Q-Learning and Algorithmic Market Making: Loss-free, Collusive, or Competitive Prices?

Antonio Guarino[*] Philippe Jehiel[†] James Symons-Hicks[‡]

July 8, 2025

Abstract

We study whether Algorithmic Market Makers using Q-learning produce competitive or supra-competitive prices in a quote-driven asset market. We show, through simulations and analytically, that the result depends on the way the algorithm is set up. A basic Q-learning algorithm leads to loss-free prices and is, therefore, not fit for trade. Carefully choosing the exploration and learning parameters leads to less extreme prices, but still far away from the competitive ones. When we endow the algorithm with a basic understanding of the market and basic information about outstanding quotes, the Q-learning algorithms produce competitive prices.

[*] Guarino: Department of Economics, University College London and CEPR (email: a.guarino@ucl.ac.uk).

[†] Jehiel: PSE and University College London (email: jehiel@enpc.fr).

[‡] Symons-Hicks: London School of Economics (email: j.symons-hicks@lse.ac.uk).

We are grateful to Xiaotong Wu and Yihan Xu for outstanding research assistance. We acknowledge the use of the UCL Myriad High Performance Computing (Myriad@UCL) and the LSE Fabian High Performance Computing facilities, and associated support services, in the completion of this work.

1 Introduction

Does the use of algorithms to price financial assets lead to competitive or supra-competitive prices? The answer that we provide in this paper is: it depends on the design of the algorithm. A basic Q-learning algorithm produces loss-free prices, which essentially makes the algorithm unsuitable for trading; a Q-learning algorithm with some carefully chosen learning and experimentation parameters leads to pricing somewhere in between the competitive and loss-free pricing levels, which can be viewed as a form of supra-competitive pricing; however, endowing the algorithm with the minimal market understanding that pricing more aggressively than competing market makers results in capturing the entirety of the trading activity, together with minimal information about the market prices, leads to the algorithms setting competitive prices. Overall, while the existing literature on machine learning in economics and finance emphasizes the negative effects of the use of algorithmic pricing in terms of supra-competitive prices (often linked to tacit collusion), our work shows that there is nothing intrinsic to the use of machine learning that leads to supra-competitive prices; when supra-competitive prices arise, they are not due to collusion but to the avoidance of losses in a setting in which fundamental values are not deterministically defined.

Our work is based on the seminal paper by Glosten and Milgrom (1985). We consider a one-period version of the model and we change it in one fundamental dimension: while in the original model market makers set competitive quotes (ask and bid prices) to trade with informed and uninformed traders, in our work, prices are set by Algorithmic Market Makers (AMMs) that use a Q-learning algorithm. With Q-learning, a specific example of reinforcement learning, these AMMs work as follows. They attach a specific value to each possible ask and bid price and then proceed by trial and error over the course of time. Sometimes they "exploit" (choosing the price that has the highest attached value) and sometimes they "explore" (experimenting with random prices). The payoff that they receive after setting prices is the feedback (the immediate reward) used to update the value that they attach to prices; they then use the updated values to choose the new prices at the next trading time.

In our first version of the model, AMMs only update the Q-value for the specific ask and bid they have chosen. We find that, eventually, they choose loss-free prices; that is, they choose an ask weakly higher than the maximum value of the asset and a bid weakly lower than the minimum value of the asset. At these prices, informed traders have no incentive to trade. Noise traders only trade because we assume that they are perfectly price inelastic, otherwise we would have a complete market breakdown: a no-trade result. These prices are obviously extreme and show that this version of Q-learning cannot be used for market making. To make this point, we also let one AMM compete with a "human" market maker who sets a fixed ask (bid) price above (below) the competitive ask (bid) price equilibrium. The AMM never trades, thus leaving all the profits to the other market maker. These results are surprising, perhaps even unsettling. One could have imagined that, through reinforcement, AMMs would learn to "undercut" one another and, at least, to undercut a market maker with a fixed price, thus making profits. Here, instead, we have AMMs that are completely crowded out of the market. While surprising, this loss-free pricing result has a simple explanation, which we prove formally. Intuitively, it can be understood as follows. Any pricing that would not be loss-free would result in the possibility of making a loss if this market maker were to trade and the realization of the fundamental value were unfavorable. By the rule of the Q-learning algorithm, this implies that any pricing that is not loss-free will eventually (after sufficiently many unlucky draws) become associated with a negative Q-value, at which point, in the absence of experimentation, it will no longer be used. Our result is related to the *maxmin* result obtained by Sarin and Vahid (1999) in an individual decision-making problem. In fact, that result can be immediately used to prove that one AMM would choose loss-free prices against a fixed-price market maker, whereas a novel analysis is needed to prove that a similar result holds when many AMMs compete in the market.

After showing that modifying some aspects of the Q-learning algorithm, such as the learning and exploration rates, helps to have less extreme, but still supra-competitive, prices (due to the same mechanism underlying the *maxmin* result, and not to collusion), we move to a Q-learning model in which AMMs do not update the Q-values only for the prices they chose, but also for other prices for which they can make an obvious inference. For instance, if an AMM sells at an ask price of 100, they realize that they would have also sold at lower prices, and, thus, update the Q-values for all prices less than or equal to 100. Understanding that traders happy to buy at a specific ask price would also be happy to buy at a lower price is a minimal requirement. We call this model "counterfactual updating". While this model seems perfectly natural in our specific pricing problem, it has also been used, more broadly, in Game Theory, to study learning by players who may think about the counterfactual choices of their opponents (see, e.g., the "Experience-Weighted Attraction" (EWA) model of Camerer and Ho, 1999). In this model, we show, both through simulation and analytical results (making use of mathematical results in the dynamics of stochastic approximation algorithms - see Benaïm, 1999) that algorithmic prices converge to the competitive equilibrium prices. Furthermore, in the case in which there is only one AMM competing with a "human" market maker who uses the simple pricing strategy of a fixed ask price (set higher than the competitive ask price) and of a fixed bid price (set lower than the competitive bid price), the AMM does exactly what one would hope they do: they undercut the competitor and earn all trading profits. The reason for these results is that, with counterfactual updating (and vanishing small weights on the current feedback/reward), the AMMs obtain a lot of information and one can reason in terms of expected profits, thereby leading the AMMs to assess that the undercutting pricing policy would be strictly beneficial as long as the current market price is supracompetitive. These results are obtained using conditions on the learning rate that are very common in machine learning. Indeed, they are the Robbins-Monro conditions also used in the classic proof of convergence of Q-learning to the optimal policy by Watkins and Dayan (1992). In contrast with their work, however, in our model, we do not need experimentation for our results. Intuitively, counterfactual updating already provides enough information to the algorithms that they do not need to experiment to learn to undercut.

Our work contributes to the finance literature with an application of machine learning to a classic market microstructure problem. In studying how using Q-learning affects prices in the classic Glosten and Milgrom (1985) model, we also use insights from the literature on learning in Decision Theory (Sarin and Vahid, 1999) and in Game Theory (e.g., Camerer and Ho, 1999), and base some of our proofs on mathematical results in stochastic approximation (Benaïm, 1999) The closest paper to ours in terms of research question is Colliard et al. (2022). The models are quite similar, although we base our work on the standard Glosten and Milgrom (1985) model with informed and noise traders, whereas they have one type of trader but there are common and private values. The results are markedly different. They insist on the supra-competitive prices outcome, which they show through simulations ("an experimental approach" to use the authors' terminology) and attribute these results to limited learning capacity. In contrast, we show that prices vary from one extreme, the loss-free outcome, to another, the competitive equilibrium outcome, depending on how the Q-learning algorithm is designed. Our results are analytical and not only shown through simulations. The paper by Dou et al. (2024) has in common with our work that they too are interested in the (non-) competitive outcomes of Q-learning in a market with private information; however, they focus on AI informed traders, and not on market makers. Cartea et al. (2022) study learning algorithms in a market with no private information (essentially a Bertrand game with complete information) and their interest is in the effect of the tick size on market outcomes.

More broadly, in the economics literature, there has been interest in algorithmic pricing in oligopoly markets. One notable example is Calvano et al. (2020). In their setup with history-dependent states, they find that "algorithms consistently learn to charge supracompetitive prices, without communicating". In a similar setup to that of Calvano et al. (2020), Asker et al. (2024) study how “synchronous” learning leads to prices close to the competitive levels.

2 The Model

We consider a sequential trade model based on the seminal paper of Glosten and Milgrom (1985). Following Easley et al. (1997) and Cipriani and Guarino (2014), we consider a multiple-day version of the model. A single risky asset is traded in a specialist market over multiple days by informed and noise traders. The novelty of our model is that the specialist (also known as a market maker) sets ask and bid prices using a Q-learning algorithm. Given our purposes, rather than considering the entire sequence of trades in each day, we will specialize the analysis to the first trading time of each day.

2.1 The Asset

There is one risky asset, and its fundamental value in day d ($d = 1, ..., D$) is denoted by V^d. The asset value does not change during the day, but can change from one day to the next. Each day, with probability δ, the value of the asset, V^d, is equal to v_H and, with probability $1 - \delta$, is equal to v_L, where $v_H > v_L \geq 0$ and $\delta v_H + (1 - \delta)v_L = \overline{v}$.[1] At the end of the trading day, the value of the asset is known to all market participants.

2.2 The Market

The asset is exchanged in a specialist market. Its price is set by market makers who interact with a sequence of traders. Each trading day consists of T trading times, also referred to as trading periods. As we said, for our purposes, we will consider only one period; this is equivalent to setting $T = 1$.

Each day, a trader is randomly chosen to act and can buy, sell, or decide not to trade. Each trade consists of the exchange of one unit of the asset for cash. The trader's action space is, therefore, $\mathcal{A}$ =$\{-1, 0, 1\}$, where 1 and -1 are interpreted as the trader buying one unit and selling one unit of the asset, respectively. We denote the action of the trader in day d by X^d.

[1] Note that v_H and v_L are the realizations of the random variable V^d. Throughout the text, we will denote random variables with capital letters and their realizations with lowercase letters.

2.2.1 The Market Makers

Each day d, N market makers, indexed by $i = 1, ..., N$, set the prices at which a trader can buy or sell the asset.

We denote market maker i's ask price (the price at which a trader can buy) by a_i^d and, similarly, the bid price (the price at which a trader can sell) by b_i^d. Traders trade at the best ask and bid prices, denoted as follows:

$$a^d = \min\{a_1^d, ..., a_N^d\} \quad \text{and} \quad b^d = \max\{b_1^d, ..., b_N^d\}. \tag{1}$$

In the case that there are multiple market makers quoting the best price, and a trader wants to trade at that price, one market maker is randomly chosen, with equal probability, from the set of market makers quoting that price. We denote the market maker with the prevailing ask and bid quotes by i_{ask}^* and i_{bid}^*. Each day, market maker i's trade is equal to

$$\Phi_i^d = \begin{cases} 1 & \text{if } X^d = -1 \text{ and } i = i_{ask}^*, \\ -1 & \text{if } X^d = 1 \text{ and } i = i_{bid}^*, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Market maker i's profit on day d is equal to

$$\pi_i^d = (a_i^d - v^d)\mathbf{1}_{\{\Phi_i^d=-1\}} + (v^d - b_i^d)\mathbf{1}_{\{\Phi_i^d=1\}}, \tag{3}$$

where $\mathbf{1}_{\{.=.\}}$ is the indicator function.

2.2.2 The Traders

There is an uncountable number of traders. Each trader is chosen to take an action only once. Traders are of two types: informed and noise. The trader's own type is private information. On any day d, an informed trader is chosen to trade with probability μ and a noise trader with probability $1 - \mu$, with $\mu \in (0, 1)$. Noise traders buy with probability $\frac{\eta}{2}$, sell with probability $\frac{\eta}{2}$, and do not trade with probability $1 - \eta$ (with $0 \leq \eta \leq 1$).

Informed traders have private information on the value of the asset. In particular, they receive a perfectly informative private signal on the asset value; that is, they know the realization of V^d.

An informed trader's profit on day d is defined as

$$u^d = (v^d - a^d)\mathbf{1}_{\{X^d=1\}} + (b^d - v^d)\mathbf{1}_{\{X^d=-1\}}. \tag{4}$$

An informed trader chooses X^d to maximize expected profits; that is, they are risk neutral. Since a trader knows the asset value, they find it optimal to buy whenever $a^d < v^d$, and sell whenever $b^d > v^d$. They choose not to trade when $b^d < v^d < a^d$. Otherwise, they are indifferent between buying and not trading, or selling and not trading.[2]

3 Competitive Equilibrium

Before discussing algorithmic market making, it is worth discussing the competitive equilibrium.

Market makers face an adverse selection problem since a trader deciding to trade may be doing so because they know the value of the asset. In a perfectly competitive market, market makers make zero profits in expected value. Therefore, the prevailing quotes are equal to the expected value of the asset, conditional on trade at the quoted prices:

$$a^d = E(V^d | X^d = 1, a^d, b^d), \tag{5}$$

$$b^d = E(V^d | X^d = -1, a^d, b^d). \tag{6}$$

Solving Equation 5 and Equation 6, we obtain:[3]

$$a^d = \frac{(1-\mu)\eta E[V^d] + 2\mu\delta v_H}{2\mu\delta + (1-\mu)\eta} \equiv a^C, \tag{7}$$

$$b^d = \frac{(1-\mu)\eta E[V^d] + 2\mu(1-\delta) v_L}{2\mu(1-\delta) + (1-\mu)\eta} \equiv b^C. \tag{8}$$

4 Algorithmic Market Making

We now consider the case in which the function of market making is delegated to machines that set the ask and bid prices according to a Q-learning algorithm (Watkins, 1989). While in

[2] We are implicitly assuming that $a^d > b^d$, which is typically the case. If, instead, that were not the case, a trader may find it profitable both to buy *and* to sell the asset. In such a case, the informed trader chooses the action that earns the highest profit.

[3] In our simple setup, there are unique fixed points solving Equation 5 and Equation 6. In general, the ask is the minimum solution for Equation 5 and the bid is the maximum solution for Equation 6. See, e.g., Cipriani and Guarino (2008) for the formal argument.

the analysis of the competitive equilibrium, as standard, we have assumed that the model is common knowledge and that rational market makers and rational informed traders maximize their objective function on the basis of all available information, the starting point of algorithmic market making is that the machines have limited knowledge of the environment in which they are operating. The machines are used because they can exploit large data to learn the best pricing strategy, with minimal informational input.

The N Algorithmic Market Makers (AMMs) operating in our financial market set the prices based on a Q-learning algorithm, an example of off-policy reinforcement learning. The term "off-policy" refers to the fact that the algorithm uses a behavioral policy different from the optimal one. In particular, we consider an ε-*greedy* policy: each day,AMM$_i$ chooses the greedy action (i.e., chooses the price with the highest perceived payoff) with probability $1-\varepsilon_i^d$ and the explorative action (i.e., chooses a random price from its action set) with the complementary probability, ε_i^d. Later, in Section 7, we will also consider a different policy.

We consider the case in which the algorithms have very limited knowledge of the environment and set their prices without conditioning on any specific variable. This is often referred to as "stateless Q-learning," although some authors prefer to say that the state space is a singleton.

On each day d, each market maker i chooses the ask price a_i^d in the discrete space $\mathbf{A} = \{\alpha_1, \alpha_2, ..., \alpha_J\}$, where $\alpha_1 < \overline{v}$ and $\alpha_J > v_H$. Similarly, they choose the bid price b_i^d in the discrete space $\mathbf{B} = \{\beta_1, \beta_2, ..., \beta_J\}$, where $\beta_1 < v_L$ and $\beta_J > \overline{v}$. Each market maker attaches a particular value, called a "Q-value", $q_i^d(\alpha_j)$ to each possible ask price, and, similarly, a value $\hat{q}_i^d(\beta_j)$ to each possible bid price. On the first day, $d = 1$, the Q-vectors q_i^1 and $\hat{q}_i^1$ take random values. These values represent the perceived reward for choosing a given price.

