transacted in less liquid markets. In high liquidity markets, the MM frequently places orders near the fundamental, often establishing the best bid. Even if these orders execute, they are quickly replaced, shielding spoofing orders and reducing their transaction probability. Conversely, in low liquidity markets, MM bids typically fall below the best bid due to larger MM order spreads/offsets and the spoofer's more effective inflationary effect on the buy side. Thus, with slower order replacement of the best bid by background agents, spoofing orders lack the same protection, and thus, face higher transaction probabilities. Accordingly, we noticed that the negative effect on realized surplus increased proportionately to the transaction volume of the spoofing orders, as illustrated in Figure 5b.

In Table 3, we present the differences between the fast and slow spoofers in terms of average units sold, surplus, and tuned optimal offsets. As expected, we observed that the fast spoofer sold more units across all environments compared to the slow spoofer, resulting in a significant increase in surplus achieved. These differences are especially pronounced in highly liquid markets (A1, A2, and B1) because optimal spoofing behavior (i.e., low-profit, high-frequency) in these conditions depend on executing a high volume of transactions. As market liquidity decreases, the gap in units sold and the surplus difference diminish, suggesting that the advantage of arriving quickly is less pronounced in less liquid markets. This is because the optimal manipulation behavior (i.e., high-profit, low-frequency) relies on generating surplus through high-profit-per-transaction as opposed to transaction rate. Finally, we observed that the optimal offsets for the fast spoofer are higher than, or occasionally equal to, those of the slow spoofer. This is because a spoofer that arrives more frequently can manipulate the market more effectively, and thus can set higher offsets while maintaining a viable probability of execution for its sell orders.

8 Mitigating Spoofing with Liquidity

In addition to our experimental results quantifying how liquidity mitigates spoofing, in this section, we identify the two key mechanisms through which market maker orders reduce the effects of spoofing on the market.

8.1 HBL Belief Stabilization

First, we describe the mechanism through which spoofing induces HBL agents to place higher-priced bids. When HBL agents arrive, they interpret unexecuted buy orders as rejected orders. Consequently, the large quantity buy spoofing order with price $p_{SP} = BID_t - 1$ significantly inflates the *RBG* term in the agents' belief function $f_t(p)$ (i.e., Equation (2)) for bids below p_{SP}. This inflation diminishes $f_t(p)$ towards zero for bid prices $p < p_{SP}$, meaning that these lower limit prices are unlikely to be optimal for maximizing expected surplus. Thus, HBL agents place bids at higher limit prices and see a corresponding reduction in their surplus.

In high liquidity markets, the array of MM sell orders near $\tilde{r}_t$ can counteract the inflationary effect of the spoofing order. Specifically, the low-priced MM asks increase the *AL* term for buying in Equation (2) for prices - *even slightly* - higher than the MM's sell orders. This balances the relative scale of *AL* and *RBG* for those bid prices, increasing HBL beliefs for lower bid prices and thereby reducing the inflationary effect of *RBG*. Furthermore, the MM's sell orders have a symmetric effect to the spoofing buy orders. When these sell orders remain unexecuted, they contribute to the *RAL* term for sell prices above the spoofer's sell order price in Equation (2). This induces HBL agents to place their sell orders at prices lower than the MM's sell orders, creating deflationary pressure on the market price, counteracting the effect of the spoofing order.

Notably, the baseline spoofer's low-priced sell order has the same effect as the MM's sell orders. However, because the sell price is extremely low in the baseline spoofer strategy, the deflationary effect is much stronger, and thus, we observe a decrease in market midprice as shown in Figure 3a.

8.2 MM Order Competition

In high liquidity markets, another way a MM can counteract manipulation is by engaging in direct price competition with the spoofer's sell order. Such MM orders are, by definition, densely packed around the fundamental. Hence, these orders often occupy the top of the order book, particularly on the sell side.

As a result, even if the market price rises, the MM's sell orders have priority and will execute first with the higher-priced bids, *undercutting* the spoofer's sell order. Consequently, for the spoofer to profit from its manipulation, it must place sell orders at lower prices. This negatively impacts the spoofer in two ways: (1) the spoofer will earn less profit from the manipulation, and (2) the lower-priced sell order will counteract the inflationary pressure created by the spoofing order, as explained above in Section 8.1.

9 Conclusion

This study employs a MM to maintain consistent market liquidity, enabling an investigation into liquidity's role as a natural deterrent to spoofing. Our key finding is that high liquidity markets significantly reduce the effectiveness of spoofing across all tested manipulation strategies. We evaluate a spoofer's success using three metrics: achieved surplus, impact on market price, and impact on background traders, particularly the adverse effect on HBL agents. While the baseline spoofing strategy showed minimal success, our novel R-Learned and tuned strategies caused substantial losses to HBL agents and yielded significantly higher profits. Furthermore, we identified the existence of distinct, optimal spoofing regimes in high and low liquidity markets respectively. We also investigated the interplay between liquidity and the role of the spoofer's arrival frequency on its profitability. Finally, supported by these observations, we provide a detailed analysis of the mechanisms through which market liquidity counteracts spoofing.

References

[1] Aurelio F Bariviera, M Belén Guercio, Lisana B Martinez, and Osvaldo A Rosso. 2016. Libor at crossroads: Stochastic switching detection using information theory quantifiers. *Chaos, Solitons & Fractals* 88 (2016), 172–182.

