Market Making with Scaled Beta Policies

Joseph Jerome
Department of Computer Science
Liverpool, UK
j.jerome@liverpool.ac.uk

Gregory Palmer
L3S Research Center
Hannover, Germany
gpalmer@l3s.de

Rahul Savani
Department of Computer Science
Liverpool, UK
rahul.savani@liverpool.ac.uk

ABSTRACT

This paper introduces a new representation for the actions of a market maker in an order-driven market. This representation uses scaled beta distributions, and generalises three approaches taken in the artificial intelligence for market making literature: single price-level selection, ladder strategies, and "market making at the touch". Ladder strategies place uniform volume across an interval of contiguous prices. Scaled beta distribution based policies generalise these, allowing volume to be skewed across the price interval. We demonstrate that this flexibility is useful for inventory management, one of the key challenges faced by a market maker.

We conduct three main experiments: first, we compare our more flexible beta-based actions with the special case of ladder strategies; then, we investigate the performance of simple fixed distributions; and finally, we devise and evaluate a simple and intuitive dynamic control policy that adjusts actions in a continuous manner depending on the signed inventory that the market maker has acquired. All empirical evaluations use a high-fidelity limit order book simulator based on historical data with 50 levels on each side.

KEYWORDS

limit order book, market making, liquidity provision, inventory risk

1 INTRODUCTION

This paper considers the problem of a market maker acting in an order-driven market. In such markets, matched orders result in trades and unmatched orders are stored in a *limit order book*, which is split into two parts, a collection of buy orders called *bids*, and a collection of sell orders called *asks*. A market maker provides liquidity by continuously having both bids and asks in the book, thereby allowing others to trade in either direction whenever needed. The difference between the market maker's best ask and best bid is called their *quoted spread*, which could be wider and use different prices than the spread of the whole limit order book market. The

goal of a market maker is to repeatedly earn this spread by transacting in both directions. The challenge for the market maker is to mitigate the inventory risk that comes from trading with better-informed traders. That is, market makers expose themselves to *adverse selection*, a phenomenon where the market maker's counterparties exploit a technological or informational advantage when transacting with them. In particular, this causes the market maker to amass a (positive or negative) inventory, before an adverse price move causes the market maker to incur a loss on this inventory (for example, where the market maker has net sold to the market just before a significant price rise).

Market making has become increasingly automated and the frequency of trading and corresponding data requirements has grown significantly [20, 21, 26, 32]. This paper investigates a novel but natural way to represent the actions of an automated market maker. Our approach uses scaled beta distributions as a flexible and succinct way to define the volume profiles of bids and asks that a market maker places. We demonstrate the utility of this representation by backtesting and analysing market making agents that use this representation within high-fidelity simulations of the limit order book (using LOBSTER[1] data with 50 levels of limit orders on each side of the book).
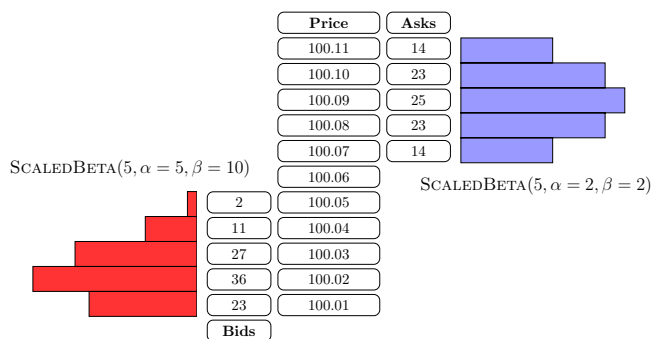

Figure 1: An example of a limit order book, with an illustration of order distributions described by beta distributions.

Figure 1 gives an illustration of how we use beta distributions to describe volume profiles for orders across price levels. For the sake of illustration, assume that all shown orders come from the same market marker. The bid and ask volumes in this example are derived from SCALEDBETA$(5, 5, 10)$ and SCALEDBETA$(5, 2, 2)$, where the first argument, 5, is the number of levels to quote at, and the second and third parameters are the shape parameters of a standard beta distribution (α and β respectively; see Section 2.3).

---

[1] https://lobsterdata.com/

1.1 Related work

Market making has been investigated within the economics, finance, and artificial intelligence (AI) literatures. The classic approach taken in the mathematical finance literature has been to treat market making as a problem of *stochastic optimal control*, where models for order arrivals and executions are developed and then control problems are designed and solved for them [1, 3, 8, 10, 17–19, 22, 23]. Because of the emphasis on analytically proving results about optimal or approximately optimal control policies, the action space of the market maker is typically quite restrictive. This often happens to the point where the market maker only controls a single order on each side of the market and therefore a single spread. The main novelty of this paper is a flexible parametric representation of order profiles, which permits order placement across many prices.

We give a brief summary of the AI for market making (AI4MM) literature. Since the main focus of this paper is the novel policy, we predominantly contrast this with the policies used in other parts of the AI4MM literature, where, broadly speaking, the action spaces can be divided into three categories: that of choosing discrete half spreads at which to place buy and sell orders from a finite set (whilst possibly also managing the amount of volume placed at both levels); that of choosing a range for a "ladder strategy"; and that of "market-making at-the-touch" where the actions consist in choosing either to place an order or not at each of the best bid and best ask. We discuss these three approaches in turn.

1.1.1 Single price-level policy. A natural two-dimensional action space is for the agent to select two *half-spreads*, i.e. a bid and ask offset from the midprice. Typically, in the literature, with this setup all orders are assumed to be of constant volume.[2] The agent then adapts these half-spreads at each time step according to the state of the market and the agent's inventory. This approach of choosing half spreads is the main one taken in the financial stochastic control literature on market making (e.g., see Avellaneda and Stoikov [3], Gueant et al. [18] or Cartea, Jaimungal and co-authors [6, 9, 10]).

While the financial stochastic control literature tends to use continuous models with corresponding continuous half-spreads, by and large, the AI4MM literature restricts to the case where market makers have discrete action spaces. In particular, the problem is that of choosing the *number of ticks* away from the touch (the best bid and ask prices) at which to quote the bid and the ask. It is worth noting that due to the action space being the product space of bid and ask actions, this can get very large unless the agent is restricted to place actions very close to the best prices.

Reducing the action space by taking differences. The first application of reinforcement learning to market making (Chan and Shelton [12]) used such a policy. However, to get around the issue of the large action space, they instead chose, from a small set of actions, how much they would increase or decrease their quotes. They then used reinforcement learning to optimise an agent's interaction with a mathematical model (similar to that proposed by Glosten and Milgrom [16]) for the market dynamics. Subsequently, Kim and Shelton [24] fitted an input-output hidden Markov model to order data from Nasdaq and use reinforcement learning to learn how

to act in the model. This was a significant improvement in terms of realism. Still, due to only observing a fraction of the order flow volume they still needed to use a model of the market. They allowed their agent to increase, decrease, or keep their bid, ask, and both associated volumes by at most one (tick or unit of the asset). Whilst this seems rather restrictive (as it requires many steps to make a dramatic change), the action space is already ($3^4 = 81$)-dimensional.