On each following day, the Q-vectors are updated on the basis of the payoffs that the market maker receives that day. In particular, the Q-learning rule is as follows. If AMM$_i$ chooses ask $a_i^d = \alpha_j$ and bid price $b_i^d = \beta_k$, then the Q-values for these particular quotes are updated as follows:

$$\begin{aligned} q_i^{d+1}(\alpha_j) &= \lambda_i^d \left(\left(\alpha_j - v^d \right) \mathbf{1}_{\{\Phi_i^d = -1\}} \right) + \left(1 - \lambda_i^d \right) q_i^d(\alpha_j), & (9) \\ \hat{q}_i^{d+1}(\beta_k) &= \lambda_i^d \left(\left(v^d - \beta_k \right) \mathbf{1}_{\{\Phi_i^d = 1\}} \right) + \left(1 - \lambda_i^d \right) \hat{q}_i^d(\beta_k), & (10) \end{aligned}$$

where $\lambda_i^d \in [0,1]$ is the weight that market maker i places on the realized payoff on that day (the "learning rate").[4]

For all other (i.e., unchosen) ask prices α_l ($l \neq j$) and bid prices β_m ($m \neq k$), the Q-values are unchanged:

$$q_i^{d+1}(\alpha_l) = q_i^d(\alpha_l), \tag{11}$$

$$\hat{q}_i^{d+1}(\beta_m) = \hat{q}_i^d(\beta_m). \tag{12}$$

As we said, on each day, AMM$_i$ chooses to "exploit" or "explore", with probability $1 - \varepsilon_i^d$ and ε_i^d, respectively. The value of ε_i^d is given by

$$\varepsilon_i^d = c_i + (1 - c_i)e^{-\gamma_i d}, \tag{13}$$

where c_i is a constant (weakly between 0 and 1) and γ_i is a parameter (weakly larger than 0) that controls the rate at which this probability converges to the constant. If a market maker chooses to exploit on a given day, they do so for both the ask and bid prices; the same is true if they choose to explore. The market maker exploits the greedy action by choosing the ask price α_j associated with the highest value of q_i^d and the bid price β_k associated with the highest value of $\hat{q}_i^d$. That is, the greedy ask and bid prices of AMM$_i$ on day d are given by

$$a_i^{d*} = \arg\max_{\alpha_j} q_i^d(\alpha_j), \tag{14}$$

$$b_i^{d*} = \arg\max_{\beta_k} \hat{q}_i^d(\beta_k). \tag{15}$$

When, instead, the AMM explores, α_j and β_k are drawn independently from uniform distributions over $\mathbf{A}$ and $\mathbf{B}$, respectively.

Note that, in our analysis, we will also consider a variant of this model, with only one AMM ($N = 1$) and one "human" market maker. Nevertheless, the general analysis is for $N \geq 2$ AMMs that compete among themselves.

[4] Note that the chosen ask (bid) is updated independently of whether there is a buy (sell) from the trader, trade occurs on the other side of the market, or there is no trade. An alternative formulation could consist in updating the Q-value for the chosen ask (bid) only when the trader buys (sells). We will come back to this issue when we present the results in Section 6.

5 Implementation of the Baseline Case

Let us now implement our financial market model with AMMs through simulations. We run each simulation for one million days, to give plenty of time for price convergence. Of course, taken literally, one million days is an unrealistically long period. We use this terminology mainly for exposition. Day d can be thought of as any unit of time after which the asset value changes, and the process of trading restarts, so that the market maker faces exactly the same situation, that is, the same pricing problem.[5] We repeat the simulations 1,000 times and then present average results over these repetitions.

For our baseline simulation, we specialize the analysis as follows.

First, we set the two asset values to be $v_H = 102$ and $v_L = 98$, and the probability of these two values to be $\delta = 0.5$. This implies that the mean value of the asset, $\overline{v}$, is equal to 100. Moreover, we set the probability of an informed trader to $\mu = 0.3$, and the probability of of a noise trader buying or selling is $\frac{\eta}{2} = \frac{1}{2}$. As a tie-breaking rule, we assume that informed traders trade when they are indifferent between trading and not (the cases of indifference can arise for $a^d = v_H$ and for $b^d = v_L$).

Second, we consider the case in which there are $N = 2$ AMMs. They both choose ask prices in $\mathbf{A} = \{\alpha_1, \alpha_2, ..., \alpha_J\}$, where, $\alpha_1 = 99.5$ and $\alpha_J = 103$. Furthermore, we assume that the other ask prices are evenly spaced between these two values, with a tick size of 0.05. Note that $\mathbf{A}$ satisfies the conditions indicated in Section 4.1, that is, $\alpha_1 < \overline{v}$ and $\alpha_J > v_H$. Similarly, they choose bid prices in $\mathbf{B} = \{\beta_1, \beta_2, ..., \beta_J\}$, where, $\beta_1 = 97$ and $\beta_J = 100.5$, and the other bid prices are evenly spaced between these two values, with a tick size of 0.05. Overall, market makers choose among 71 ask and 71 bid prices, that is, $J = 71$.

Third, we specify how the Q-learning algorithm works. We assume that both market makers use the same learning rate $\lambda_i^d = \lambda = 0.1$, constant over all days. They explore with probability $\varepsilon_i^d = \exp(-0.00004d)$, that is, $c_i = 0$ and $\gamma_i = 0.00004$.

Finally, we assume that the initial Q-values $q_i^1(\alpha_j)$ and $\hat{q}_i^1(\beta_j)$ for each AMM are independently drawn from the same uniform distribution on $[5, 8]$.

For the reader's convenience, we summarize all parameter values in Table 1.

A few comments are in order. The values for the asset and for the traders are chosen for

[5] For instance, Easley et al. (2012) advocate the use of volume time rather than clock time. In their framework, the process of trading "restarts" every time a "volume bucket" is filled.

Table 1: Baseline Simulation

Parameter	Value	Description
The Simulation		
K	1,000	Number of simulation repetitions
D	1,000,000	Number of trading days
The Asset		
v_H	102	High value of the asset
v_L	98	Low value of the asset
δ	0.5	Probability of the high value
The Traders		
μ	0.3	Probability of informed trading
η	1	Probability of a noise trader buying or selling
The AMMs		
N	2	Number of AMMs
λ_i^d	0.1	Learning rate
γ_i	0.00004	Exploration decay rate
c_i	0	Minimum exploration rate

illustration, and do not play a particular role. The two sets for ask and bid prices for the AMMs are chosen as if the AMMs know that the asset value in the previous day is 100 and might have gone up or down, but by not more than 3. It is of course important that these sets include the competitive equilibrium prices, since we are interested in understanding whether the prices set by the AMMs converge to these prices. Given our parameter values, if the market makers could choose in a continuous space, the competitive equilibrium bid and ask prices, as computed in expressions 7 and 8, would be 99.4 and 100.6, respectively. Because of the discrete space, with a tick size of 0.05, there are three equilibrium bid prices (99.3, 99.35, and 99.4), and there are three equilibrium ask prices (100.6, 100.65, and 100.7). The value of the exploration rate can be more easily interpreted by looking at the number of times the algorithm explores each possible price. As $D \to \infty$, the number of times the AMM is expected to explore is

$$\sum_{d=1}^{\infty} e^{-\gamma_i d} = \frac{e^{-\gamma_i}}{1 - e^{-\gamma_i}},$$

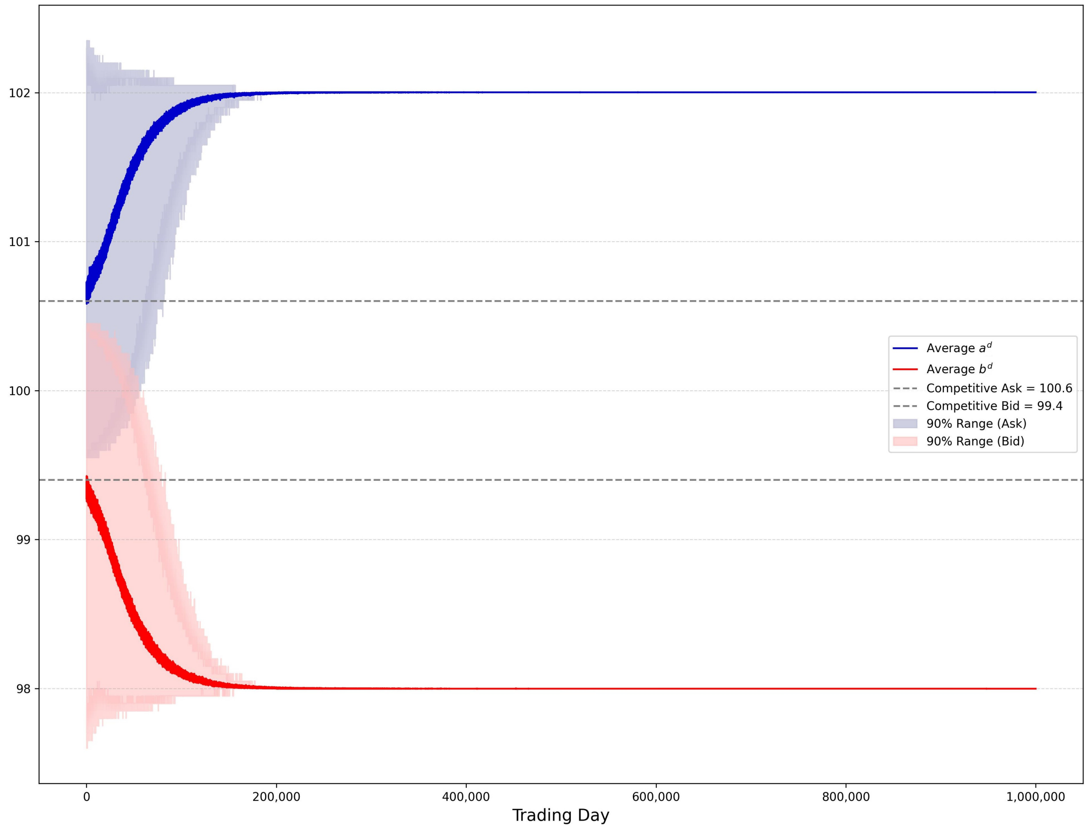
which, for $\gamma_i = 0.00004$, is equal to 25,000; this means that each price is randomly visited approximately 352 times. The learning rate of 0.1 means that the Q-value for a particular price is updated rather substantially whenever the price is chosen by the AMM. Finally, the initial Q-values are at least equal to 5, which is the maximum profit an AMM can get, e.g., by selling an asset worth 98 for 103. By doing this, all actions will be played initially as Q-values are revised downwards; thus, we avoid becoming stuck on certain prices early on or avoiding certain actions. This will play a role in our results, and we will discuss it in the next section.

6 Results for the Baseline Case

We now illustrate the results for the baseline case. Figure 1 shows the mean of the best ask prices and of the best bid prices (the market prices) across our 1,000 repetitions over the one million days.[6] The mean ask price tends towards v_H and, similarly, the mean bid price tends towards v_L. The convergence is reached after approximately 200,000 days. The graph also reports the 90% ranges for the best ask and bid prices across the simulations, in lighter blue and lighter red. In the first days, the ranges are large, as the probability of exploration is near one. As the exploration falls, the AMMs settle on ask prices around 102 and bid prices around 98. Figure 2 shows how the distribution of prices across the simulations change over time. After 250,000 days, all 1,000 repetitions have converged to ask prices weakly above v_H and bid prices weakly below v_L; in most cases, prices have converged to exactly v_H and v_L. By the final trading day, in almost all simulations (964 out of 1,000), the best ask submitted by the AMMs is 102 (v_H) and, in the remaining cases, the best ask is above 102. Similarly, in 974 simulations, the best bid is equal to 98 (v_L); in the remaining simulations, the best bids are strictly lower. When the market ask (bid) is weakly higher than v_H (respectively, lower than v_L), informed traders no longer have an incentive to trade. Profitable trade from the perspective of the AMMs occurs only because we have assumed that noise traders are completely inelastic. If we had assumed that noise traders also do not trade when the ask and bid prices reach the highest and lowest possible asset valuation, the quotes set by the AMMs would imply a market breakdown. Note also that, by setting the same price, the two AMMs get an equal share of the market. If an AMM undercut the other, they would achieve a much larger volume of trade and higher profits in expectation.

The reason the AMMs do not undercut one another is that the Q-values associated with prices in $(98, 102)$ are all strictly negative by the end of the one million days, as we show in Figure 3. Given that the probability of exploration is approximately zero in the final day, actions in the range $(98, 102)$ with negative Q-values are not picked as there exist actions for which the expected Q-value is, at a minimum, zero. This also occurs far before the final day.

[6] As we explained in Section 5, the word "day" is used in the model for exposition and should not be taken literally.



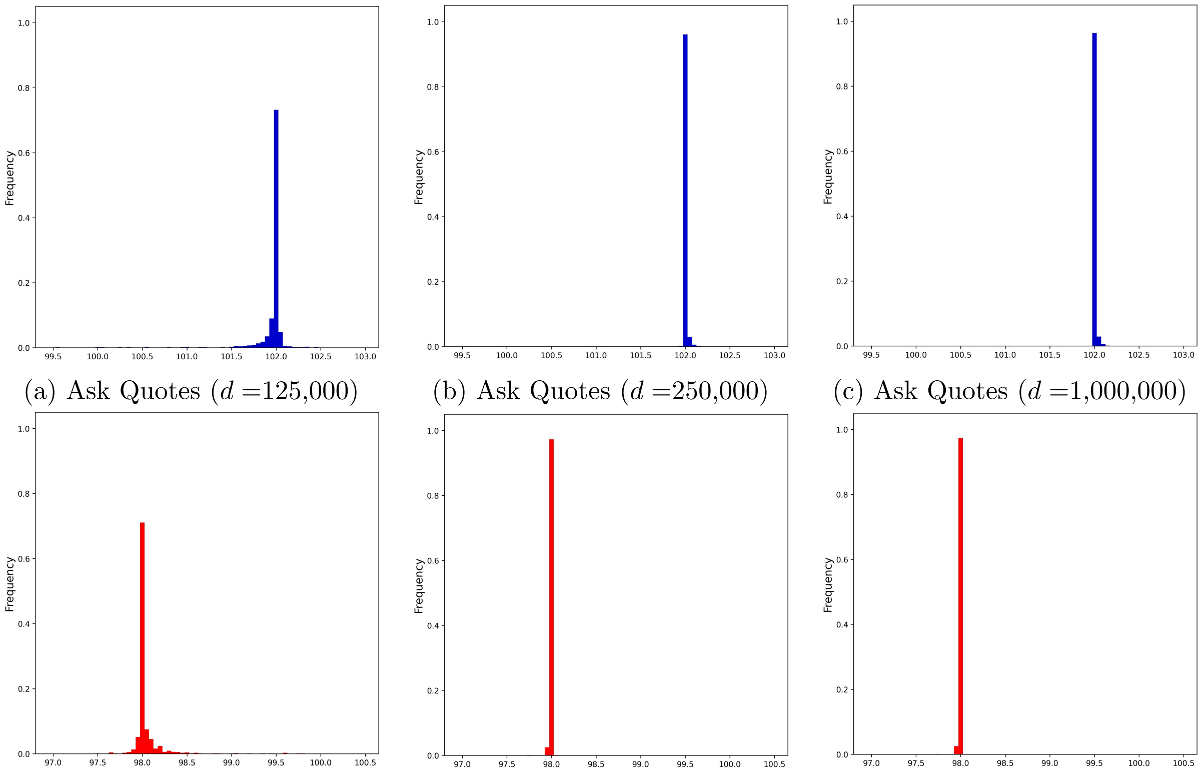
The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 102 and 98, respectively.

Figure 1: Baseline - Mean Market Quotes Over Days

When the probability of exploration becomes sufficiently low (e.g., ε_i^d is approximately 0.03% after 200,000 days), then almost all actions chosen are the greedy actions, which, as we have seen, are asks at least equal to v_H and bids not higher than v_L. Figure 3 also shows that the ask prices sufficiently above 102 and the bid prices sufficiently below 98 all have a Q-value of zero, suggesting some downward updating of asks above 102 and bids below 98. This occurs because, if one AMM has settled on an ask price of 102, then the other setting a price above this would earn a reward of zero.