[2] David Byrd. 2022. Learning not to spoof. In *3rd ACM International Conference on AI in Finance (ICAIF)*. 139–147.

[3] Álvaro Cartea, Patrick Chang, and Gabriel García-Arenas. 2023. Spoofing Order Books with Learning Algorithms. *Available at SSRN, 4639959* (2023).

[4] Álvaro Cartea, Sebastian Jaimungal, and Yixuan Wang. 2020. Spoofing and price manipulation in order-driven markets. *Applied Mathematical Finance* 27, 1-2 (2020), 67–98.

[5] Tanmoy Chakraborty and Michael Kearns. 2011. Market making and mean reversion. In *12th ACM conference on Electronic Commerce (EC)*. 307–314.

[6] Tarun Chordia, Asani Sarkar, and Avanidhar Subrahmanyam. 2005. An empirical analysis of stock and bond market liquidity. *The Review of Financial Studies* 18, 1 (2005), 85–129.

[7] Darrell Duffie and Piotr Dworczak. 2021. Robust benchmark design. *Journal of Financial Economics* 142, 2 (2021), 775–802.

[8] Alexander Eisl, Rainer Jankowitsch, and Marti G Subrahmanyam. 2017. The manipulation potential of Libor and Euribor. *European Financial Management* 23, 4 (2017), 604–647.

[9] Steven Gjerstad and John Dickhaut. 1998. Price formation in double auctions. *Games and Economic Behavior* 22, 1 (1998), 1–29.

[10] Dhananjay K Gode and Shyam Sunder. 1993. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy* 101, 1 (1993), 119–137.

[11] John M. Griffin and Amin Shams. 2018. Manipulation in the VIX? *Review of Financial Studies* 31, 4 (2018), 1377–1417.

[12] Eun Jung Lee, Kyong Shik Eom, and Kyung Suh Park. 2013. Microstructure-based manipulation: Strategic behavior and performance of spoofing traders. *Journal of Financial Markets* 16, 2 (2013), 227–252.

[13] Enrique Martínez-Miranda, Peter McBurney, and Matthew JW Howard. 2016. Learning unfair trading: A market manipulation analysis from the reinforcement learning perspective. In *IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*. IEEE, 103–109.

[14] Chris Mascioli, Anri Gu, Yongzhao Wang, Mithun Chakraborty, and Michael P. Wellman. 2024. A Financial Market Simulation Environment for Trading Agents Using Deep Reinforcement Learning. In *5th ACM International Conference on AI in Finance*.

[15] Takanobu Mizuta. 2020. Can an AI perform market manipulation at its own discretion? – A genetic algorithm learns in an artificial market simulation –. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. 407–412.

[16] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research* 22, 268 (2021), 1–8. http://jmlr.org/papers/v22/20-1364.html

[17] Bernhard Rauch, Max Goettsche, and Florian El Mouaaouy. 2013. LIBOR Manipulation: Empirical Analysis of Financial Market Benchmarks Using Benford's Law. *Available at SSRN, 2363895* (2013).

[18] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[19] Megan Shearer, Gabriel Rauterberg, and Michael P. Wellman. 2023. Learning to manipulate a financial benchmark. In *4th ACM International Conference on AI in Finance (ICAIF)*. 592–600.

[20] Alexis Stenfors, Mehrdaad Doraghi, Cristina Soviany, Masayuki Susai, and Kaveh Vakili. 2023. Cross-market spoofing. *Journal of International Financial Markets, Institutions and Money* 83 (2023), 101735.

[21] Xuan Tao, Andrew Day, Lan Ling, and Samuel Drapeau. 2022. On detecting spoofing strategies in high-frequency trading. *Quantitative Finance* 22, 8 (2022), 1405–1425.

[22] Andy Verity and Eleanor Lawrie. 2020. Hound of Hounslow: Who is Navinder Sarao, the 'flash crash trader'? — bbc.com. https://www.bbc.com/news/explainers-51265169.

[23] Andrew Verstein. 2015. Benchmark manipulation. *BCL Rev.* 56 (2015), 215.

[24] Elaine Wah, Mason Wright, and Michael P. Wellman. 2017. Welfare effects of market making in continuous double auctions. *Journal of Artificial Intelligence Research* 59 (2017), 613–650.

[25] Xintong Wang, Christopher Hoang, Yevgeniy Vorobeychik, and Michael P. Wellman. 2021. Spoofing the limit order book: A strategic agent-based analysis. *Games* 12, 2 (2021), 46.

[26] Xintong Wang and Michael P. Wellman. 2020. Market manipulation: An adversarial learning framework for detection and evasion. In *29th International Joint Conference on Artificial Intelligence (IJCAI)*. 4626–4632.

[27] Yun-Yi Wang. 2019. Strategic spoofing order trading by different types of investors in Taiwan Index futures market. *Journal of Financial Studies* 27, 1 (2019), 65–103.

[28] Carter Yagemann, Pak Ho Chung, Erkam Uzun, Sai Ragam, Brendan Saltaformaggio, and Wenke Lee. 2021. Modeling large-scale manipulation in open stock markets. *IEEE Security & Privacy* 19, 6 (2021), 58–65.