Choosing an action from a prespecified subset of available actions. More recently, Spooner et al. [34] considered a much more realistic market simulator, using 5 levels of order book data, along with transactions. This was the first paper to train reinforcement learning agents on the vast quantities of so-called *Level II* data now available. However, there is still a slight partial observability problem: when the order book changes, and a transaction doesn't occur, it is not possible to know from where in the queue this cancellation/deletion came from. In particular, when interacting with the market, it is necessary to assume a distribution of such cancellations. The authors of [34] chose a uniform distribution.

The action space of Spooner et al. [34] is an octuple of prespecified half spread pairs, along with an action which clears the entirety of the agent's inventory using a market order. It is worth noting that some of these actions are skewed to favour filling on one side. This approach permits a basic form of the inventory control discussed in Section 3.6, whilst keeping the action space of manageable size. This paper was the first to use such a finite pre-specified selection of actions and it has since been utilised by Xu et al. [36] and Sadighian [31]. Whilst Spooner et al. [34] used SARSA(λ) and a state-space discretisation, [36] used a variant of a deep Q networks and Sadighian [31] used proximal policy optimisation [33]. At a similar time to Spooner et al. [34], Lim and Gorse [27] considered a stochastic model driven by Poisson processes of the form proposed by Cont et al [13], and allowed an agent to quote a single bid and ask of fixed volume a number of ticks away from the best bid, here chosen from the set $\mathcal{A} = \{1, 2, 3\}^2$. A similar policy is adopted by Patel [29]. However, to reduce the dimensionality of the action space, only one half spread is chosen (on either side) and a limit order is placed at the best price on the other side of the book.

A continuous action space. Spooner and Savani [35] considered a robust version of the Avellaneda and Stoikov model [3] in which an adversarial "market" agent controls the drift of the financial market. This problem ends up being equivalent to the problem considered by Nyström et al. [28]. Here, the action space is given by four continuous parameters controlling the mean and variance of the agent's bids and asks. It is learnt by approximating the value function using cubic polynomials and then performing least squares policy iteration [25].

A final form of action space which falls into the *single price-level* category is that of choosing a continuous half spread on each side of the book and then quantising to submit orders that are on the price grid. This approach is taken by Gašperov and Kostanjčar [15] in which they use neuroevolution to train a policy given by a deep neural network. They use historical Bitcoin trade data with only the first level of the order book.

It is worth noting that such policies based on half-spreads are a subclass of the scaled beta policy that we introduce here (they can be recovered by letting the variance decrease to zero). However,

---

[2]Alternatively, one could select a volume at each of these levels, making the action space of the control problem four-dimensional.

the authors feel that choosing half-spreads is not so sensible for the following reason: when actually implementing such a strategy, to change the half spread on each side of the book, it is necessary to actively cancel orders at each time step and place new orders at the new level. Furthermore, even if the agent doesn't change their spread it is necessary to cancel and place new orders to maintain a fixed spread (when the midprice moves). This causes the agent to perpetually lose queue position on price-time priority exchanges when they place new orders (at the back of the queue). In particular, very few of their orders actually get filled. In contrast, if an agent places orders according to a ScaledBeta distribution, their orders get updated in a smooth manner by constantly adding or removing smaller orders to the book (see Figure 3 for an illustration).

1.1.2 Ladder strategies. Another relevant strand of the existing literature was started by Chakraborty and Kearns [11]. In this paper, the authors introduced and studied the use of *ladder strategies*, which place a unit of volume at all prices in two price intervals, one on each side of the book. [11] theoretically proved the utility of these strategies in mean-reverting markets with Ornstein-Uhlenbeck price dynamics. Inspired by [11], Abernethy and Kale [2] considered related order placement strategies where limit orders for one unit of volume are placed at all price levels (right down to the lowest possible price and up to some predefined highest price) outside of a window around the midprice that defines the market maker's spread. [2] presented an online learning scheme that mixes between parametrisations of their ladder strategies[3] and in doing so provably guarantees to perform competitively with the single best parameter choice in hindsight. They do an empirical evaluation with real data, but only using the price time series of trades, rather than actually modelling the limit order book process as we do here. This requires one to make assumptions about fill-rates, agent queue positions and many other aspects of the market's microstructure. That being said, such a strategy deals with the problem mentioned at the end of Section 1.1.1 of losing queue position, and we use it as a benchmark throughout the paper. Both [11] and [2] are primarily about the theoretical guarantees that their market making strategies provide, whereas the focus in this paper is on exploring the utility of policies based on our representation of order profiles in realistic high-fidelity simulations. Moreover, the order profiles that our representations allow are much more flexible, generalising ladder strategies, which are recovered as a single parameter choice within our representation.

1.1.3 Market-making at the touch. Finally, it is worth discussing the problem of "market-making at-the-touch". This problem is considered in Cartea et al. [9, Chapter 10.2.2] and Cartea et al. [7] in a continous time and space mathematical model of the market. In the reinforcement learning literature, this approach is taken by Zhong et al. [37]. Here, the agent's actions consist in choosing to have an order or not at both the touch of the bid and ask sides of the book at every time step. By discretising the observation space similarly to [34], they achieve decent results with a Q-learning agent. This extremely simple action space fares quite well, and avoids the issues with the single price-level policy of Section 1.1.1, provided the

agent does not cancel and replace orders too frequently. Again, this policy can effectively be recreated using a beta policy, which we describe in detail in Section 3.1.

1.2 Our contributions
The key contributions of this paper are as follows:

- We introduce a new parametric representation of market maker policies via scaled beta distributions. We show how these policies capture – as special cases – standard actions spaces from the literature, including single-level orders, ladder strategies, and market making at the touch. The resulting continuous action space is far more flexible than these special cases and allows the market maker to skew orders to address the problem of accumulated inventory whilst maintaining their queue position.
- We have developed a high-fidelity order book simulator, using LOBSTER data which is combined with orders from our agents. We empirically evaluate our beta policies, showing first the benefit of non-uniform policies over ladder strategies.
- We then explore inventory management. First we demonstrate the cost of using market orders to control inventory. Motivated by this we use our scaled beta policies to design an inventory-driven policy that automatically skews the distributions to favour driving the absolute value of inventory back towards zero. The policy controls inventory using only limit orders.

The source code for our simulator and market-making agents is available at https://github.com/JJJerome/rl4mm.