6.1 Playing Against a Fixed-price Market Maker

The bid-ask spread set by the AMMs seems implausibly large, to the point of eliminating any incentive to trade on private information. Each AMM could increase its profits consistently by undercutting the other, but it does not learn to do so. We now ask whether this result is due to the strategic interaction between the two AMMs. Specifically, we ask whether a single AMM would at least be able to learn to undercut a “human” market maker who adopts the

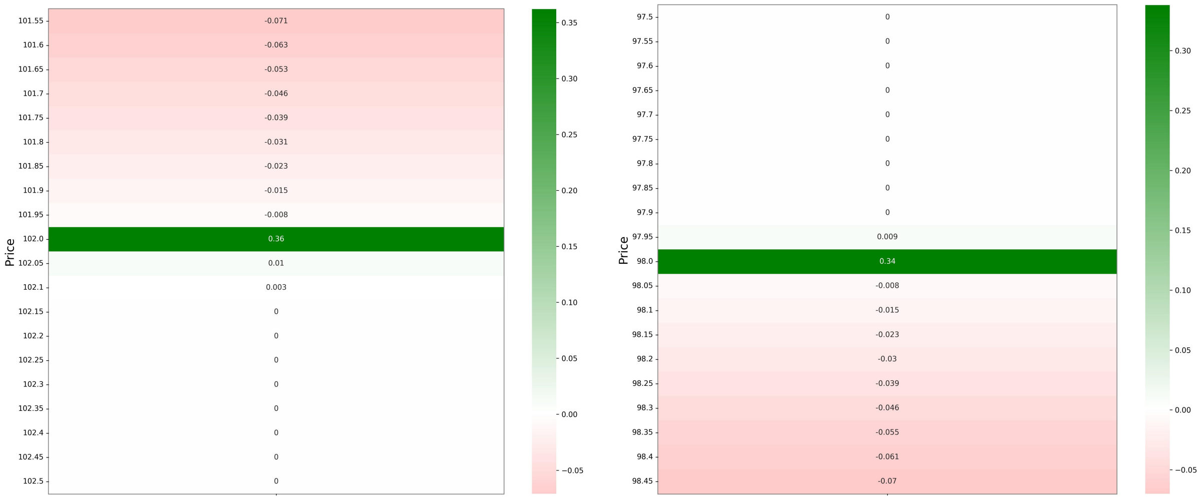


(a) Ask Quotes (d =125,000) (b) Ask Quotes (d =250,000) (c) Ask Quotes (d =1,000,000)

(d) Bid Quotes (d =125,000) (e) Bid Quotes (d =250,000) (f) Bid Quotes (d =1,000,000)

The graph plots the distributions of market ask and bid prices across the 1,000 repetitions at different points in the simulation: days 125,000, 250,000, and 1,000,000. The graph refers to the baseline Q-learning setup.

Figure 2: Baseline – Distributions of Market Ask and Bid Quotes over Time



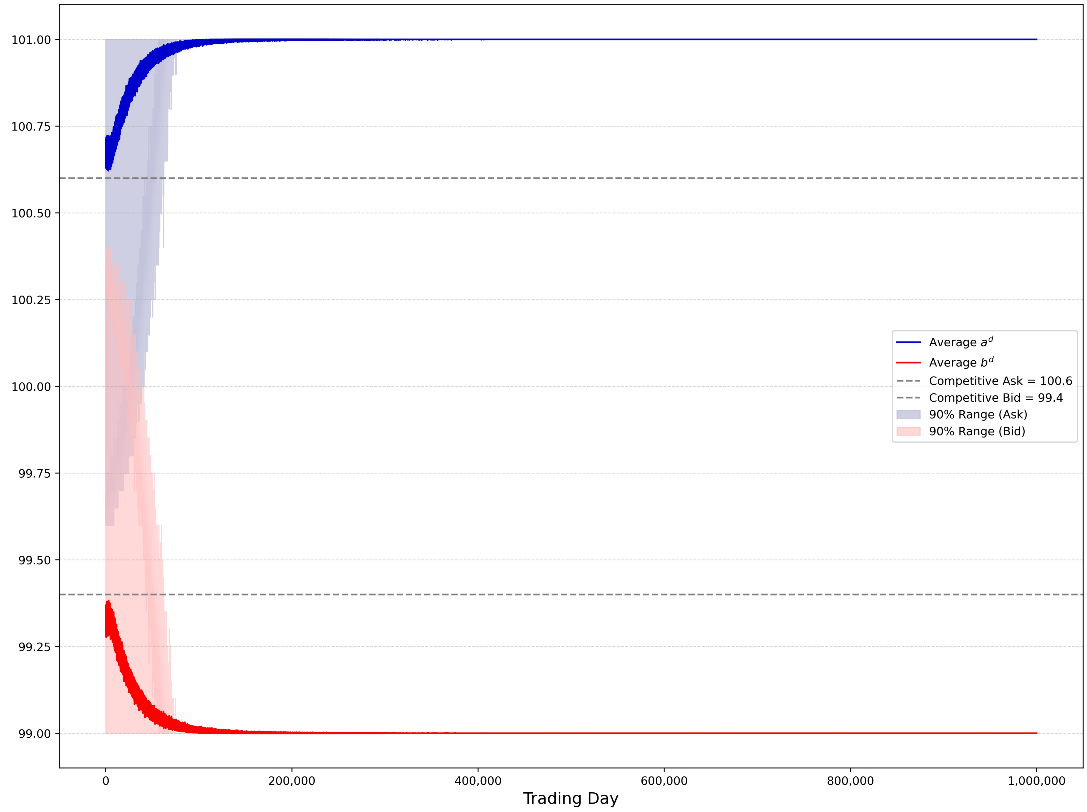
(a) Ask Q-Table (b) Bid Q-Table

The graph plot the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure 3: Baseline - Final Mean Q-Tables

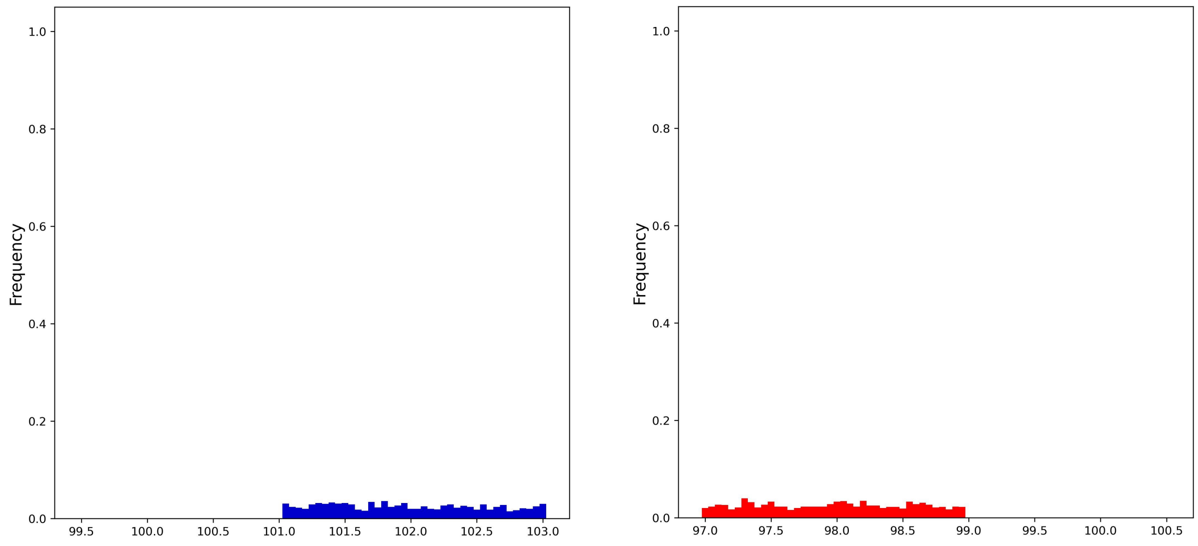
very simple strategy of setting a fixed ask lower than the asset's high value (v_H) (and higher than the competitive equilibrium ask) and a fixed bid higher than the asset's low value (v_L) (and lower than the competitive equilibrium bid). Specifically, we set a fixed ask price of 101 and a fixed bid price of 99; these prices are away from the competitive equilibrium prices, and also away from the possible asset value realizations.

Figure 4 is the analogue of Figure 1 for this case. The market quotes, after some initial experimentation, are the two fixed prices. Figure 5 shows that the AMM sets quotes that are strictly less competitive than the fixed-price market maker, with the distribution of ask quotes with support being strictly above 101 and that of bid quotes with support strictly below 99. Rather than undercutting these prices by one tick-size, which would be the best response, the AMM chooses less competitive quotes.



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 101 and 99, respectively.

Figure 4: Fixed-price Market Maker - Mean Market Quotes Over Days



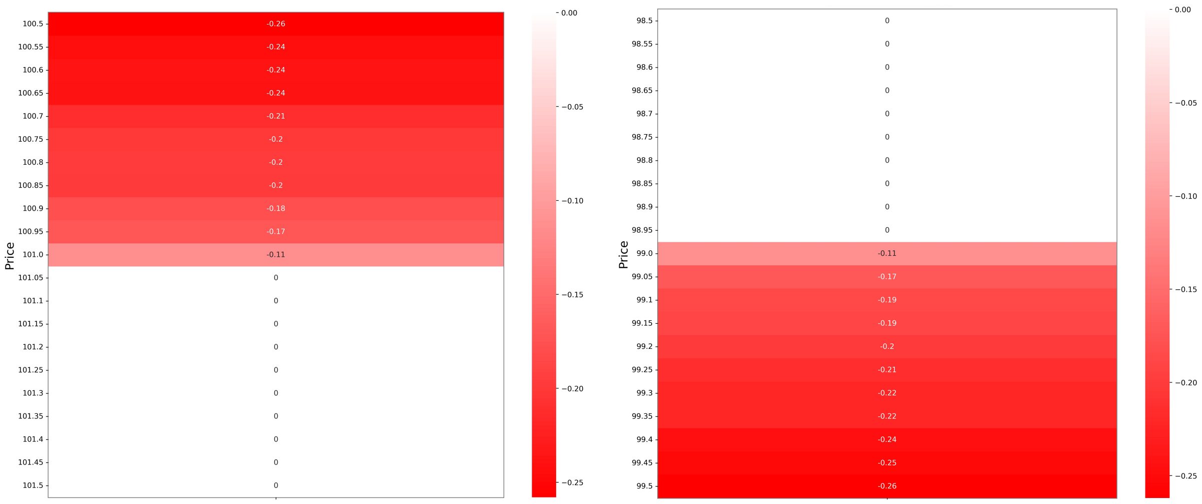
(a) Ask

(b) Bid

The graph shows the distribution of the ask and bid quotes set by the AMM in the final day, across the 1,000 repetitions. The graph refers to the baseline Q-learning setup.

Figure 5: Fixed-price Market Maker - Distribution of AMM Quotes in $d = D = 1,000,000$

Figure 6 shows the reason for this. All ask prices weakly lower than 101 are attached a negative Q-value; similarly, all bid prices weakly larger than 99 are attached a negative Q-value. Ask prices strictly larger than 101 and bid prices strictly lower than 99, instead, have a Q-value of 0, which is intuitive, since every time the AMM chooses them (by experimentation or exploitation) they do not trade and, hence, receive a payoff of zero.



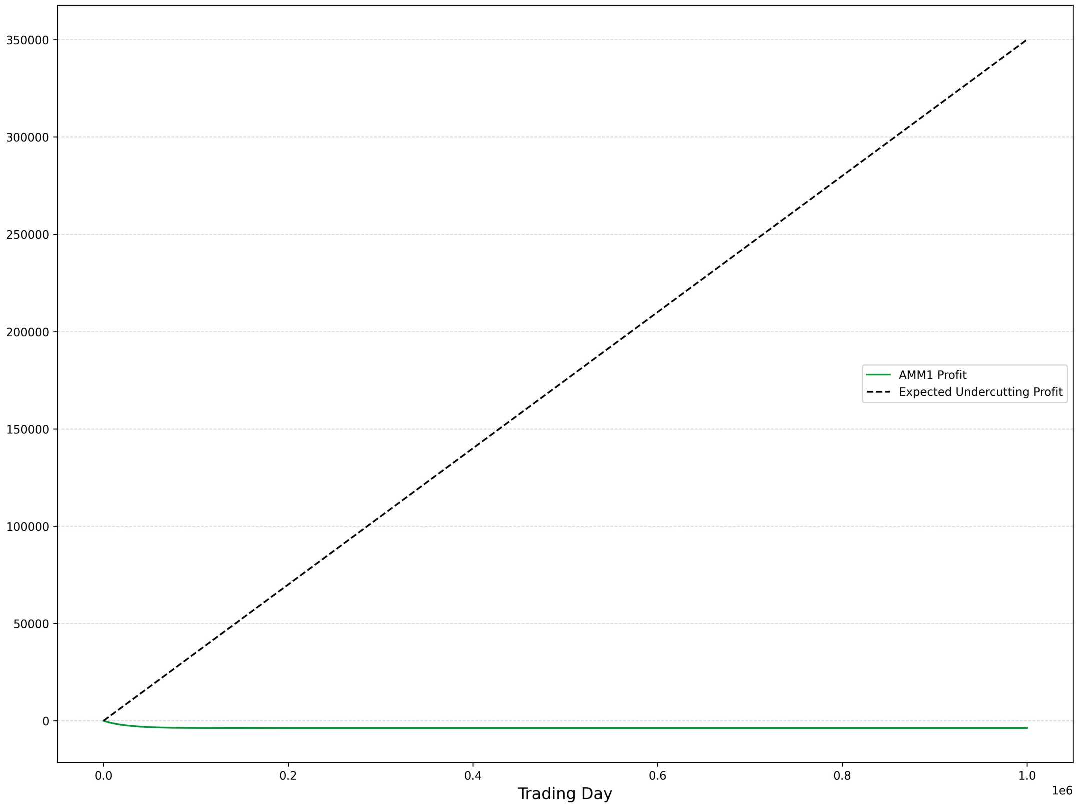
(a) Ask Q-Table

(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure 6: Fixed-price Market Maker - Final Mean Q-Tables

Overall, through simulations, we have obtained two unsettling results: first, AMMs using Q-learning do not learn to undercut, which implies an implausibly large bid-ask spread when they compete with each other; second, even more surprisingly, an AMM would not be able to outplay a human market maker who would simply set ask and bid prices at a fixed level, far from the competitive prices. By undercutting these fixed prices, the AMM would earn a large profit in expected value, but they are not able to do so. Figure 7 makes this point in a stark way by comparing the cumulative profits made by the AMM and those they would have made by undercutting the human market maker by one tick size. They make some initial losses in the first 100,000 days and then begin setting prices that are strictly less competitive; at this point, the AMM earns zero profits in each day.



The graph shows the mean of the cumulative profits across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The green line plots the cumulative profit between day 1 and day d, where the daily profit is computed as in Equation 3. The dashed line represents the expected cumulative profit from undercutting the fixed price market maker by one tick, i.e., continually setting an ask of 100.95 and a bid of 99.05.

Figure 7: Fixed-price Market Maker - Realized vs Undercutting Cumulative Profits

6.2 The Loss-free Pricing Result

While these simulation results are, at first glance, unsettling and surprising, they have a simple explanation. We now present a formal result stating that, in the absence of experimentation, the AMMs will eventually settle down on ask prices no smaller than v_H and on bid prices no larger than v_L whenever at least one ask above v_H and one bid below v_L have weakly positive initial Q-values. Note that this is a very minimal assumption given that, for very large asks and very low bids, an AMM is sure not to make a loss.

Proposition 1. *Suppose that $\lambda_i^d = \lambda$, and $\varepsilon_i^d = 0$ for every day d and every AMM$_i$. Moreover, suppose that, for every AMM$_i$, for at least one $\alpha_j \geq v_H$ and at least one $\beta_k \leq v_L$, the initial Q-values, $q_i^1(\alpha_j)$ and $\hat{q}_i^1(\beta_k)$, are weakly positive. Then, almost surely, there exists a $\overline{d} > 0$ such that, for every day $d > \overline{d}$, it holds that $a_i^d \geq v_H$ and $b_i^d \leq v_L$ for every AMM$_i$.*

Proof. We show the result for ask prices. The same reasoning can be used to show the result for bid prices.

First, observe that if, for $\alpha_j \geq v_H$, $q_i^1(\alpha_j) \geq 0$, then $q_i^d(\alpha_j) \geq 0$ for all d, since the payoff from quoting α_j (used in the updating of the Q-values) is never negative when $\alpha_j \geq v_H$.

Next, suppose, by contradiction, that there are infinitely many days d such that $a_i^d < v_H$ for at least one AMM$_i$. Define α^* as follows:

$$\alpha^* = \arg\min(\alpha_k \text{ such that } a_i^d = \alpha_k \text{ for infinitely many } d \text{ for at least one AMM}_i).$$

Given our hypothesis (by contradiction), $\alpha^* < v_H$.

Now, observe that the Q-value $q_i^d(\alpha^*)$ is no larger than the maximum of the initial Q-value $q_i^1(\alpha^*)$ and of $\alpha^* - v_L$, since the latter is the largest payoff that can be obtained with the ask price α^*, no matter what d is.

Next, observe that, for d large enough, whenever the ask price α^* is chosen by AMM$_i$, there is a positive probability that AMM$_i$ trades (because this is best ask price proposed by all AMMs) and that the resulting payoff is $\alpha^* - v_H$ (whenever the asset has high value).

Given that $\alpha^* - v_H < 0$ and that $\lambda_i^d = \lambda$ for all d, there must be exist k^* such that

$$(1-\lambda)^{k^*} \max(q_i^1(\alpha^*), \alpha^* - v_L) + (1 - (1-\lambda)^{k^*})(\alpha^* - v_H) < 0.$$

After k^* consecutive such draws, which eventually must occur with probability 1, we have that, almost surely, $q_i^{\overline{d}}(\alpha^*) < 0$ for some finite $\overline{d}$.