2 PRELIMINARIES
2.1 Experimental setup
Our empirical evaluation uses 50 levels of limit order book data provided by LOBSTER[4]. The data is replayed in a custom-built data-driven high-fidelity simulator that integrates the historical orders with orders placed by test agents. We used data from the first two weeks of March 2022 for the following symbols for our evaluation.

Ticker	Description	Exchange	Sector
AXP	American Express	NYSE	Finance
BA	Boeing Company	NYSE	Industrials
BAC	Bank of America Corp.	NYSE	Finance
CAT	Caterpillar, Inc.	NYSE	Industrials
GE	General Electric	NYSE	Consumer Discretionary
HPQ	HP Inc.	NYSE	Technology
IBM	Int. Business Machines Corp.	NYSE	Technology
JNJ	Johnson & Johnson	NYSE	Health Care
JPM	JP Morgan Chase & Co.	NYSE	Finance
KO	Coca-Cola Company	NYSE	Consumer Staples
MMM	3M Company	NYSE	Industrials
TXN	Texas Instruments Inc.	NASDAQ-GS	Technology
WMT	Walmart Inc.	NYSE	Consumer Discretionary

Table 1: Tickers for our empirical evaluation.

2.2 The market-replay gym environment
To test our agents, we created a gym environment which mimics an exchange with price-time priority. The environment allows the agent to interact periodically and place limit orders, market orders, cancellations or deletions and accrue cash and inventory of the

---

[3]In doing so, they allow trading fractional units, which is not realistic. However, their focus is primarily theoretical.

[4]https://lobsterdata.com/

traded asset. Between these interaction times, ultra fine-grained (Nasdaq or NYSE) order data is replayed and the state of the order book is updated. By allowing the agent to place their own orders and interact with the order book directly, certain properties of real order books naturally arise. For example, the agent faces market impact when placing market orders and "walking the book".

The mechanism of running an episode is as follows:

(1) Choose a random start time for the episode such that the entire episode occurs within the random trading day.
(2) Initialise the order book using the top (50 levels) of the historical order book at that point in time.
(3) Let the agent choose their desired order distribution (levels and volumes) in the order book.
(4) Using the desired and existing orders, create orders that will achieve the desired orders. This may require cancelling orders from levels with too much volume, or placing limit orders. In addition, the agent may wish to place a market order to clear inventory if it is outside of some threshold amount.
(5) Replay the historical orders that occurred between the current time step and the next time step. In doing so update the agent's cash and inventory of the traded asset and track profit and loss (PnL) and any other desired quantities.
(6) Repeat steps (3)-(5) until the episode is finished.

The strengths and weaknesses of using market replay are discussed in [4]. Most notably, the main weakness of using market replay, compared to an interactive method such as an agent-based model, is the lack of adaptiveness of the future order flow to orders placed by the agent. In an agent-based simulator it is possible to include (exogenous) agents that react to the state of the order book when modified by the internal agent. However, agent-based models are notoriously hard to calibrate and so, whilst they possess reactivity to agents' actions, their realism is questionable.

In contrast, market-replay simulators are highly realistic. If the agent chooses not to place any orders in the book, then the evolution of the order book in the gym environment is, by definition, perfect. Furthermore, if we are willing to make the assumption that an agent placing small orders does not affect the future order flow too dramatically, then market replay should also accurately model the case with agent interaction. This is important to keep in mind when allowing the agent to interact with the book – if they constitute too large a proportion of the order flow volume then such an assumption is clearly violated.

2.3 Beta volume profiles

A market maker will continually place a set of bids and a set of asks. We first set a parameter, total_volume, which specifies the total amount of volume that the market maker will place on each side of the market. Having this quantity fixed is not as restrictive as it sounds, since the market maker can skew the volume distribution across prices so that an arbitrary proportion of volume can be far from the current best prices and thus very unlikely to be executed.

Let $f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)}$ be the probability density function of a beta distribution with parameters α and β, where $\mathrm{B}(\cdot,\cdot)$ is the beta function. It can then easily be confirmed by a change of variables

that the *scaled beta distribution* with probability density function

$$g(x) = \frac{1}{\texttt{n\_levels}} f\left(\frac{x}{\texttt{n\_levels}}\right) \tag{1}$$

and support $[0, \texttt{n\_levels}]$ is also a probability distribution.

We may therefore represent the distribution of the two sets of orders (for bids and asks) with two scaled beta distributions,

$$\textsc{BidOrders} \sim \textsc{ScaledBeta}(\texttt{n\_levels}, \alpha^{\mathrm{bid}}, \beta^{\mathrm{ask}}),$$

$$\textsc{AskOrders} \sim \textsc{ScaledBeta}(\texttt{n\_levels}, \alpha^{\mathrm{ask}}, \beta^{\mathrm{ask}}),$$

where n_levels is a fixed integer than specifies the support of the scaled distribution, which will correspond to the set of (contiguous) price levels at which the agent will quote.[5]

We will see in Section 3.1 that this approach generalises the three natural approaches taken in the literature – the single order policy, the ladder strategy and market making at the touch – whilst permitting significantly more flexibility.[6]

2.4 Implementing a beta volume profile

To implement a volume profile in practice, two steps must be taken. First, the *continuous* scaled beta distribution must be *quantised* to represent orders of integer size at integer levels. Next, the quantised distribution must be converted into orders and sent to the exchange.

Quantising the beta distribution. To decide how much volume to place at each level, the agent evaluates g from (1) at $x_i = i - 0.5$ for $i \in I := \{1, 2, \dots, \texttt{n\_levels}\}$ and normalises the results so that they sum to 1. This approximates the continuous ScaledBeta distribution by a discrete distribution with support I. Then, the agent scales this result by their total_volume and rounds the volume placed at each level to the nearest integer.[7] This is the agent's desired *limit order profile*.

Converting the quantised distribution into orders. The simulator keeps track of the agent's 'internal order book' – the limit orders belonging to the agent that are currently active. At each time step, the agent converts this order book into two vectors representing the current limit order profile held on the bid and ask sides of the order book. The agent then subtracts these vectors from the desired limit profile to get a signed volume difference which is converted into orders. If the difference is positive, then the agent places a limit order for that amount. If the order is negative, then the agent places cancellations for limit orders in the queue at that level. Here, we assume that volume is cancelled from the back of the queue first. This choice is based upon the assumption that, on average, having orders nearer the front of the queue is beneficial to the agent.

2.5 Representing a beta distribution by its mode and concentration.

The use of α and β to specify a beta distribution are arguably not as natural as using other distributional statistics such as the mean,

---

[5]All prices are relative prices measured away from the touch. If p_b^* is the best bid, p_a^* is the best ask and Δ is the minimum tick size, then the bid prices that the agent quotes at are $p_b^* - (i-1)\Delta$ and the ask prices are $p_a^* + (i-1)\Delta$ for $i \in \{1, 2, \dots, \texttt{n\_levels}\}$.
[6]To fully generalise the ladder strategy significantly, an extra parameter min_quote needs to be defined as the lowest level at which the agent places orders. We remark in Section 3.4 that we may set min_quote = 0 without losing much generality.
[7]This might cause the total volume of orders that the agent places in the book to deviate slightly from the parameter total_volume, however it is equal on average.

(a) $(\alpha^{\text{bid}}, \beta^{\text{bid}}) = (1,1), (\alpha^{\text{ask}}, \beta^{\text{ask}}) = (1,1)$ **(b)** $(\alpha^{\text{bid}}, \beta^{\text{bid}}) = (2,5), (\alpha^{\text{ask}}, \beta^{\text{ask}}) = (2,2)$ **(c)** $(\alpha^{\text{bid}}, \beta^{\text{bid}}) = (5,1), (\alpha^{\text{ask}}, \beta^{\text{ask}}) = (5,1)$ **(d)** $(\alpha^{\text{bid}}, \beta^{\text{bid}}) = (1,5), (\alpha^{\text{ask}}, \beta^{\text{ask}}) = (5,1)$
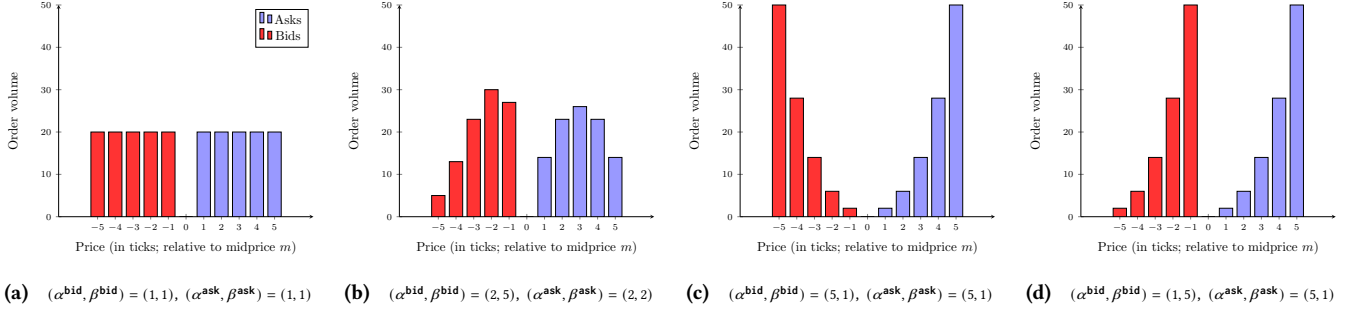
Figure 2: A range of different action choices: a) is a ladder strategy; b) is the example from Figure 1; c) skews the volume profile on both sides away from the best prices; d) skews bids towards the best prices and asks away from the best prices, so would be a sensible action to take to try and redress a negative inventory.

variance, or mode, which are more interpretable. In this paper, for our inventory-based policy we use the *mode* and *concentration* to specify beta distributions that correspond to the agent's desire for a mean-reverting inventory. [8] The following equations relate the mode and concentration to α and β.

Definition 2.1 (Mode and concentration). The concentration, κ of a beta distribution is defined as $\kappa = \alpha + \beta$. When $\alpha, \beta > 1$, which we will assume throughout this paper, the mode, ω is $\omega = \frac{\alpha-1}{\alpha+\beta-2}$. Given suitable ω and κ, the corresponding α and β are given by:

$$\alpha = \omega(\kappa - 2) + 1, \beta = (1 - \omega)(\kappa - 2) + 1. \tag{2}$$

In a slight abuse of notation, we use the parameter ω of the underlying Beta distribution to parametrise ScaledBeta. This can be thought of as the *proportion* of the n_levels at which the agent wants their mode, so that the mode for ScaledBeta is $\omega \cdot$ n_levels.

3 BETA POLICIES FOR MARKET MAKING

We assume that n_levels is fixed during an episode. Then, at each time step, the basic beta action is defined by the following 4-tuple, which defines two scaled beta distributions:

$$a = (\alpha^{\text{bid}}, \beta^{\text{bid}}, \alpha^{\text{ask}}, \beta^{\text{ask}}) \tag{3}$$

We also consider restrictions, where the concentration of the beta distribution is set as a constant and the mode is then varied, with (2) used to recover the corresponding α and β. This reduces the action space of the agent to be 2 dimensional and resembles a smoothed version of the *single price-level* action space.

3.1 Special cases of scaled beta distributions

Before discussing one of the main advantages of beta policies – the ability to alter one's skew while maintaining queue position – we explain how beta policies generalise the three types of prominent action spaces from the literature. As mentioned above, to emulate a single price-level policy (Section 1.1.1), one can set the variance of the beta distribution close to zero. To recreate a ladder strategy

one uses Beta(1,1). Finally, to emulate market making at the touch, to place volume at the touch the agent would choose parameters $\alpha = 1, \beta \gg 1$ so that the scaled beta distribution reduces to a Dirac delta at the first level. Similarly, if the agent wants to not place any volume at the touch, they can choose $\beta = 1, \alpha \gg 1$. Provided that n_levels is large enough, this amounts to placing orders far away from the touch. In particular, they are highly unlikely to ever get executed on that side.

3.2 Maintaining queue position

As described in Section 1.1.1, when placing orders at a single level, the agent must frequently cancel and replace orders at the levels they have chosen in that time step. Due to the price-time priority mechanism implemented by most major exchanges, this causes them to lose their queue position and join the back of the queue at the new level. However, this phenomenon is much weaker for scaled beta policies. This is because much of the agent's volume is unaltered when adjusting the desired mode of the distribution and so they may continuously update their action tuple, whilst not constantly requiring that they leave and rejoin the queue. This is illustrated in Figure 3, representing the change in order volume for an agent that wanted to change $\omega = 0.4$ to $\omega = 0.6$ in their ScaledBeta order distribution, whilst maintaining a concentration of $\kappa = 10$. In particular, the volume in the shaded region is unaltered so that they do not lose their queue position on it.
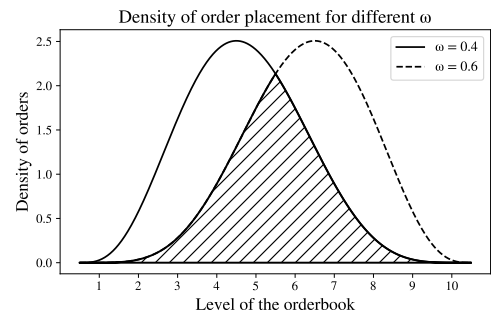

Figure 3: Book density for different values of the mode ω, with the concentration $\kappa = 10$ fixed.