Given that $q_i^d(\alpha_j) \geq 0$ for all d, the inequality holds in particular for all $d \geq \overline{d}$. Therefore, it cannot be that $a_i^d = \alpha^*$ for any $d \geq \overline{d}$, since for $d = d^*$, AMM$_i$ cannot choose α^* as $q_i^{d^*}(\alpha_j) \geq 0 > q_i^{d^*}(\alpha^*)$. For $d > d^*$, AMM$_i$ cannot choose α^* as the updating rule ensures that, for all $d > d^*$, $q_i^d(\alpha_j) \geq 0 > q_i^d(\alpha^*) = q_i^{d^*}(\alpha^*)$. This leads to the contradiction that α^* is not chosen by AMM$_i$ infinitely many times. Q. E. D. □

The logic of the proof is simple. For any ask price α^* strictly lower than v_H, there is a positive probability that the market maker choosing it sells at this price (if this market maker has the lowest ask price) and the value of the asset is high, resulting in a strictly negative payoff $\alpha^* - v_H$. After enough consecutive such events, which occur with probability 1 in the infinite sequence of days, this would lead $q_i^{d^*}(\alpha^*)$ to become negative, which, in turn, would lead this market maker to never consider this ask price again when there is no experimentation (because the Q-values for at least one $\alpha_j \geq v_H$ are always weakly positive).[7]

Our result is related to a *maxmin* result obtained by Sarin and Vahid (1999) in an individual learning environment in which different alternatives yield stochastic rewards. Considering a learning model like ours (with a constant weight on the realized immediate reward in the updating of Q-values), they show that, eventually, the decision maker chooses an alternative with the highest minimum payoff among the chosen alternatives. Note that, in our model with one AMM and one market maker setting a fixed price, the result by Sarin and Vahid (1999) indeed provides the insight that the AMM would never learn to undercut (because the undercutting would not be best from this *maxmin* perspective), resulting in all trade being performed by the fixed-price market maker. We now state this result formally:

Proposition 2. *Consider the case of one AMM (denoted AMM_1) that competes with a market maker choosing a fixed ask price $\overline{a}^d = \overline{\alpha} < v_H$ and a fixed bid price $\overline{b}^d = \overline{\beta} > v_L$ for all days d. Suppose that $\lambda_1^d = \lambda$, $\varepsilon_1^d = 0$ for all days d, and that the initial Q-values $q_1^1(\alpha_j)$ and $\hat{q}_1^1(\beta_k)$ are weakly positive for at least one ask price $\alpha_j \geq v_H$ and at least one bid price $\beta_k \leq v_L$. Then, almost surely, there exists some $\overline{d} > 0$ such that, for all days $d > \overline{d}$, $a_1^d > \overline{\alpha}$ and $b_1^d < \overline{\beta}$. Asymptotically, almost surely, the AMM will not trade.*

Proof. First, note that the minimum payoff for all asks $\alpha_j > \overline{\alpha}$ and all bids $\beta_k < \overline{\beta}$ is zero. In contrast, all asks weakly lower than $\overline{\alpha}$ and all bids weakly higher than $\overline{\beta}$ have a negative minimum payoff (in the case of equality, there is a positive probability the AMM will buy an asset with value v_L or sell an asset with value v_H). Then, observe that Proposition 1 and Corollary 1 of Sarin and Vahid (1999) immediately imply that asymptotically, for $d > \overline{d}$, almost surely, $a_1^d > \overline{\alpha}$ and $b_1^d < \overline{\beta}$. Given these ask and bid prices, asymptotically, almost surely, any trade occurs with the market maker setting a fixed price. Q. E. D. □

[7] As we explained in Footnote 4, one could imagine a variant in which, for example, the Q-value for the quoted ask is only updated when there is a buy order. Proposition 1 would also hold with this variant of Q-learning, and so would all other propositions we present in the paper (in fact, the proofs are easier in the variant). Similarly, the simulation results for the variant are very similar, and we present them in Appendix B.7.

In our simulation with a fixed-price market maker, we still observe this result even if we allow for some exponentially decaying experimentation.[8]

In our Proposition 1, we cannot use the result of Sarin and Vahid (1999) because of the interaction among N learning players. But, similarly to Sarin and Vahid (1999), we obtain that algorithmic learning leads the market makers to use loss-free pricing strategies in the absence of experimentation. Our simulation with exponentially decaying experimentation leads to observations very much in line with those obtained theoretically without experimentation.

It is worth noting that, while Proposition 1 rules out ask and bid prices in (v_L, v_H) in the long run, in the final days of the simulations, we mostly observe ask prices close to v_H and bid prices close to v_L. All ask (bid) prices weakly above v_H (respectively, below v_L) have the *maxmin* property, hence they can all arise according to the logic used to establish our result. Nevertheless, it would be possible to refine our result and show that, most of the time, AMMs choose v_H (respectively, v_L) as the ask (respectively, bid) price, while some stochasticity persists in the limit. The reason for the stochasticity is simple. To see it, let us focus on the ask prices, and suppose that there exists a $\overline{d}$ such that the ask price $a^d = \min\{a_1^d, ..., a_N^d\}$ is equal to α_j for all $d > \overline{d}$. Clearly, given Proposition 1, $\alpha_j \geq v_H$; let us consider, in particular, $\alpha_j > v_H$.[9] Any AMM$_i$ who does not sell (either because they have chosen an ask strictly higher than α_j or because they have chosen α_j but, by randomness, another AMM with the same ask is selected to trade) keeps receiving a payoff of zero. Therefore, their $q_i^d(\alpha_j)$ becomes closer to zero. After a sufficiently long sequence of such events, $q_i^d(\alpha_j)$ becomes smaller than the Q-value for another ask price, say α_l ($l \neq j$). AMM$_i$ then switches to α_l. Finally, note that if $\alpha_l > \alpha_j$, AMM$_i$ continues not to trade, hence also $q_i^d(\alpha_l)$ becomes closer to zero. All Q-values for ask prices weakly larger than α_j will eventually become smaller than the Q-value for some smaller ask price, which contradicts that $a^d = \alpha_j$ for all $d > \overline{d}$. While this argument shows why Q-learning cannot converge to a single price, it also unveils why, in most cases, the ask price is exactly equal to v_H. When an AMM chooses a higher ask price, they receive a zero payoff, and the logic just illustrated applies. Intuitively, among the ask prices weakly larger than v_H (the only ones not ruled out by Proposition 1), the ask price $a_i^d = v_H$ is the most frequently chosen,

[8] Sarin and Vahid (1999) obtain their result by making an assumption of "optimism" for all initial Q-values (but see their footnote 10). In the proposition, we make the weaker assumption that at least one initial Q-value has to be weakly positive, which is very natural in our context. In the simulations, we assumed that all initial Q-values are "optimistic". In other simulations, we use lower, but positive, initial Q-values, but this does not alter the results (see Appendix B.1).

[9] In our argument, we are assuming that the grid of ask prices includes v_H.

asymptotically, because it is the price at which it is more likely that an AMM will sell. It is not the ask price forever because $a^d = \min\{a_1^d, ..., a_N^d\}$ and at least one AMM will eventually (and temporarily) switch to a lower ask (or all market makers will switch to a larger ask). Note that this argument is different from the usual undercutting argument, in which the comparison between the payoffs of two prices leads to choosing the lower one. In Q-learning, for an AMM, a higher ask price leads to no trade and, hence, to a decreasing Q-value, which eventually leads the AMM to switch from that ask.[10]

7 Beyond the Baseline Case

The theoretical loss-free pricing result that we stated in the previous section is obtained in the absence of experimentation and asymptotically, as the number of days converges to infinity. Our simulation results for the baseline case, on the other hand, show that the same result occurs even in the presence of experimentation and in finite time. It is worth studying what happens when we consider alternative parameterizations to the baseline case. One possibility is to let AMMs react to market feedback in a different way: this means that they update the Q-values on the basis of a different value of λ_i^d (set equal to 0.1 in the baseline case); another is to let them have different exploration dynamics, due to different values of the parameters γ_i and c_i (set equal to 0.00004 and 0, respectively, in the baseline case), or to a different functional form of ε_i^d. Here, we only report the main results for these alternative specifications in terms of the mean market prices over days, and refer the reader to Appendix B for more details.

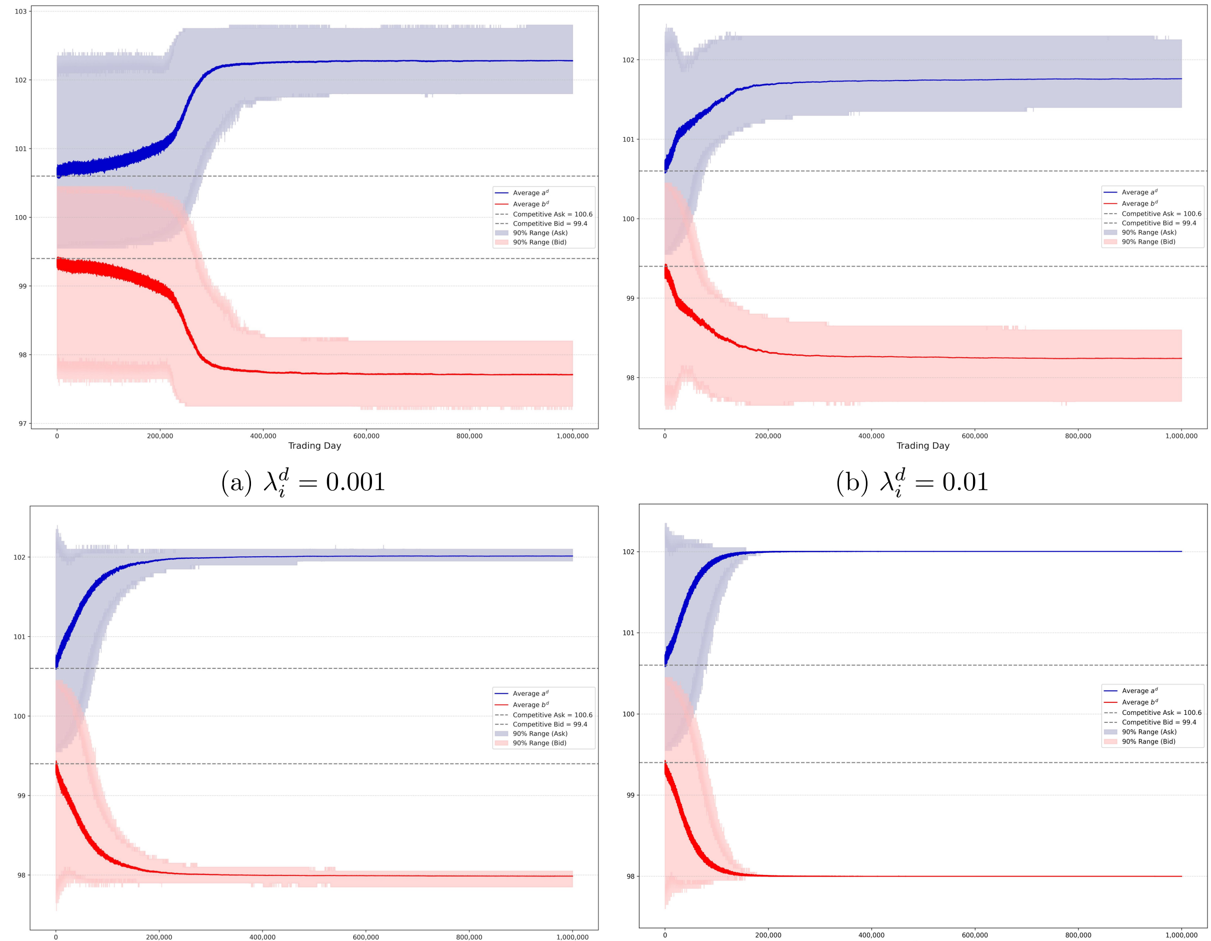
7.1 Varying the Learning Rate

As a first exercise, we set $\lambda_i^d = 0.01$ for both AMMs (i.e., $i = 1, 2$), leaving all other parameters unchanged from the baseline case. AMMs now give much less weight to any feedback they receive from the market. The resulting ask and bid prices are shown in Figure 8b, the analog of Figure 1. The final mean ask is 101.76, lower than v_H (102), although not by much, and still substantially far from the competitive equilibrium ask. Similarly, the final mean bid is 98.24,

[10] As a final note, it is worth observing that this logic does not hold in the case in which there is only one AMM competing with a market maker setting fixed ask and bid prices. In this case, there is no reason to expect that the AMM will mainly choose an ask close to v_H. On the contrary, the Q-values for all the ask prices above v_H will eventually reach the same level and all these ask prices will keep being chosen. Accordingly, our simulations show that, in the last of trading, the distribution of all ask prices chosen by the AMM as the greedy action is approximately uniform, above v_H (see Figure 5).

higher than v_L (98), but still far from the competitive bid.

In the other panels of Figure 8, we consider alternative λ_i values. We find that $\lambda_i^d = 0.05$ produces a very similar result to the baseline of $\lambda_i^d = 0.1$ (reported in Panel d). Interestingly, a similar result is also observed for $\lambda_i^d = 0.001$ (Figure 8a), in this case due to insufficient learning.



(a) $\lambda_i^d = 0.001$

(b) $\lambda_i^d = 0.01$

(c) $\lambda_i^d = 0.05$

(d) $\lambda_i^d = 0.1(*)$

Each panel, for a given value of λ_i^d, plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The "(*)" refers to the baseline parameterization that we considered in Section 6. The mean market ask (bid) quotes in the final day are as follows: $\lambda_i^d = 0.001$ — 102.28 (97.71); $\lambda_i^d = 0.01$ — 101.76 (98.24); $\lambda_i^d = 0.05$ — 102.01 (97.99); $\lambda_i^d = 0.1$ — 102 (98).

Figure 8: λ_i^d Comparative Statics - Mean Market Quotes Over Days

To understand these results, it is worth going back to the intuitions for Proposition 1. The loss-free pricing result is obtained because, in infinite horizon, with probability 1, an ask strictly lower than v_H is associated with a sufficiently large number of negative payoffs that its Q-value becomes negative, and, thus, it can no longer be the greedy action. In the absence of experimentation, that ask is never chosen again. With relatively large λ_i^d, as in our baseline simulation, just a few negative payoffs are enough to make the Q-value of an ask price below v_H

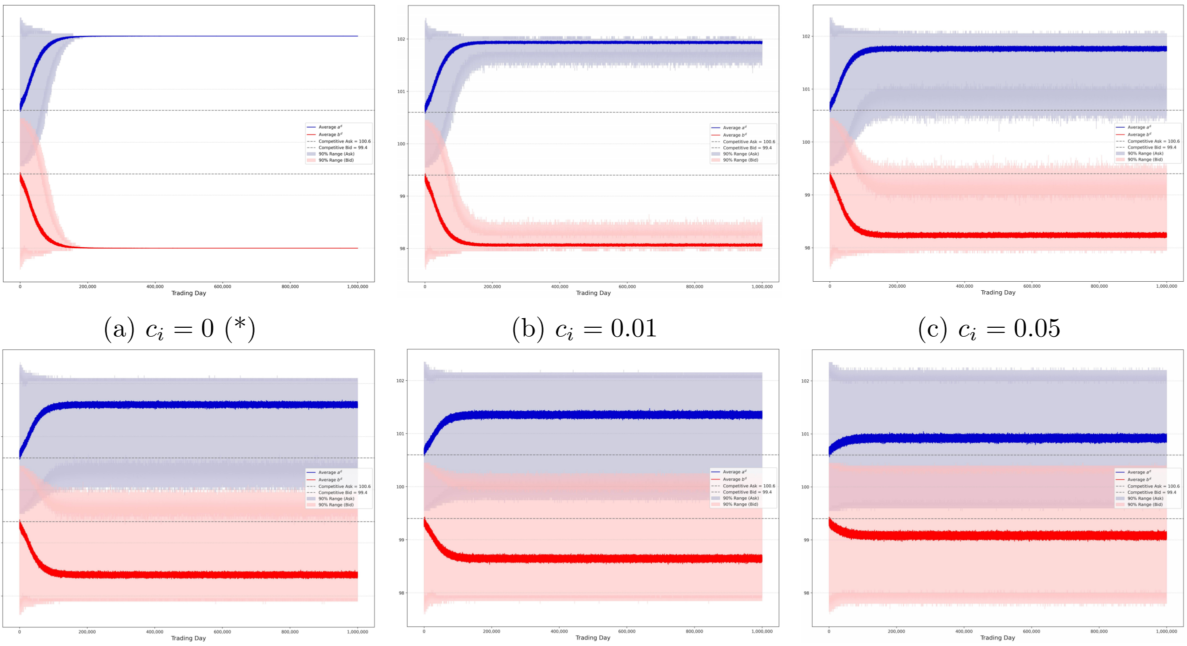
negative. When $\lambda_i^d = 0.01$ rather than 0.1, the number of negative experiences needed to bring the Q-value below zero becomes much larger, since each experience is weighted much less in the updating process.[11] As our simulations reveal, we were able to move away from the loss-free pricing result with this lower λ_i^d.[12] When λ_i^d is too small, however, the algorithm does not react strong enough to market feedback and the baseline result occurs again.