---

[8]It is also possible to parameterise the distribution by its mean and variance. For a chosen mean μ and variance σ^2, α and β are given by $\alpha = ((1 - \mu)\sigma^{-2} - \mu^{-1})\mu^2$ and $\beta = \alpha(\mu^{-1} - 1)$. However, when choosing these parameters, it is necessary that $\sigma^2 < \mu(1 - \mu)$. This is problematic for an agent that wishes to *learn* these parameters, since the shape of the control space is unusual.

3.3 Why the beta distribution?

In this section, we explain why we chose the scaled beta distribution in terms of its advantages compared to prominent alternatives. For our application, its main benefits are that it:

(1) allows an independent choice of the mean and variance;[9]
(2) only requires two intuitive parameters ω and κ;
(3) has compact support over a fixed number of levels.

For (1), the flexibility in terms of choosing the mean and variance independently allow us to generate a wide variety of volume profile "shapes" with the scaled beta distribution; in particular we can emulate both ladder strategies (high variance) and single-price level actions (low variance). In contrast, alternatives such as the binomial distribution or the beta-binomial distribution, have strong constraints on the variance. The binomial distribution has a fixed variance for a given mean. The beta-binomial relaxes this slightly, but still constrains the variance within a fixed interval for a given mean. This would mean that certain low variance profiles that an agent may want to use would be unavailable to them.

For (2), we remark that having just two intuitive parameters, not only aids interpretability, but also simplifies any learning problems for which the volume profile parameters are the output. An alternative would be to use a distribution given by a polynomial or a neural network, but this would require significantly more parameters that are less interpretable.[10]

For (3), we remark that for computational and practical reasons it would be undesirable to have a policy that could place limit orders at arbitrarily many different price levels. It would be possible to use a distribution with unbounded support and then truncate; the scaled beta distribution's bounded support means that this is not necessary and we naturally place orders at no more than n_levels from the touch. It is for this reason that we prefer the scaled beta distribution to Gaussian, Poisson, gamma, and lognormal distributions.

3.4 A comparison of fixed beta policies

A ladder strategy is parametrised by its two end points. We consider intervals of 10 units of the minimum tick size and vary the "best price" of the ladder. Here, we vary the best price between -2 ticks, which means go within the spread by 2 ticks where possible, up to 7, which means start at 7 ticks outside of the touch.

We tested a variety of fixed ladder strategies using 60 randomly drawn hour-long episodes on the 13 assets over the timeframe outlined in Section 2.1. To compute "returns" we divided the profit or loss by the first price of the respective asset during the time period, and we report the mean and standard deviation of these 60×13 "returns" in each row. We also report the number of profitable tickers in terms of the total profit and loss (so the maximum possible in this column is 13). Table 2 shows that the basic ladder strategy never has a positive mean return for any minimum quote level, with the largest number of profitable tickers being 4.

$\alpha^{\text{bid}} = \alpha^{\text{ask}}$	$\beta^{\text{bid}} = \beta^{\text{ask}}$	min quote	# profitable	mean return	std returns
1	1	−1	0	−1.96	8.80
1	1	−2	3	−2.58	20.13
1	1	0	0	−1.06	5.22
1	1	1	0	−0.68	3.57
1	1	2	2	−0.48	3.27
1	1	3	4	−0.28	2.69
1	1	4	0	−0.28	3.08
1	1	5	1	−1.07	6.30
1	1	6	2	−0.24	2.87
1	1	7	1	−0.33	2.83

Table 2: Performance of ladder strategies for different minimum quote levels, from two ticks inside the spread (when possible) to seven ticks from the touch.

We next allow non-uniform beta actions, and sweep over some simple parameter combinations. In these experiments we always used minimum quote level 0. Recall that a non-uniform beta action can place its mode at any of the possible 10 quote levels. This means that the generality gained from adding a different minimum quote level is minor. We further ran some sweeps in which the minimum quote level was allowed to vary along with the paramters α and β, but found that the optimal value for the minimum quote level was close to zero. We do not include these tables due to space constraints.

The results of the sweep over fixed parameter beta policies is given in Table 3. Here, we see that the number of profitable tickers increases and for two non-uniform beta policies a positive mean return is obtained. The profitable fixed action policies were symmetric SCALEDBETA(10, 1, 2) and SCALEDBETA(10, 1, 5) policies, which correspond to BETA(1, 5) and BETA(2, 5) respectively and therefore are skewing the volume profile *towards* the best prices.

While we have demonstrated the potential of non-uniform beta policies, it is worth noting that their returns have large standard deviations (a simple measure of risk). We will see in Section 3.5 that this is due to a lack of inventory control demonstrated by such policies. We address this in the rest of this section.

$\alpha^{\text{bid}} = \alpha^{\text{ask}}$	$\beta^{\text{bid}} = \beta^{\text{ask}}$	min quote	# profitable	mean return	std returns
1	1	0	0	−1.06	5.22
1	2	0	6	0.58	16.04
1	5	0	7	0.06	35.66
2	1	0	3	−3.10	18.92
2	2	0	0	−1.39	5.92
2	5	0	3	−1.42	16.70
5	1	0	3	−3.06	29.63
5	2	0	7	−0.30	14.67
5	5	0	0	−1.40	6.76

Table 3: Performance of fixed actions, including ladder strategies, for minimum quote level 0 (at the touch).

3.5 Controlling inventory with market orders

The main source of risk for a market maker comes from holding inventory during adverse price swings. Therefore one of the main goals of a market maker is to make sure their inventory remains within some reasonable bounds. Figure 4 shows that the optimal fixed beta strategy found in Section 3.4 accumulates a huge negative inventory, causing it to face wild price swings. One of the options available to a market maker is to place a market order to liquidate some of their position in the asset.

---

[9]Technically, the mean and variance must be possible for the bounded random variable, and given a bounded support and choice of mean, the variance will be constrained. However, compared to alternative distributions the constraints are weak

[10]In Ganesh et al. [14], the authors use a fitted quadratic function to represent the exchange order book shape. However their goal here is not to represent an agent's policy with this approach, but rather to simulate the exchange order book from which they take liquidity to hedge a position gained from trading in over-the-counter markets.

This extra action of placing a market order (with size proportional to the agent's inventory) was considered in [34]. In their experiments they set the proportion to be 1, which means that the agent liquidates their entire inventory when it becomes too large.