7.2 Varying the Minimum Exploration Probability

Rather than changing the learning rate, λ_i^d, let us now see what happens if we change the frequency with which the AMMs explore rather than exploit. As a first exercise, we set $c_i = 0.05$, which means that, in each day, there is at least a 5% probability of exploration. As Figure 9c shows, the mean ask is below the loss-free price, although still far from the competitive price. The exploration helps to have a lower ask, although the change is not very large. Part of this may look mechanical, since prices (including lower ask prices) are chosen by exploration without being optimal. But there is a more important mechanism at work: intuitively, without exploration, once the Q-value for an ask price above the competitive equilibrium becomes negative, that price will not be used again (since it cannot be the greedy action) and the AMM is unable to learn that it is profitable (in expectation). With a constant experimentation probability, the price will be used again and, eventually, the Q-value can become positive. With a sequence of one million days, AMMs learn that at least some prices are profitable, although not those too close to the competitive equilibrium. Indeed, for asks and bids closer to the competitive equilibrium, when their Q-value is negative, it requires that they are chosen more often through random exploration for the Q-values to become positive again (since the profit from these prices is smaller than for more extreme prices). Another notable feature in Figure 9 is that even, at the end of one million days, there is a great deal of heterogeneity in prices, as shown by the thicker mean price lines (greater variation) and wider ranges across simulations for both ask and bid when $c_i > 0$. This is the cost of experimentation, since all prices keep being tried.

[11] For example, suppose that the Q-value for an ask price of 101 is 0.2: when $\lambda_i^d = 0.1$, it takes two losses (the loss in our given example is -1) to reach a negative Q-value; in contrast, for $\lambda_i^d = 0.01$, the number of required losses is 19.

[12] In other words, across the simulations, there are sequences of one million days in which the Q-values of ask prices below v_H are positive in the last days.



(a) $c_i = 0$ (*)

(b) $c_i = 0.01$

(c) $c_i = 0.05$

(d) $c_i = 0.10$

(e) $c_i = 0.20$

(f) $c_i = 0.50$

Each panel, for a given value of c_i, plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The "(*)" refers to the baseline parameterization that we considered in Section 6. The mean market ask (bid) quotes in the final day are as follows: $c_i = 0$ — 102 (98); $c_i = 0.05$ — 101.94 (98.06); $c_i = 0.10$ — 101.77 (98.24); $c_i = 0.20$ — 101.37 (98.65); $c_i = 0.50$ — 100.92 (99.07).

Figure 9: c_i Comparative Statics - Mean Market Quotes Over Days

An alternative to considering different lower bounds on the level of exploration (c_i) is to consider different values for the exploration decay parameter, γ_i. We have done this exercise for different values of γ_i:

$$\gamma_i = \{1 \times 10^{-4}, \quad 4 \times 10^{-5}, \quad 2 \times 10^{-5}, \quad 1 \times 10^{-5}, \quad 5 \times 10^{-6}, \quad 2.5 \times 10^{-6}\}.$$

Broadly speaking, these different parameter values do not produce significantly different results. We refer the reader to Appendix B.4 for all the details.

7.3 Soft-max Exploration

The ε-greedy exploration is only one of the possible exploration strategies considered in machine learning. Another common exploration method is the *logit exploration*, also known in the machine learning literature as "Boltzmann exploration" or "soft-max exploration" (see, e.g., Sutton and Barto, 2018).[13]

Rather than having the best action chosen with probability $1 - \varepsilon_i^d$ (in case of exploitation) and all actions chosen uniformly with probability ε_i^d (in case of experimentation), with logit exploration more promising actions are explored with higher probabilities. In our context, this means that AMMs choose asks and bids with higher Q-values with a higher probability. Specifically, on day d, AMM$_i$ chooses the ask α_j with the following probability:

$$\Pr(a_i^d = \alpha_j) = \frac{e^{\tau Q_i(\alpha_j)}}{\sum_{\alpha_l \in \mathbf{A}} e^{\tau Q_i(\alpha_l)}},$$

and, similarly, chooses the bid β_j with the following probability:

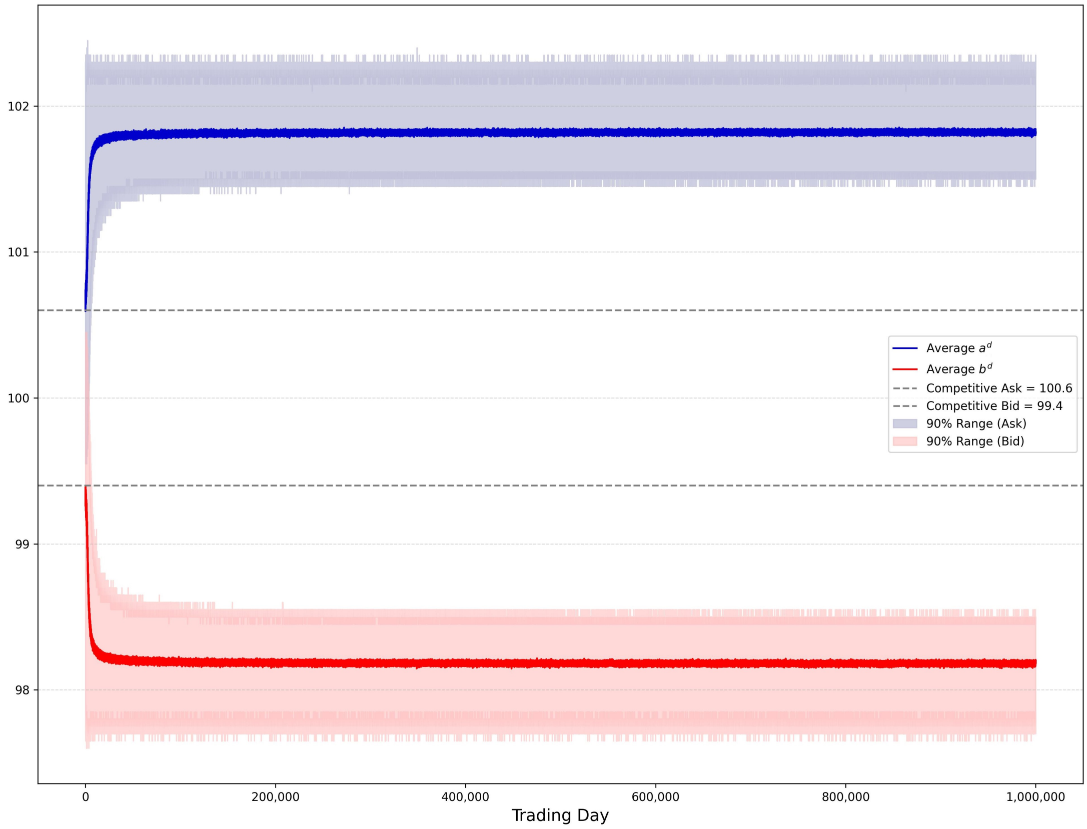
$$\Pr(a_i^d = \beta_j) = \frac{e^{\tau Q_i(\beta_j)}}{\sum_{\beta_l \in \mathbf{B}} e^{\tau Q_i(\beta_l)}},$$

where the parameter τ controls the extent of randomness: when $\tau = 0$, then this method chooses all asks (bids) uniformly, and when τ goes to ∞, then this method chooses the greedy ask (bid) with probability 1. For intermediate values, this method offers a smoother way of choosing different actions than ε-greedy exploration. The soft-max exploration functions differently from the ε-greedy in that zero is no longer a "cut-off" point for the Q-values. Previously, when a Q-value dropped below zero and exploration was near zero, this action would not be chosen again, even if the chosen alternative was near in value, but just positive. Now, if two Q-values are close but one is slightly positive and the other slightly negative, then the probabilities of these actions being chosen are quite similar. As a result, actions with potential losses can still be chosen through the soft-max exploration.

We now present some results for the illustrative case of $\tau = 20$. In the final day, we see (Figure 10) that the mean ask and bid prices are 101.82 and 98.20, within the $(98, 102)$ interval but still away from equilibrium prices.

In contrast to the Q-tables displayed in Figure 3, we see that the AMMs attach positive Q-values to ask (bid) prices below v_H (respectively, below v_L) even in the last trading day (see Appendix B.5).

[13] Logit exploration has also been used in learning models in Game Theory, see, e.g., Mookherjee and Sopher (1997) and Camerer and Ho (1999).



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 101.82 and 98.20, respectively.

Figure 10: Soft-max Exploration - Mean Market Quotes Over Days

8 Counterfactual Updating

In the previous section, we have studied how changing the learning and exploration parameters affect the results of the baseline case. We now proceed in a different way. We note that, in a standard theoretical analysis of financial markets, we typically assume that the model is known by all players (in fact, even common knowledge among them). The advantage of Q-learning pricing algorithms is that we can dispense with such an assumption: in our stateless Q-learning, the AMMs only know the range in which to find the best quotes, their own payoffs, and nothing else. While a complete knowledge of the model is a heroic assumption, we can, however, endow AMMs with at least an elementary understanding of a financial market. It seems plausible that a market maker should understand that a trader willing to buy the asset at a specific ask price would also buy it at any lower price. Similarly, they should understand that a trader willing to sell the asset at a specific bid price would also be willing to sell it at any greater price. We follow this line of reasoning and let AMMs update not only the Q-values for the quotes they chose, but also for the other quotes for which an obvious inference could be made. In addition to this simple understanding of how markets work, we endow the AMMs with minimal information, that of the market quotes (i.e., the best ask and the best bid), an uncontroversial hypothesis for any market maker model.[14]

As an example, consider the case of two AMMs, and suppose a trader bought the asset from AMM$_1$ at the price of $a^d = \alpha_j$. AMM$_1$ knows that the trader would have also bought had he chosen any other price lower than α_j. Hence, in our new algorithm, they also update the Q-values for all ask prices lower than α_j. They do not update ask prices greater than α_j since they do not know for sure whether they would have been able to sell at those prices. What about AMM$_2$ who had chosen an ask greater than α_j? Observing the trade at $a^d = \alpha_j$, they know that they could have undercut their competitor, while they would not have sold for any ask greater than α_j. For an ask price of α_j, they would have sold with probability 50%. Therefore, AMM$_2$ updates the Q-values for all asks: for any ask strictly greater than α_j the Q-values are updated with a payoff of zero; for any ask weakly lower than α_j, the the Q-value are updated taking into account that the AMM would have sold (with probability 50% in the case of a tie at α_j).

One could observe that a realized profit may be more relevant than a hypothetical one (in the counterfactual case) in the updating of the Q-values. For instance, the AMM could be unsure of whether in the case they had chosen a lower ask, so would have done the other AMM. We can accommodate this variant by putting a weight ρ ($0 \leq \rho \leq 1$) on the counterfactual payoff.

Let us now illustrate the general case of N AMMs in detail. The following possibilities can occur:

1. **AMM$_i$ quotes the best ask $\boldsymbol{a_i^d = \alpha_j = a^d}$ and sells the asset**. AMM$_i$ updates the Q-values for α_j, the action chosen, on the basis of the *realized* payoff, and the Q-values for all asks lower than α_j using the payoffs they would have received by selling at those prices (the *counterfactual* payoff). Since we have assumed that AMMs only know their own quotes and the market quotes, AMM$_i$ does not update the Q-value for any ask greater than α_j, since they do not know whether they would have been able to sell.

[14] We could assume that AMMs are also aware of their competitors' ask and bid quotes (e.g., through the order book), but we do not use this assumption here.

Formally, the Q-values are updated as follows:

$$q_i^{d+1}(\alpha_j) = \lambda_i^d(\alpha_j - v^d) + (1 - \lambda_i^d)q_i^d(\alpha_j), \tag{16}$$

$$q_i^{d+1}(\alpha_l) = \lambda_i^d \rho(\alpha_l - v^d) + (1 - \lambda_i^d)q_i^d(\alpha_l), \qquad \text{for all } \alpha_l < \alpha_j, \tag{17}$$

$$q_i^{d+1}(\alpha_k) = q_i^d(\alpha_k), \qquad \text{for all } \alpha_k > \alpha_j. \tag{18}$$

2. **AMM$_i$ quotes the ask $\boldsymbol{a_i^d \geq \alpha_j = a^d}$ and another AMM sells the asset**. Since AMM$_i$ is aware of their own ask price and of the market ask at which the transaction occurred, they update the Q-values for all asks as follows: for any ask *strictly* greater than α_j, the Q-value is updated with a payoff of zero; if AMM$_i$ quotes $a_i^d > \alpha_j$, then the Q-value associated with α_j is updated with the *counterfactual* payoff (i.e., taking into account that the AMM would have sold with a probability of $\frac{1}{n+1}$, where n is the number of AMMs choosing the market ask);[15] if, instead, AMM$_i$ quotes $a_i^d = \alpha_j$ but does not trade, then they update with the *realized* profit (i.e., zero) rather than the counterfactual profit; and, for ask prices strictly lower than α_j, the Q-values are updated with the counterfactual payoff (taking into account that the AMM would have sold with probability 1).

$$q_i^{d+1}(\alpha_k) = (1 - \lambda_i^d)q_i^d(\alpha_k), \qquad \text{for all } \alpha_k > \alpha_j, \tag{19}$$

$$q_i^{d+1}(\alpha_j) = \lambda_i^d \rho \frac{1}{n+1}(\alpha_j - v^d) + (1 - \lambda_i^d)q_i^d(\alpha_j), \qquad \text{if } a_i^d > \alpha_j, \tag{20}$$

$$q_i^{d+1}(\alpha_j) = (1 - \lambda_i^d)q_i^d(\alpha_j), \qquad \text{if } a_i^d = \alpha_j, \tag{21}$$

$$q_i^{d+1}(\alpha_l) = \lambda_i^d \rho(\alpha_l - v^d) + (1 - \lambda_i^d)q_i^d(\alpha_l), \qquad \text{for all } \alpha_l < \alpha_j. \tag{22}$$

3. **AMM$_i$ quotes the ask $\boldsymbol{a_i^d \geq \alpha_j = a^d}$ and there is no buy**. Note that this can occur either because there is no trade or because there is a sell order. AMM$_i$ updates the Q-values for all prices that are greater than or equal to α_j since, at all these prices, the payoff is, or would have been, zero. AMM$_i$ does not update the Q-values for the lower asks, since they do not know whether setting a lower price would have led to a sell.

$$q_i^{d+1}(\alpha_k) = (1 - \lambda_i^d)q_i^d(\alpha_k), \qquad \text{for all } \alpha_k \geq \alpha_j, \tag{23}$$

$$q_i^{d+1}(\alpha_l) = q_i^d(\alpha_l), \qquad \text{for all } \alpha_l < \alpha_j. \tag{24}$$

[15] Here we are assuming that the AMMs know the number of other AMMs quoting the best prices. In the case of only one AMM quoting at the best ask, this probability would be 50%. Alternative assumptions would not alter the analysis significantly.

For the bid, the cases are analogous; just note that higher bids are preferred by the traders, since they are the prices at which they sell to the market maker. We present the formulas in Appendix A.