To add this option to our agent with a scaled beta policy controlled by (3), we add two extra parameters: the first is the maximum absolute inventory (max_inv) that the agent is willing to hold, and if they surpass this level they place a market order to reduce their absolute inventory; the second is the fraction of their inventory that they liquidate in such a situation. Since our agents have a continuous action space, we decided that it was natural to turn the impulse control problem of choosing when to liquidate into a continuous control problem of choosing an appropriate risk limit defined in terms of the absolute inventory. Therefore, the agent's action 4-tuple in (3) becomes a 6-tuple,

$$a = (\alpha^{\text{bid}}, \beta^{\text{bid}}, \alpha^{\text{ask}}, \beta^{\text{ask}}, \text{max\_inv}, \text{frac\_inv}). \qquad (4)$$
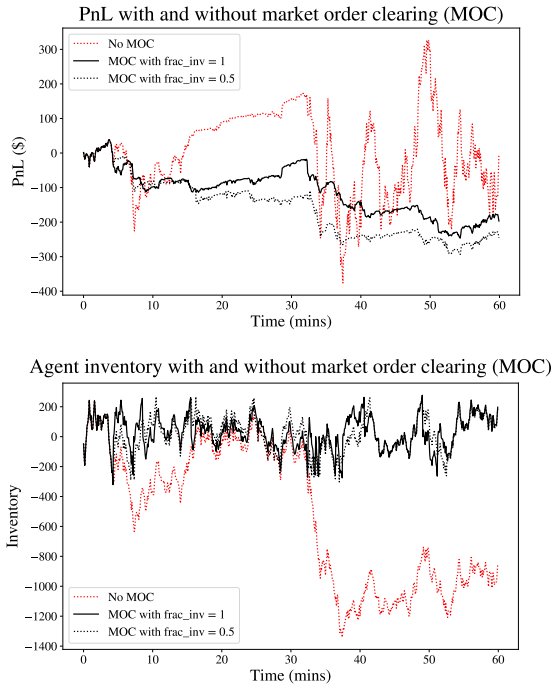

Figure 4: A comparison of the PnL and the inventory for an agent with the optimal fixed action $\alpha^{\text{bid}} = \alpha^{\text{ask}} = 1$ and $\beta^{\text{bid}} = \beta^{\text{ask}} = 2$ from the sweep in Section 3.4 and a variety of different market order inventory clearing policies. These plots are for the ticker JPM on the first date of the data (2nd March 2022) between 10:30 am and 11:30 am with max_inv = 250.

Figure 4 compares three different market order clearing policies (no market clearing, market clearing with max_inv = 1000 and frac_inv = 1, and market clearing with max_inv = 1000 and frac_inv = 0.5). In particular, the strategy with frac_inv = 1 manages inventory very well but has worse returns than that with frac_inv = 0.5. Both greatly outperform the agent with no market clearing. However, the agent still pays a cost for this risk management in the form of crossing the spread. In the next section,

an alternative strategy is introduced, which manages to control inventory, whilst only placing limit orders.

3.6 An inventory-driven policy

In the mathematical finance literature, it is well established that an inventory-aware strategy will skew its bid and ask according to its current inventory level. In particular, an optimal agent trading in a financial market with midprice process given by arithmetic Brownian motion and Poisson market order arrivals (see [3, 9]) should skew the midprice of their bid and ask quotes in the opposite direction to their asset holdings; for example, if they hold a negative inventory, they should skew their quotes so that their quoted midprice is higher than that of the market. Whilst the models of Avellaneda and Stoikov [3] and Cartea et al. [9] are not necessarily very realistic, this intuitively makes sense since skewing in this way makes it more likely that they will get filled on the side that brings their inventory closer to zero, helping them to complete round-trip trades and encouraging mean-reversion of their inventory (to zero).

It is this strand of literature that inspires our inventory-driven beta policy. In particular, the following parametric form ensures that the agent skews their active orders in such a way as to encourage mean-reversion. One can see that the parametric form given below implies that if the agent's inventory is close to their maximum desired inventory, then ω^{bid} is close to one (the mode of their bid orders is far away to the midprice), and ω^{ask} is close to zero (the mode of their ask orders is close to the midprice). When inventory is zero the distribution of the agent's orders will have a mode that is some proportion ω_0 (of the n_levels they quote at) into the book.

Inventory-driven policy

Parameters: concentration κ; max absolute inventory max_inv; exponent p; default value ω_0 for ω^{ask} and ω^{bid}. Set:

$$f_1(\text{inv}) := \omega_0 \left[1 + \left(\frac{1}{\omega_0} - 1 \right) \text{clamp} \left(\left| \frac{\text{inv}_t}{\text{max\_inv}} \right| \right)^p \right],$$

$$f_2(\text{inv}) := \omega_0 \left[1 - \text{clamp} \left(\left| \frac{\text{inv}_t}{\text{max\_inv}} \right| \right)^p \right],$$

$$\omega^{\text{bid}}(\text{inv}) := \mathbb{1}_{\text{inv} \geq 0} f_1(\text{inv}) + \mathbb{1}_{\text{inv} < 0} f_2(\text{inv}),$$

$$\omega^{\text{ask}}(\text{inv}) := \mathbb{1}_{\text{inv} < 0} f_1(\text{inv}) + \mathbb{1}_{\text{inv} \geq 0} f_2(\text{inv}),$$

where $\text{clamp}(x) = \min(1, \max(-1, x))$. Then, use (2) to set $\alpha^{\text{bid}}, \beta^{\text{bid}}, \alpha^{\text{ask}}, \beta^{\text{ask}}$ according to $\omega^{\text{bid}}, \omega^{\text{ask}}$ and κ.

Here, the exponent p controls the convexity of ω^{ask} as a function of inventory and generally takes a value $p \geq 1$. Such superlinear dependence of inventory on midprice skew is observed in the mathematical finance literature [9, Chapter 10]. We further generalise to the case when there exists a default value κ_0 and a maximum value $\kappa_{\max}$ for κ. In this case, we define

$$\kappa(\text{inv}) := (\kappa_{\max} - \kappa_0) \text{clamp} \left(\left| \frac{\text{inv}_t}{\text{max\_inv}} \right| \right)^p + \kappa_0.$$

The functions $\omega(\text{inv})$ and $\kappa(\text{inv})$ are plotted in Figure 5 for $\omega_0 = 0.2$, $\kappa_0 = 5$, $\kappa_{\max} = 20$ and $p = 2$.
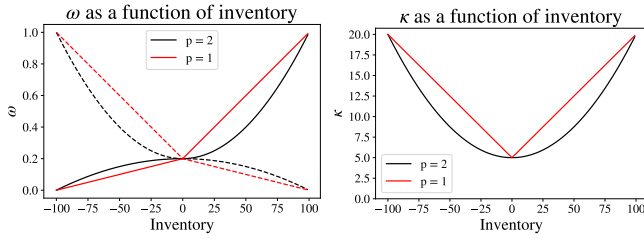
Figure 5: ω and κ as functions of Inventory. Here, the dotted line in the left hand figure is ω^{ask} and the solid line is ω^{bid}.

3.6.1 Mean reversion of inventory. The dependence of ω on inventory in the inventory-driven policy encourages mean-reversion of the inventory by skewing the best bid and the best offer. This can be seen in the bottom panel of Figure 6, in which the agent skews their midprice offset in the opposite direction to their inventory. This successfully induces a form of mean-reversion of the inventory levels.

Unlike the market order placing strategies of Section 3.5, the inventory-driven agent manages to do this without crossing the spread and incurring a cost. Comparing the PnL graph of the inventory-driven policy in the top panel of Figure 6 with the PnL graph of the various market order agents in the top panel of Figure 4 (which are all calculated over the same period), we see that such a strategy is effective. Finally, it is worth noting that the agent manages to capture the spread and make a profit even though the market is trending against them (they hold a negative inventory for the duration of the episode, but the drift is positive).

3.6.2 Tuned performance. When running experiments, we found that such a strategy is not robust across tickers and needed to be tuned. This could largely be due to the difference in relative tick size for different assets – an issue discussed in the subsequent section. We tuned for the ticker JPM and, across 300 one-hour episodes, obtained the following (in-sample) distribution for the "returns".

mean	0.52	While it still suffered large losses
std	5.50	on some episodes when it was un-
min	-38.47	able to control its inventory during
25%	0.26	highly trending periods, it managed
50%	1.20	to make money on 78% of episodes
75%	2.52	and had the best risk-reward profile
max	10.94	of all agents tested in the paper.

3.7 Beta policies and tick sizes

It is worth noting that the approach taken in this paper works best for assets whose relative tick sizes lie 'within a reasonable range'. If the relative tick size is too large (the asset is *tick-constrained*) then all activity happens at the top level of the book.[11] This means that, whilst there is benefit to maintaining queue position, it is unnecessary to have the control over the volume profile afforded by the beta distribution. The agent can just place block orders at a

---

[11]There is also a time component. If the agent acts every minute instead of every second, then by their perception of time, movements in the best bid and ask occur more frequently. In this case, the approach of this paper is appropriate again.
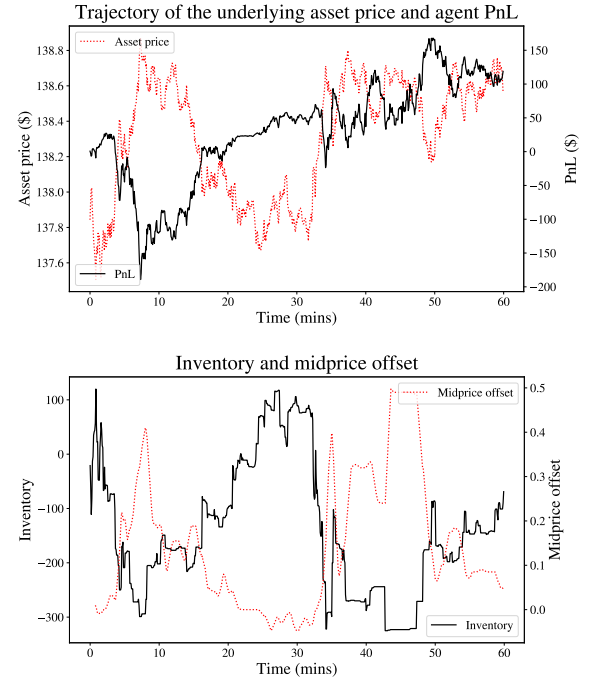

Figure 6: PnL for the inventory-driven agent compared with: the underlying asset price; the agent's inventory; and their midprice offset. These plots were generated for the ticker JPM and the same period as used in Figure 4.

reasonable quantity of levels as in a ladder strategy and then cancel volume at the touch when they want to reduce fills.

If the relative tick size is too small, then the beta volume profile also needs to be modified. This is because it is not sensible to post limit orders at every level: firstly it is computationally expensive to track the agent's orders at so many levels; secondly, if implemented in practice it would telegraph the agent's presence to all of the other market participants.

A naive solution to the issues presented by small-tick stocks would be to only post orders at uniformly-spaced price levels. However, this would not disguise the agent's presence at all. A better solution would be to post orders at randomly spaced price levels that are reasonably close to the desired uniformly spaced price levels. This approach would require further development though, since the random lattice would need to be updated more slowly than the agent takes actions (or the agent would end up constantly placing and removing orders).

The parameter `n_levels` is also sensitive to the relative tick size of the asset. A more robust approach would be to use a beta distribution to characterise volume profiles across price levels given in terms of basis points up to a value `max_bps`. After adding some randomisation to disguise the agent's presence in the market and rounding to the nearest tick, this approach would be robust across a range of asset sizes and could improve generalisability of learnt strategies substantially. This is an easy addition to the code-base and will be included in future work.

4 CONCLUSIONS AND FURTHER WORK

We have introduced a new representation for market maker policies in limit order book markets that derive limit order volume profiles from scaled beta distributions. This – in contrast to most work in the AI4MM literature – is a *continuous action space*, which makes it highly expressive. It further encompasses the key special cases, of single orders, ladder strategies, and market making at the touch, that have previously been studied in the AI4MM literature. However, the approach is also significantly more general in terms of the market maker's ability to simultaneously skew volume to favour orders on one side of the market and maintain queue position, while using only a small number of easy-to-interpret parameters.

We believe that scaled beta volume profiles can form the foundation for sophisticated and performant market marking agents based on state-of-the-art learning approaches. For example, the following directions for further work seem promising:

- Bayesian optimization would be a natural approach to tune the parameters of our inventory-based control policy since sampling trajectories is relatively costly.
- Abernethy and Kale [2] treat different ladder strategies as "experts" and use multiplicative weights updates to dynamically select mixtures of these experts; it would be interesting to treat scaled beta policies in the same way. A key difference between the set of ladder strategies and scaled beta volume profiles is that the former set of strategies is finite whereas the latter is infinite; suitable approaches for infinite mixtures exist, e.g., [30]. In [2] they allow fractional volumes arising from mixtures of ladders; for realism, ideally a mixture approach would round volumes to integers, while still maintaining good performance.
- Arguably the most natural AI4MM approach to use with scaled beta volume profiles is *Reinforcement Learning* (RL). With this approach, in each step scaled beta actions would be chosen based on features that describe both the market state (e.g. trendiness and volatility, and limit order book features that are known to have short term predictive value such as the book imbalance [5]) and the market marker's state (e.g. inventory). It would be extremely interesting to explore the use of state-of-the-art continuous control RL methods with a rich state space.