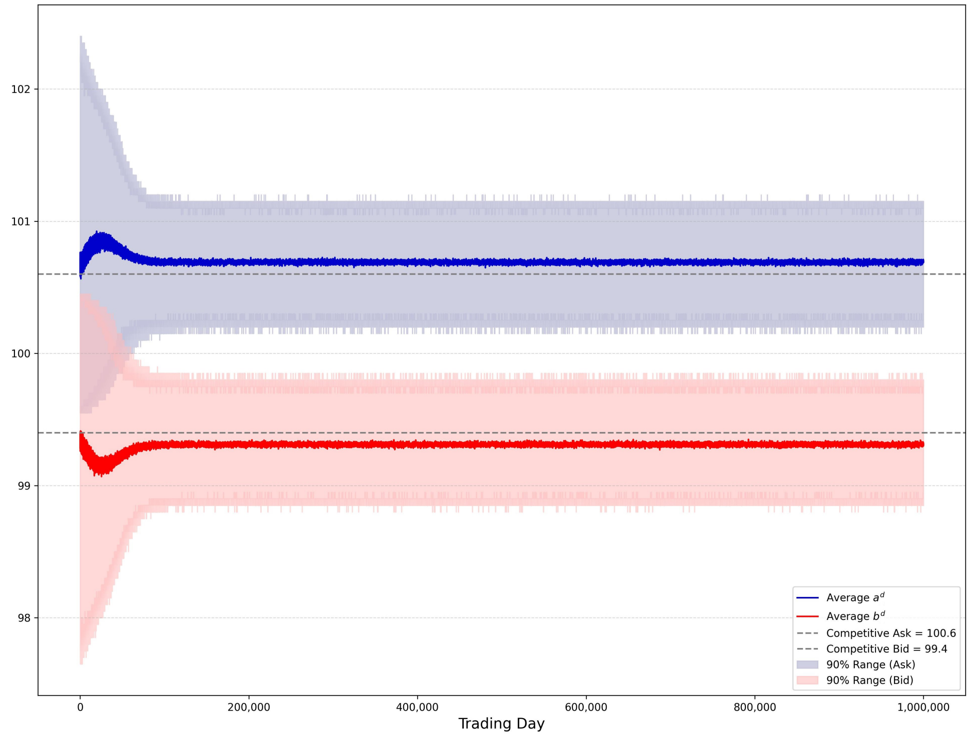
We are not the first to suggest that a model of reinforcement learning can (and, perhaps, should) incorporate some understanding of the economy or of the game that is being played. Models of learning with counterfactual updating have, indeed, been studied in Game Theory. Camerer and Ho (1999) propose a model of "Experience-Weighted Attraction" (EWA) learning which encompasses both reinforcement learning and belief learning (fictitious play). Our learning model with counterfactual updating is a special case of the EWA learning, and so is the individual decision making model of Sarin and Vahid (1999) previously discussed. Note that also Camerer and Ho (1999) allow for the counterfactual updating to have a lower weight (captured by their parameter δ). More recently, counterfactual updating has been studied by Asker et al. (2024) in their work on algorithmic pricing in an oligopoly. They observe that a simple reinforcement learning model like our baseline Q-learning and a model of learning in which the algorithm has a lot of knowledge and information about the economy (e.g., they know the competitors' prices) are two extreme models; in between, there are others which only use some understanding of the economy and some amount of information. Our model in which AMMs only know and use the market quotes and their own quotes for updating the Q-values is similar to their Imperfect Counterfactual Updating.[16]

9 Results for Counterfactual Updating

Figure 11 shows the mean market ask and bid prices over days for the Q-learning with counterfactual updating. Prices converge towards the competitive equilibrium. In the final days, in all simulations, ask and bid are either at the competitive levels or very close to them. The figure refers to simulations in which the parameter values are identical to those listed in Table 1 (and, for exposition, we have set $\rho = 0.6$).

Figure 11 shows that the market quotes very quickly reach the competitive levels; in particular, the ask and bid quotes in the final day are 100.74 and 99.26, respectively. These prices are reached by $d = 100{,}000$ and the average quotes remain around these levels for the remainder

[16] In Asker et al. (2024) the knowledge of the downward-sloping demand curve allows updating more than just the chosen price (in what they call the "synchronous learning"). As we wrote, we are assuming that the AMMs are only aware of their own prices and of the market prices. They do not know the prices quoted by other AMMs. If they could also use this information, this could further refine the updating and improve the learning.



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the counterfactual-updating Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 100.69 and 99.31, respectively.

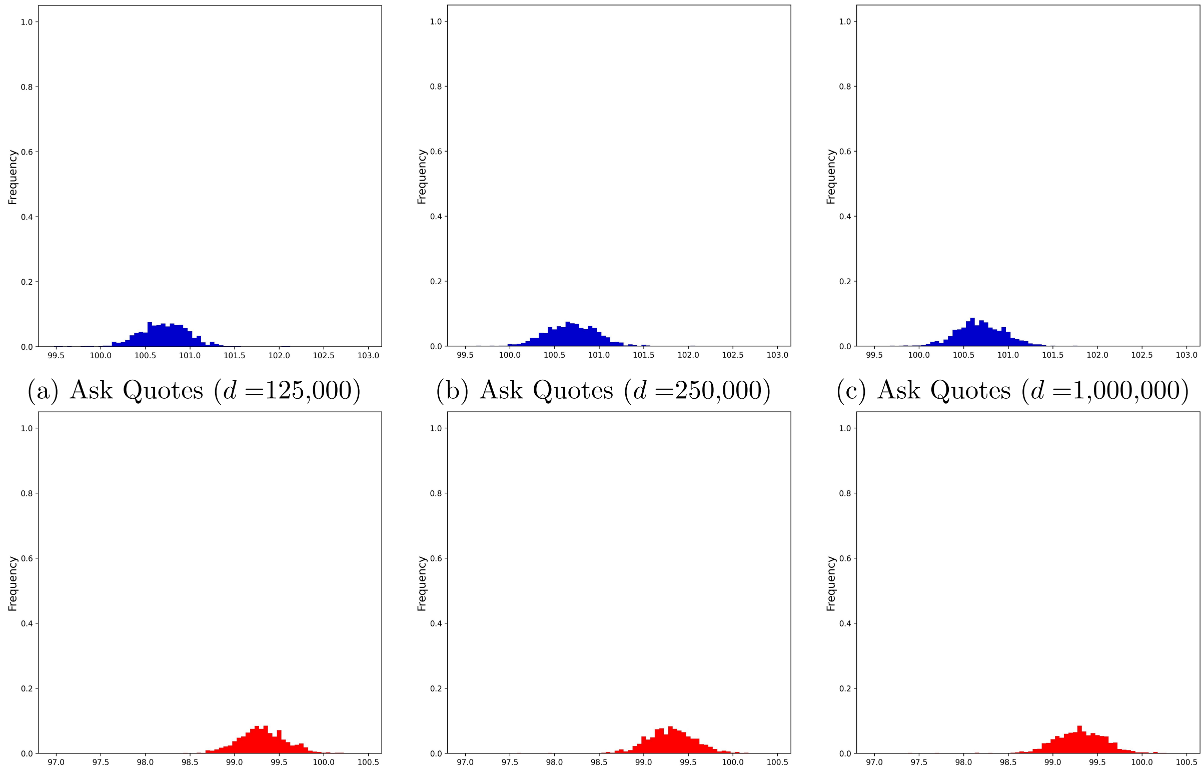
Figure 11: Counterfactual Updating - Mean Market Quotes Over Days

of the one million days. As can be seen both in the range of market quotes in Figure 11 and the distributions in Figure 12, the counterfactual updating also leads to more noise than the baseline case, as multiple prices continue to be updated simultaneously.

9.1 The Competitive Equilibrium Pricing Result

The contrast between these simulation results and those we obtained for the baseline case could not be stronger. Letting the AMMs have a minimal understanding of the market is enough to move from loss-free pricing to competitive pricing.

To understand what is at the root of these simulation findings, we now provide a formal result. To obtain a sharp theoretical result, we assume that the learning rates λ_i^d in the updating rule satisfy the Robbins-Monro conditions, $\sum_d \lambda_i^d = \infty$ and $\sum_d \left(\lambda_i^d\right)^2 < \infty$, which will allow us to rely on continuous time approximation to analyze the long-run dynamics of the system (see Benaïm (1999) for an exposition of the mathematical background establishing such connections). The reader familiar with Q-learning will notice that these conditions are the same as used in the classic convergence theorem of Watkins and Dayan (1992). They guarantee that decision makers (the AMMs in our context) continue to learn (the learning rate decays slowly), but the learning

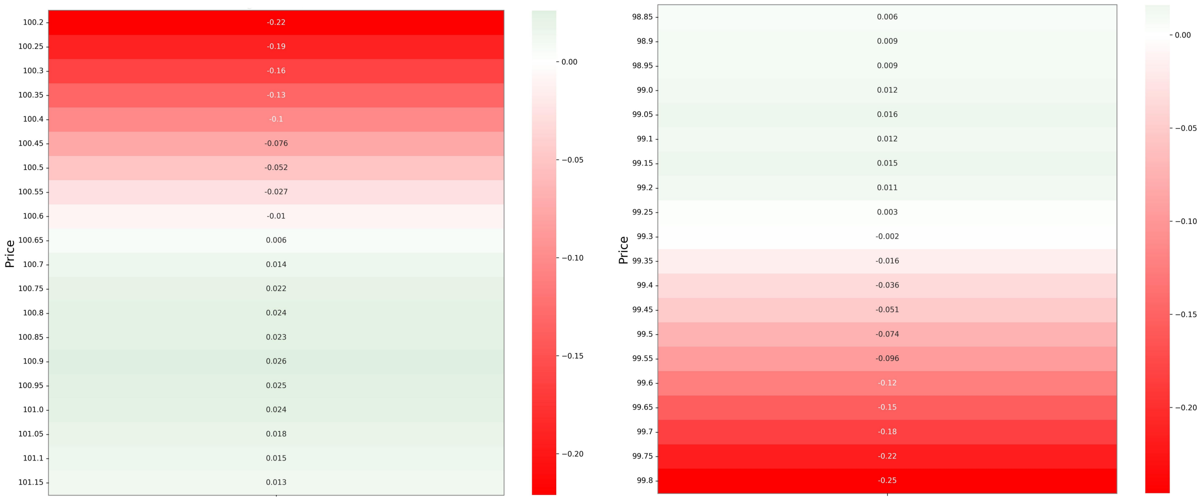


(a) Ask Quotes (d =125,000) (b) Ask Quotes (d =250,000) (c) Ask Quotes (d =1,000,000)

(d) Bid Quotes (d =125,000) (e) Bid Quotes (d =250,000) (f) Bid Quotes (d =1,000,000)

The graph plots the distributions of market ask and bid prices across the 1,000 repetitions at different points in the simulation: days 125,000, 250,000, and 1,000,000. The graph refers to the counterfactual-updating Q-learning setup.

Figure 12: Counterfactual Updating – Distributions of Market Ask and Bid Quotes over Time



(a) Ask Q-Table (b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the counterfactual-updating Q-learning setup.

Figure 13: Counterfactual Updating - Final Mean Q-Tables

rate is not too fast (the decay is not too slow). Similarly to the theoretical result in the baseline Q-learning model presented in Section 6, we also assume that there is no experimentation, although experimentation would not alter our conclusions much. Finally, we assume that any ask and bid prices in the continuum can be considered, even if similar conclusions would hold for the case of the finite ask and bid grid that we have employed in our simulations.

With these premises, we are now ready to state that, under a minimal condition on the initial Q-values, similar to that we used for Proposition 1, there is asymptotic convergence to the competitive equilibrium ask and bid prices, a^C and b^C:

Proposition 3. *Suppose that, for every AMM$_i$, the sequences of λ_i^d satisfies $\sum_d \lambda_i^d = \infty$ and $\sum_d \left(\lambda_i^d\right)^2 < \infty$, and, moreover, that $\varepsilon_i^d = 0$ for all d; assume that $\rho > 0.5$ and suppose that, for every AMM$_i$, for at least one $\alpha_j \geq a^C$ and at least one $\beta_k \leq b^C$, the initial Q-values, $q_i^1(\alpha_j)$ and $\hat{q}_i^1(\beta_k)$, are weakly positive. Then, asymptotically, in the limit of a continuous grid of ask and bid prices, every AMM$_i$ quotes the competitive ask price a^C and bid price b^C almost surely.*

Proof. The proof is for the ask price; the proof for the bid price follows a similar logic.

As a preliminary note, in this proof, we find it convenient to change the notation and, instead of d (for days), we use t (for time), which is more common in continuous-time analysis. Moreover, to simplify the exposition, we will consider the case of $\lambda_i^t = \lambda^t$ for every i.

Given the conditions for λ^t, we can employ a continuous-time approximation of the discrete-time dynamics of Q-values for the ask prices. More precisely, we use Propositions 4.1 and 4.2, along with the Limit Set Theorem (Theorem 5.7), of Benaïm (1999) to ensure that the asymptotics of the discrete-time dynamics are well approximated by the continuous-time motion that we now describe.

Calling $\alpha^{\min}$ the prevailing ask price at time t (recall that this is the minimum ask quoted by all AMMs at that time), the continuous time approximation of the motion of Q-values for any ask price α strictly below $\alpha^{\min}$ writes

$$\frac{dq_i^t}{dt}(\alpha) = \rho[(1-\mu)\frac{\eta}{2} + \mu\delta][\alpha - a^C] - q_i^t(\alpha). \tag{25}$$

This expression can be understood as follows. The Q-values of ask prices below $\min(v_H, \alpha^{\min})$ can be updated with the counterfactual formula. When the ask price is $\alpha^{\min}$, the expected payoff obtained with any ask price strictly below $\min(v_H, \alpha^{\min})$ is $[(1-\mu)\frac{\eta}{2} + \mu\delta][\alpha - a^C]$ since trade on this side occurs with probability $[(1-\mu)\frac{\eta}{2} + \mu\delta]$ and, when it occurs, it results in an expected payoff of $\alpha - a^C$ (recall that informed traders buy if $v = v_H$, and recall, also from our updating formulas that, when trade is on the other side, or there is no trade, the payoff is zero); moreover, given that $\alpha < \alpha^{\min}$, it was not chosen at time t, hence the payoff is updated with the ρ discount.

In the same spirit, for an AMM$_i$ quoting $\alpha^{\min}$, we have:

$$\frac{dq_i^t}{dt}(\alpha^{\min}) = \frac{1}{M}[(1-\mu)\frac{\eta}{2} + \mu\delta][\alpha^{\min} - a^C] - q_i^t(\alpha^{\min}) \tag{26}$$

where M is the number of AMMs quoting $\alpha^{\min}$ at time t. The difference with the previous expression is that there is no ρ discount in the update, since the ask price $\alpha^{\min}$ is chosen by AMM$_i$ and, in the case several (i.e., M) AMMs choose $\alpha^{\min}$, AMMs are selected to be the seller with equally probability.

Now, define α_{inf} as the infimum of the market ask prices that can be quoted asymptotically and let α_{inf}^- be the next ask price below α_{inf} in the grid (assumed to be close to the continuous grid).

We first show that α_{inf} cannot strictly exceed a^C (in the limit of the fine grid). This follows because

$$\frac{dq_i^t}{dt}(\alpha_{\text{inf}}^-) - \frac{dq_i^t}{dt}(\alpha_{\text{inf}}) > 0$$

for every AMM$_i$ whenever $\alpha_{\text{inf}} > a^C$ and $\rho > 0.5$ and the minimum ask price is α_{inf}. (When the minimum ask price is different from α_{inf}^-, $\frac{dq_i^t}{dt}(\alpha_{\text{inf}}^-) - \frac{dq_i^t}{dt}(\alpha_{\text{inf}})$ gets arbitrarily close to zero in the limit of the continuous grid.) This implies that eventually, for some $\bar{t}$ and for every AMM$_i$, we must have that, for all $t > \bar{t}$, $q_i^t(\alpha_{\text{inf}}^-) > q_i^t(\alpha_{\text{inf}})$, implying that α_{inf} is not chosen infinitely many times by AMM$_i$.

We next observe that we cannot have $\alpha_{\text{inf}} < a^C$ as this would imply that eventually, for some $\bar{t}$ and every AMM$_i$ and $t > \bar{t}$, $q_i^t(\alpha_{\text{inf}}) < 0$, making it impossible that α_{inf} is chosen infinitely many times by AMM$_i$. We conclude that $\alpha_{\text{inf}} = a^C$.

Consider an arbitrary accumulation point α^* of the sequences of ask prices a_i^t for the various AMM$_i$. We must have that $\alpha^* \geq \alpha_{\text{inf}} = a^C$ as we have just shown. But $\alpha^* > a^C$ can be ruled out as, by the same argument as the one used above, we would have that, for some $\bar{t}$ and for all AMM$_i$, it must be that, for all $t > \bar{t}$, $q_i^t(\alpha^{*-}) > q_i^t(\alpha^*)$, implying that α^* is not chosen infinitely many times by any AMM$_i$. Q. E. D. □

At an intuitive level, when the ask on some given day d is $\alpha^{\min}$, any AMM can assess that with a counterfactual ask price slightly lower than $\alpha^{\min}$, they would get $\alpha^{\min} - V^d$ in the case of a sale and zero otherwise. This gives rise to a stochastic payoff, but the conditions on λ_i^d stated in the proposition ensure that, in the limit (as d grows large), one can reason with the induced mean of the reward. That is, using a continuous-time approximation, after some re-writing and after noting that counterfactual payoffs are discounted by the ρ factor, one gets (25). Similarly, for the on-path ask price $\alpha^{\min}$, one gets (26). For $\rho > 0.5$, these expressions ensure that the Q-value of an ask price slightly lower than $\alpha^{\min}$ must grow more quickly than the Q-value of $\alpha^{\min}$ whenever $\alpha^{\min}$ lies strictly above the competitive ask price a^C.[17] Therefore, starting from ask prices above the competitive one, AMMs will eventually drift their choice of ask prices toward the competitive one through a gradual undercutting process. The system will not go beyond the competitive ask price because, at a lower ask price, AMMs would be making losses in expectation. This overall dynamics leads to the competitive pricing we described in Section 3. Note that the argument works even in the absence of any experimentation (whereas in the classic one-agent problem, Q-learning requires experimentation to obtain the optimal policy - see Watkins and Dayan, 1992). Intuitively, at ask (bid) prices higher (lower) than the competitive equilibrium, the counterfactual updating offers enough information that experimentation is not needed.

To emphasize the logic of our proof, we now focus on just one step of the process: we consider the case of one AMM competing against a market maker who uses a fixed ask (higher than the competitive ask price) and a fixed bid prices (lower than the competitive bid price).

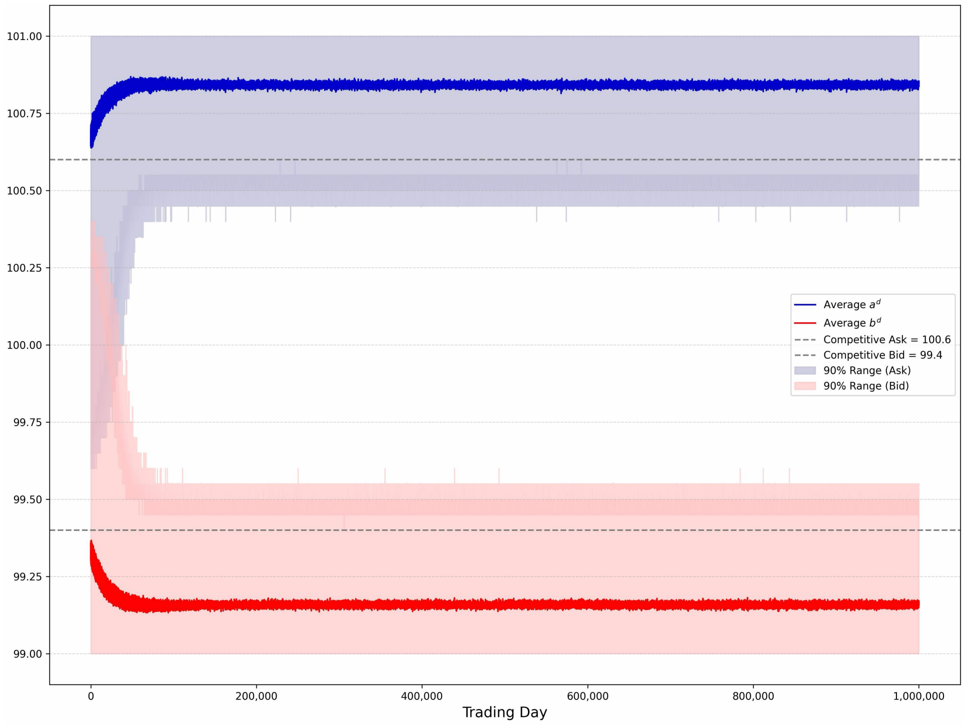
Proposition 4. *Consider the case of one AMM (denoted AMM$_1$) who competes with a market maker choosing a fixed ask price $\overline{\alpha}$ that is at least two ticks above a^C and a fixed bid price $\overline{\beta}$ that is at least two ticks below b^C in the respective finite grids of ask and bid prices. Suppose that the sequences of λ_1^d satisfy $\sum_d \lambda_1^d = \infty$ and $\sum_d \left(\lambda_1^d\right)^2 < \infty$, and, moreover, that $\varepsilon_1^d = 0$ for all d; assume that $\rho > 0.5$ and suppose that, for every AMM$_i$, for at least one $\alpha_j \geq a^C$ and at least one $\beta_k \leq b^C$, the initial Q-values, $q_i^1(\alpha_j)$ and $\hat{q}_i^1(\beta_k)$, are weakly positive. Then, asymptotically, almost surely, AMM$_1$ quotes an ask price that is one tick below $\overline{\alpha}$ and a bid price that is one tick above $\overline{\beta}$, and, as a result, AMM$_1$ does all the trading.*

Proof. For AMM$_1$, one can obtain the same expressions (25) and (26) in the proof of the previous proposition. From these expressions, it appears that $q_i^t(\overline{\alpha}^-)$ (defined as in the previous proof) grows more quickly than the Q-value of any other ask price, thereby leading, eventually, AMM$_1$ to choose an ask price equal to $\overline{\alpha}^-$. Q. E. D. □

With counterfactual updating, Q-learning produces exactly the result that someone using

[17] The sufficient condition $\rho > 0.5$ comes from the observation that, independently of the number N of AMMs, an AMM could be quoting the best price with only one other AMM; by undercutting, the AMM would *double* their payoff. They would more than double it in other cases.

an algorithmic market maker would like to see: when facing another market maker using a trivial pricing strategy (fixed bid and ask prices), the AMM just undercuts the competitor and earns all trading profits. Figure 14 shows the results, with experimentation, for a single AMM competing against a fixed-price market maker.



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the counterfactual-updating Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 100.84 and 99.16, respectively. The parameter ρ is set equal to 1.

Figure 14: Counterfactual Updating, Fixed-price Market Maker - Mean Market Quotes Over Days

Given the stark contrast in results between the baseline Q-learning (Propositions 1 and 2 and related simulations) and the counterfactual-updating Q-learning (Propositions 3 and 4 and related simulations), we would like to re-emphasize the mechanisms that lead to them. For exposition, let us focus on the ask prices. Suppose that, at a particular trading time, the ask with the highest Q-value for an AMM is in between the competitive equilibrium ask and the highest value of the asset. In expectation, this ask price is profitable. In the baseline Q-learning with a fixed learning rate, despite the value for this ask being positive in expectation, the corresponding Q-value will eventually become negative and, at that point, the AMM will not choose it again (for ever, if there is no experimentation). The reason the Q-value will eventually become negative is that, almost surely, there will be a sufficiently long sequence of bad draws in which the AMM will make negative profits, which will push the Q-value to the negative region. Thus, although profitable in expectation, this ask price cannot be chosen in the long run. Observe that the logic of this result holds independently of the size of the learning rate, even if, with a smaller learning rate, a longer sequence of bad draws may be needed to make the Q-value cross the negative region.

Consider now the case of the counterfactual-updating Q-learning. Here, there is a "race" among Q-values, since more than one Q-value is updated every time. In expectation, the Q-values for smaller asks (but higher than the competitive equilibrium one) will grow faster because the AMM choosing them is the one that manages to make the entire trade while still selling at a profitable price. This eventually leads to the competitive equilibrium (or to an ask just below the fixed price, in the individual decision-making case), by the standard logic of the undercutting. While our formal results are obtained for sequences of learning rates satisfying the Robbins-Monro conditions (which allow for crisper results using stochastic approximation results), qualitatively similar results would be obtained in the case of constant but small learning rates. Unlike in the baseline Q-learning, even assuming constant but small learning rates, if the chosen ask prices were consistently above a level strictly higher than the competitive ask price, any ask price price strictly below that level would asymptotically get a Q-value close to the correct mean value associated with that ask price (because updating would take place for all realizations of the uncertainty and the law of large number could be used to establish this). Our simulations confirm this and also show that the results we obtain analytically extend to other parameter value and, in particular, to the case of experimentation.

10 Conclusion

We have studied how algorithmic market makers using a stateless Q-learning algorithm price a financial asset of uncertain value. We have shown that the result varies from loss-free pricing to competitive pricing, depending on how the algorithms are set up. When the algorithms update only the values of the prices they have used, we can get the extreme of loss-free prices. When, instead, they are endowed with even a minimal understanding of the market (that traders prefer to buy at lower prices and sell at higher prices) and some minimal information (the prevailing market quotes), then we get competitive prices. In our stateless Q-learning, there is no collusion.

Of course, algorithmic market makers can use much more information and condition on many states, including, for instance, the level of their inventories (see, e.g., Ganesh et al., 2019 and other papers reviewed in Bai et al., 2024). In our work, we have considered a very simple, stateless, setup, with the purpose of making the causes of supra-competitive or competitive prices transparent. We think this is important both for positive reasons, for our understanding of markets, and for normative reasons, for regulatory and policy aims. There is no collusion arising in our context, but of course we do not claim that there is no collusion in real markets; this is an empirical question to which our theoretical work does not aim to speak. We also do not exclude that supra-competitive prices can arise due to other mechanisms (see, e.g., the work by Dou et al., 2024, studying algorithmic traders rather than algorithmic market makers).

We have considered a simple setup (one trader "per day" in our terminology) in which stationarity allows the use of stateless Q-learning. With sequences of trades, the number of Q-values grows very quickly. Dealing with such natural extensions would require, for practical purposes, amending the basic Q-learning so that several Q-values are updated at the same time. This could call for extensions such as the Coarse Q-learning in which different alternatives are bundled into similarity classes, as explored in decision problems by Jehiel and Satpathy (2025).

References

Asker, J., Fershtman, C., & Pakes, A. (2024). The Impact of Artificial Intelligence Design on Pricing. *Journal of Economics & Management Strategy*, *33*(2), 276–304. https://doi.org/10.1111/jems.12516

Bai, Y., Gao, Y., Wan, R., Zhang, S., & Song, R. (2024, November). A Review of Reinforcement Learning in Financial Applications [arXiv:2411.12746 [q-fin]]. https://doi.org/10.48550/arXiv.2411.12746

Benaïm, M. (1999). Dynamics of stochastic approximation algorithms. In J. Azéma, M. Émery, M. Ledoux, & M. Yor (Eds.), *Séminaire de Probabilités XXXIII* (pp. 1–68). Springer. https://doi.org/10.1007/BFb0096509

Calvano, E., Calzolari, G., Denicolò, V., & Pastorello, S. (2020). Artificial Intelligence, Algorithmic Pricing, and Collusion. *American Economic Review*, *110*(10), 3267–3297. https://doi.org/10.1257/aer.20190623

Camerer, C., & Ho, T.-H. (1999). Experience-weighted Attraction Learning in Normal Form Games. *Econometrica*, *67*(4), 827–874. https://doi.org/10.1111/1468-0262.00054

Cartea, Á., Chang, P., & Penalva, J. (2022, May). Algorithmic Collusion in Electronic Markets: The Impact of Tick Size. https://doi.org/10.2139/ssrn.4105954

Cipriani, M., & Guarino, A. (2008). Herd Behavior and Contagion in Financial Markets. *The B.E. Journal of Theoretical Economics*, *8*(1). https://doi.org/10.2202/1935-1704.1390

Cipriani, M., & Guarino, A. (2014). Estimating a Structural Model of Herd Behavior in Financial Markets. *The American Economic Review*, *104*(1), 224–251. https://doi.org/10.1257/aer.104.1.224

Colliard, J.-E., Foucault, T., & Lovo, S. (2022). Algorithmic Pricing and Liquidity in Securities Markets. https://doi.org/10.2139/ssrn.4252858

Dou, W. W., Goldstein, I., & Ji, Y. (2024, May). AI-Powered Trading, Algorithmic Collusion, and Price Efficiency. https://doi.org/10.2139/ssrn.4452704

Easley, D., Kiefer, N. M., & O'Hara, M. (1997). One Day in the Life of a Very Common Stock. *The Review of Financial Studies*, *10*(3), 805–835. https://doi.org/10.1093/rfs/10.3.805

Easley, D., López de Prado, M. M., & O'Hara, M. (2012). Flow Toxicity and Liquidity in a High-frequency World. *The Review of Financial Studies*, *25*(5), 1457–1493. https://doi.org/10.1093/rfs/hhs053

Ganesh, S., Vadori, N., Xu, M., Zheng, H., Reddy, P., & Veloso, M. (2019). Multi-Agent Simulation for Pricing and Hedging in a Dealer Market.

Glosten, L. R., & Milgrom, P. R. (1985). Bid, ask and transaction prices in a specialist market with heterogeneously informed traders. *Journal of Financial Economics*, *14*(1), 71–100. https://doi.org/10.1016/0304-405X(85)90044-3

Jehiel, P., & Satpathy, A. (2025). Learning to be Indifferent in Complex Decisions: A Coarse Payoff-Assessment Model.

Mookherjee, D., & Sopher, B. (1997). Learning and Decision Costs in Experimental Constant Sum Games. *Games and Economic Behavior*, *19*(1), 97–132. https://doi.org/10.1006/game.1997.0540

Sarin, R., & Vahid, F. (1999). Payoff Assessments without Probabilities: A Simple Dynamic Model of Choice. *Games and Economic Behavior*, *28*(2), 294–309. https://doi.org/10.1006/game.1998.0702

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press.

Watkins, C. J. C. H. (1989). *Learning from Delayed Rewards* [Doctoral dissertation].

Watkins, C. J. C. H., & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*(3), 279–292. https://doi.org/10.1007/BF00992698

A Counterfactual Updating - Bid-side Updating

Here, we repeat the explanation of the counterfactual updating but for the bid-side updating.

1. **AMM$_i$ quotes $\boldsymbol{b_i^d = \beta_j = b^d}$ buys the asset.** The AMM$_i$ updates the Q-values for β_j on the basis of the *realized* payoff and the Q-values for all the bids greater than β_j for the payoffs they would have received by buying at those prices. AMM$_i$ does not update the Q-values for any bid less than β_j since they do not know whether they would have been able to buy. Formally, the Q-values are updated as follows:

$$\hat{q}_i^{d+1}(\beta_j) = \lambda_i^d(v^d - \beta_j) + (1 - \lambda_i^d)\hat{q}_i^d(\beta_j) \tag{27}$$

$$\hat{q}_i^{d+1}(\beta_k) = \lambda_i^d\rho(v^d - \beta_k) + (1 - \lambda_i^d)\hat{q}_i^d(\beta_k), \qquad \text{for } \beta_k > \beta_j \tag{28}$$

$$\hat{q}_i^{d+1}(\beta_l) = \hat{q}_i^d(\beta_l), \qquad \text{for } \beta_l < \beta_j \tag{29}$$

2. **AMM$_i$ quotes $\boldsymbol{b_i^d \leq \beta_j = b^d}$ and another AMM buys the asset.** Since the AMM is aware of their own bid price and of the market bid at which the transaction occurred, they update the Q-values for all the bid as follows: for any bid strictly less than β_j, the Q-value is updated with a payoff of zero; if $b_i^d < \beta_j$, the Q-value is updated with the *counterfactual* payoff (i.e., taking into account that the AMM would have bought with a probability of $\frac{1}{n+1}$, where n is the number of AMMs choosing the market bid); if, instead, $b_i^d = \beta_j$, then the Q-value is updated with the *realized* payoff (i.e., zero); and, for the bid prices strictly higher than β_j, the Q-values are updated with the counterfactual payoff.

$$\hat{q}_i^{d+1}(\beta_l) = (1 - \lambda_i^d)\hat{q}_i^d(\beta_l), \qquad \text{for all } \beta_l < \beta_j, \tag{30}$$

$$\hat{q}_i^{d+1}(\beta_j) = \lambda_i^d\rho\frac{1}{n+1}(v^d - \beta_j) + (1 - \lambda_i^d)\hat{q}_i^d(\beta_j), \qquad \text{if } b_i^d < \beta_j \tag{31}$$

$$\hat{q}_i^{d+1}(\beta_j) = (1 - \lambda_i^d)\hat{q}_i^d(\beta_j), \qquad \text{if } b_i^d = \beta_j, \tag{32}$$

$$\hat{q}_i^{d+1}(\beta_k) = \lambda_i^d\rho(v^d - \beta_k) + (1 - \lambda_i^d)\hat{q}_i^d(\beta_k), \qquad \text{for all } \beta_k > \beta_j. \tag{33}$$

3. **AMM$_i$ quotes $\boldsymbol{b_i^d \leq \beta_j = b^d}$ and there is no trade.** AMM$_i$ updates the Q-values for all prices that are smaller than or equal to β_j since the payoff would have been zero. AMM$_i$ does not update the Q-values for the higher bid, since they do not know whether

setting a higher bid would have led to a buy.