ACKNOWLEDGMENTS

REFERENCES

[1] Frédéric Abergel, Marouane Anane, Anirban Chakraborti, Aymen Jedidi, and Ioane Muni Toke. 2016. *Limit Order Books*.
[2] Jacob D. Abernethy and Satyen Kale. 2013. Adaptive Market Making via Online Learning. In *Proc. of NIPS*.
[3] Marco Avellaneda and Sasha Stoikov. 2008. High-frequency trading in a limit order book. *Quantitative Finance* 8, 3 (2008), 217–224.
[4] Tucker Hybinette Balch, Mahmoud Mahfouz, Joshua Lockhart, Maria Hybinette, and David Byrd. 2019. How to Evaluate Trading Strategies: Single Agent Market Replay or Multiple Agent Interactive Simulation? *CoRR* abs/1906.12010 (2019). arXiv:1906.12010 http://arxiv.org/abs/1906.12010
[5] Jean-Philippe Bouchaud, Julius Bonart, Jonathan Donier, and Martin Gould. 2018. *Trades, quotes and prices: financial markets under the microscope.* Cambridge University Press.

[6] Álvaro Cartea, Ryan Donnelly, and Sebastian Jaimungal. 2017. Algorithmic Trading with Model Uncertainty. *SIAM Journal on Financial Mathematics* 8, 1 (2017), 635–671.
[7] Alvaro Cartea, Ryan Donnelly, and Sebastian Jaimungal. 2018. Enhancing trading strategies with order book signals. *Applied Mathematical Finance* 25, 1 (2018), 1–35.
[8] Álvaro Cartea and Sebastian Jaimungal. 2015. Risk Metrics and Fine Tuning of High-Frequency Trading Strategies. *Mathematical Finance* 25, 3 (2015), 576–611.
[9] Álvaro Cartea, Sebastian Jaimungal, and José Penalva. 2015. *Algorithmic and High-Frequency Trading.* Cambridge University Press.
[10] Álvaro Cartea, Sebastian Jaimungal, and Jason Ricci. 2014. Buy low, sell high: A high frequency trading perspective. *SIAM Journal on Financial Mathematics* 5, 1 (2014), 415–444.
[11] Tanmoy Chakraborty and Michael Kearns. 2011. Market Making and Mean Reversion. In *Proc. of ACM EC*. 307–314.
[12] Nicholas T. Chan and Christian R. Shelton. 2001. *An Electronic Market-Maker.* AI Memo 2001-005. MIT AI Lab.
[13] Rama Cont, Sasha Stoikov, and Rishi Talreja. 2010. A Stochastic Model for Order Book Dynamics. *Oper. Res.* 58, 3 (2010), 549–563.
[14] Sumitra Ganesh, Nelson Vadori, Mengda Xu, Hua Zheng, Prashant Reddy, and Manuela Veloso. 2019. Reinforcement Learning for Market Making in a Multi-agent Dealer Market. *arXiv:1911.05892* (2019).
[15] Bruno Gasperov and Zvonko Kostanjcar. 2021. Market Making With Signals Through Deep Reinforcement Learning. *IEEE Access* 9 (2021), 61611–61622.
[16] Lawrence R Glosten and Paul R Milgrom. 1985. Bid, Ask and Transaction Prices in a Specialist Market with Heterogeneously Informed Traders. *Journal of Financial Economics* 14, 1 (1985), 71–100.
[17] Sanford J Grossman and Merton H Miller. 1988. Liquidity and Market Structure. *The Journal of Finance* 43, 3 (1988), 617–633.
[18] Olivier Guéant, Charles-Albert Lehalle, and Joaquin Fernandez-Tapia. 2011. Dealing with the Inventory Risk: A solution to the market making problem. *Mathematics and Financial Economics* 7, 4 (2011), 477–507.
[19] Fabien Guilbaud and Huyen Pham. 2011. Optimal High Frequency Trading with limit and market orders. *CoRR* abs/1106.5040 (2011).
[20] Joel Hasbrouck and Gideon Saar. 2013. Low-latency trading. *Journal of Financial Markets* 16, 4 (2013), 646–679.
[21] Richard Haynes and John S Roberts. 2015. Automated Trading in Futures Markets. *CFTC White Paper* (2015).
[22] Thomas Ho and Hans R Stoll. 1981. Optimal Dealer Pricing Under Transactions and Return Uncertainty. *Journal of Financial Economics* 9, 1 (1981), 47–73.
[23] Joseph Jerome, Leandro Sánchez-Betancourt, Rahul Savani, and Martin Herdegen. 2022. Model-based gym environments for limit order book trading. *arXiv preprint arXiv:2209.07823* (2022).
[24] Adlar J Kim and Christian R Shelton. 2002. Modeling stock order flows and learning market-making from data. (2002).
[25] Michail G Lagoudakis and Ronald Parr. 2003. Least-Squares Policy Iteration. *JMLR* 4 (2003), 1107–1149.
[26] M. Leaver and T. W. Reader. 2016. Human Factors in Financial Trading: An Analysis of Trading Incidents. *Human Factors* 58, 6 (2016), 814–832.
[27] Ye-Sheen Lim and Denise Gorse. 2018. Reinforcement Learning for High-Frequency Market Making. In *Proc. of ESANN*.
[28] Kaj Nyström, Sidi Mohamed Ould Aly, and Changyong Zhang. 2014. Market making and portfolio liquidation under uncertainty. *International Journal of Theoretical and Applied Finance* 17, 05 (2014), 1450034.
[29] Yagna Patel. 2018. Optimizing Market Making using Multi-Agent Reinforcement Learning. *arXiv preprint arXiv:1812.10252* (2018).
[30] Carl Edward Rasmussen and Zoubin Ghahramani. 2001. Infinite Mixtures of Gaussian Process Experts. In *NIPS*. MIT Press, 881–888.
[31] Jonathan Sadighian. 2019. Deep reinforcement learning in cryptocurrency market making. *arXiv preprint arXiv:1911.08647* (2019).
[32] Rahul Savani. 2012. High-frequency trading: The faster, the better? *IEEE Intelligent Systems* 27, 4 (2012), 70–73.
[33] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017). arXiv:1707.06347 http://arxiv.org/abs/1707.06347
[34] Thomas Spooner, John Fearnley, Rahul Savani, and Andreas Koukorinis. 2018. Market Making via Reinforcement Learning. In *Proc. of AAMAS*. 434–442.
[35] Thomas Spooner and Rahul Savani. 2020. Robust Market Making via Adversarial Reinforcement Learning. *Proc. of IJCAI* (2020).
[36] Ziyi Xu, Xue Cheng, and Yangbo He. 2022. Performance of Deep Reinforcement Learning for High Frequency Market Making on Actual Tick Data. In *Proc. of AAMAS*. 1765–1767.
[37] Yueyang Zhong, YeeMan Bergstrom, and Amy R. Ward. 2020. Data-Driven Market-Making via Model-Free Learning. In *Proc. of IJCAI*. 4461–4468.