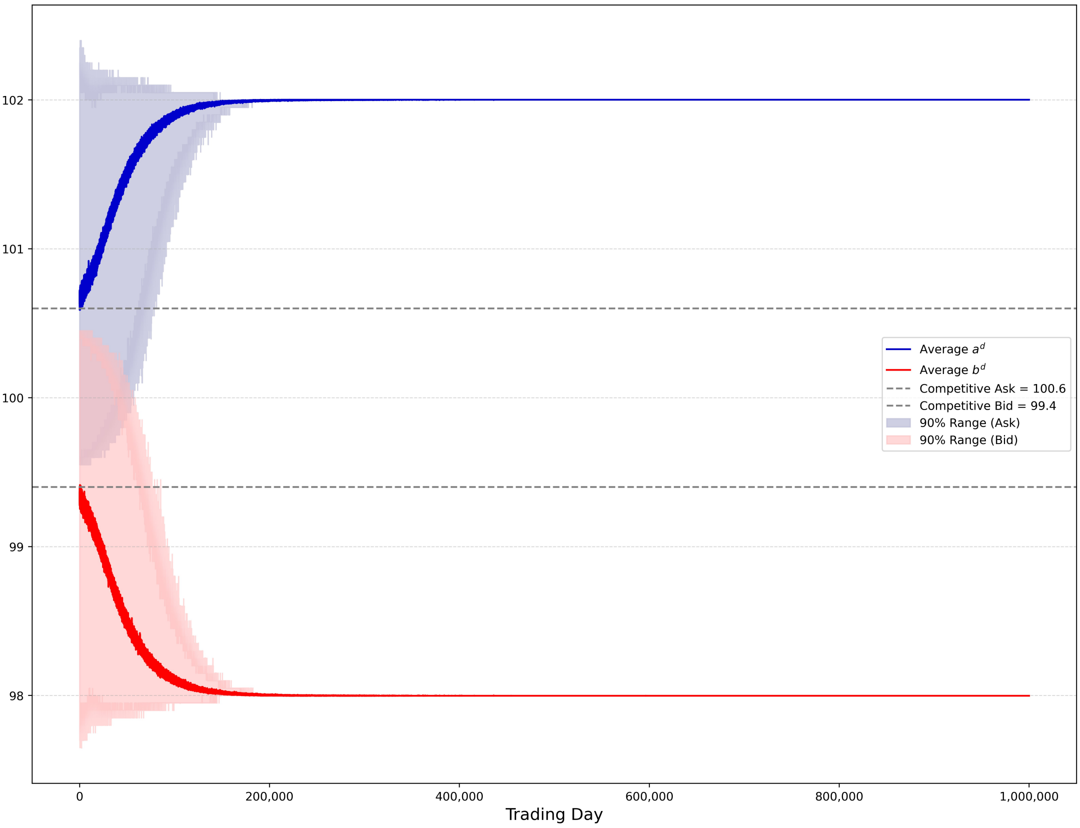
$$\hat{q}_i^{d+1}(\beta_k) = \hat{q}_i^d(\beta_k), \qquad \text{for } \beta_k > \beta_j, \tag{34}$$

$$\hat{q}_i^{d+1}(\beta_l) = (1 - \lambda_i^d)\hat{q}_i^d(\beta_l), \qquad \text{for } \beta_l \leq \beta_j. \tag{35}$$

B Additional Results

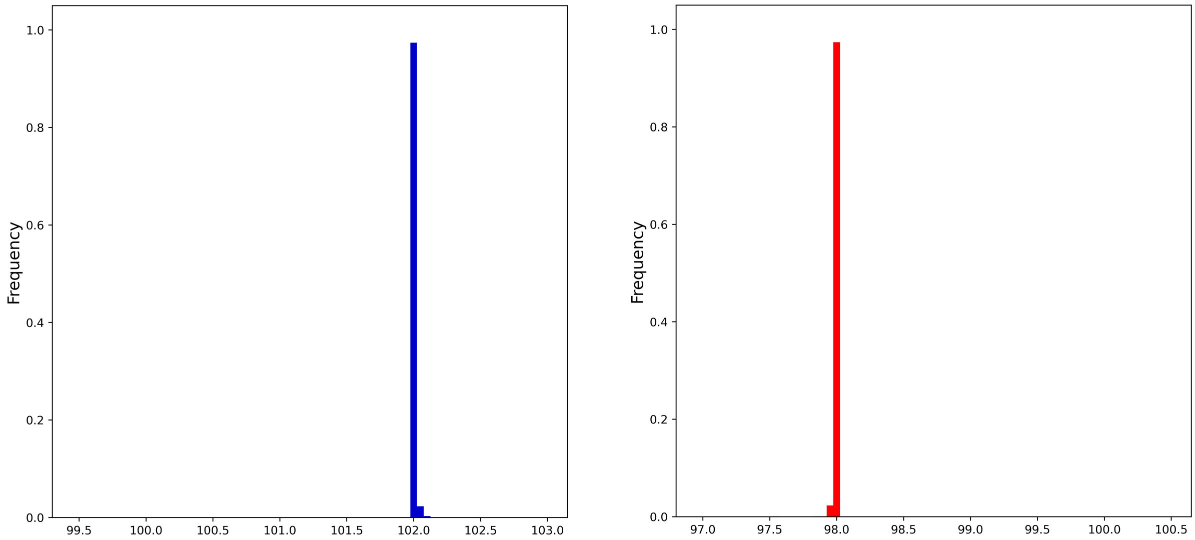
B.1 Lower Initial Q-values

In Section 6, we considered an "optimistic" distribution of initial Q-values, i.e., the initial Q-values were initialized on a uniform distribution over $[5, 8]$. For this section, we initialize using positive but lower initial Q-values, namely, a uniform distribution over $[0.01, 0.05]$. All other parameters are as listed in Table 1. We find no significant difference between the results below and those discussed in Section 6.



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 102 and 98, respectively.

Figure B1: Lower Initial Q-values - Mean Market Quotes Over Days

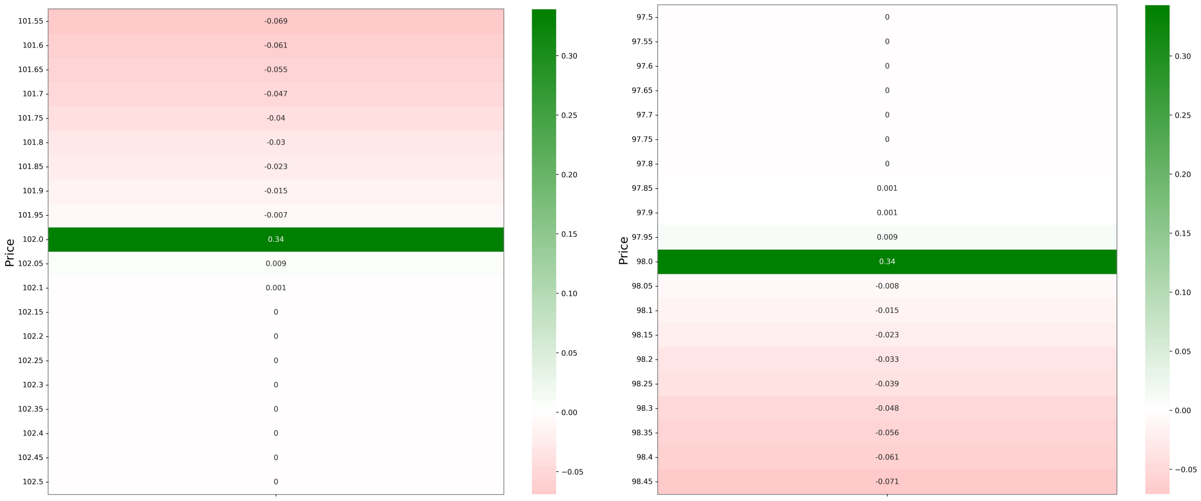


(a) Ask

(b) Bid

The graph shows the distribution of best ask quotes across the 1,000 repetitions in the final day for both the ask and bid quotes. The graph refers to the baseline Q-learning setup.

Figure B2: Lower Initial Q-values - Distribution of AMM Quotes in $d = D = 1,000,000$



(a) Ask Q-Table

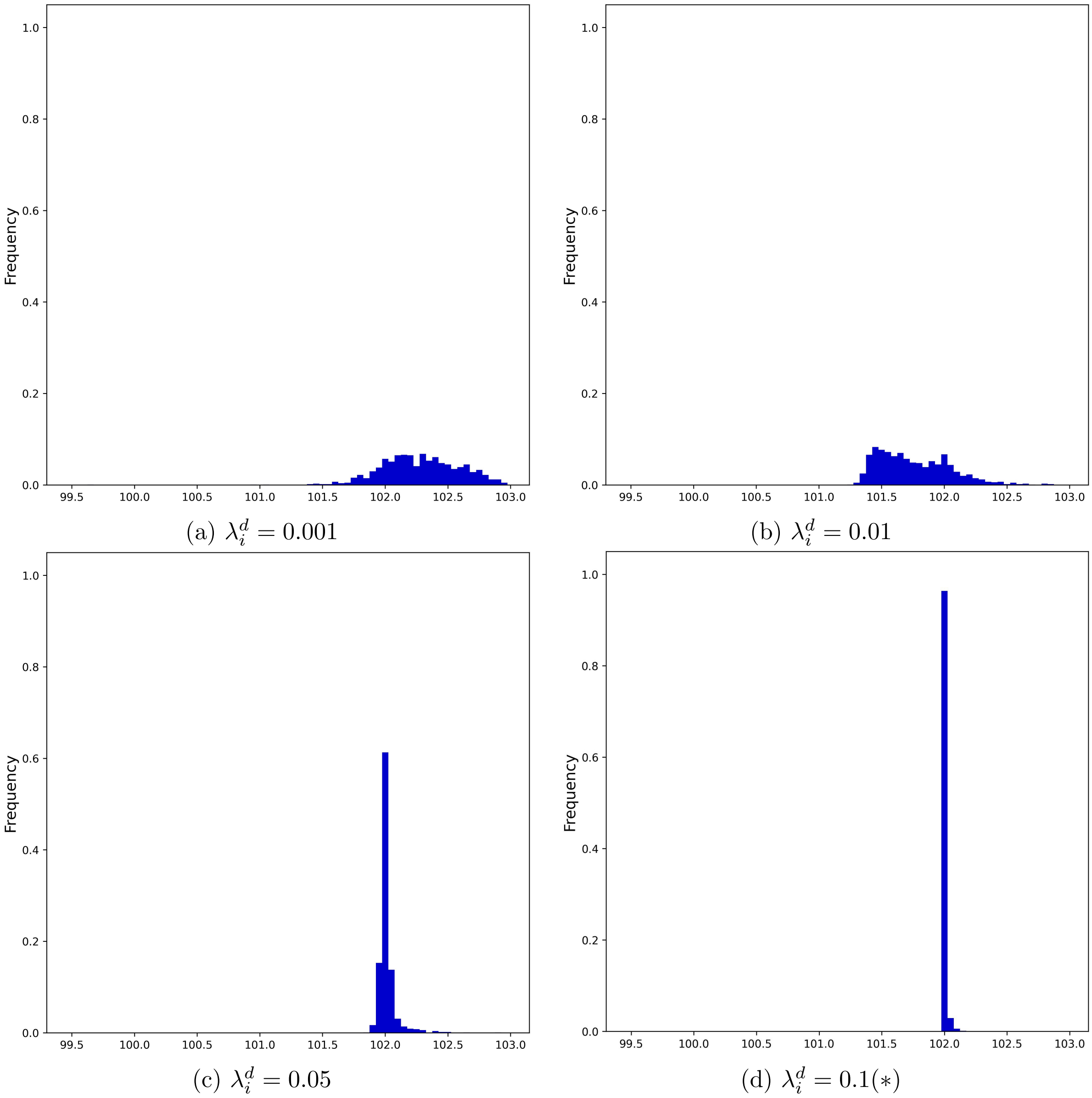
(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure B3: Lower Initial Q-values - Final Mean Q-Tables

B.2 Varying the Learning Rate

The distributions in Figures B4 and B5, and the Final Mean Q-Tables in Figures B6, B7, and B8 help to show the difference with the baseline case. The distribution of asks in the the final day has a mass below v_H and that of bid prices above v_L, in contrast with the baseline case. The mean Q-values are now positive even for ask prices below 102. Whilst, on average, the highest Q-value is still attached to an ask price of 102, there are simulations in which the highest Q-value is attached to a price below this; thus, in these simulations, the greedy action was below 102.



(a) $\lambda_i^d = 0.001$

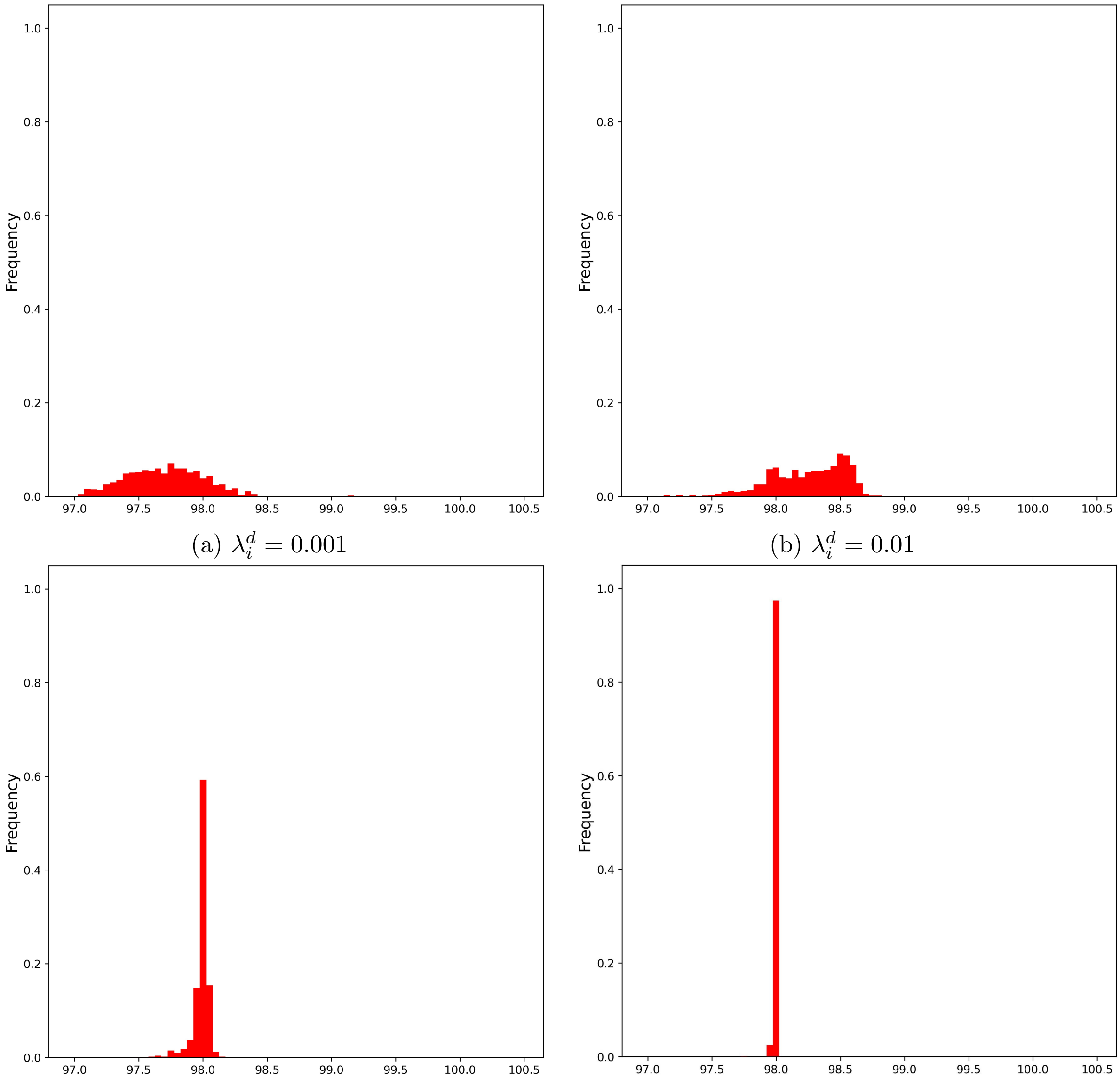
(b) $\lambda_i^d = 0.01$

(c) $\lambda_i^d = 0.05$

(d) $\lambda_i^d = 0.1(*)$

Each panel shows the distribution of the market ask quotes across the 1,000 repetitions in the final day, for a given value of λ_i^d. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6

Figure B4: λ_i^d Comparative Statics - Distribution of Market Ask Prices in Day $d = 1{,}000{,}000$



(a) $\lambda_i^d = 0.001$

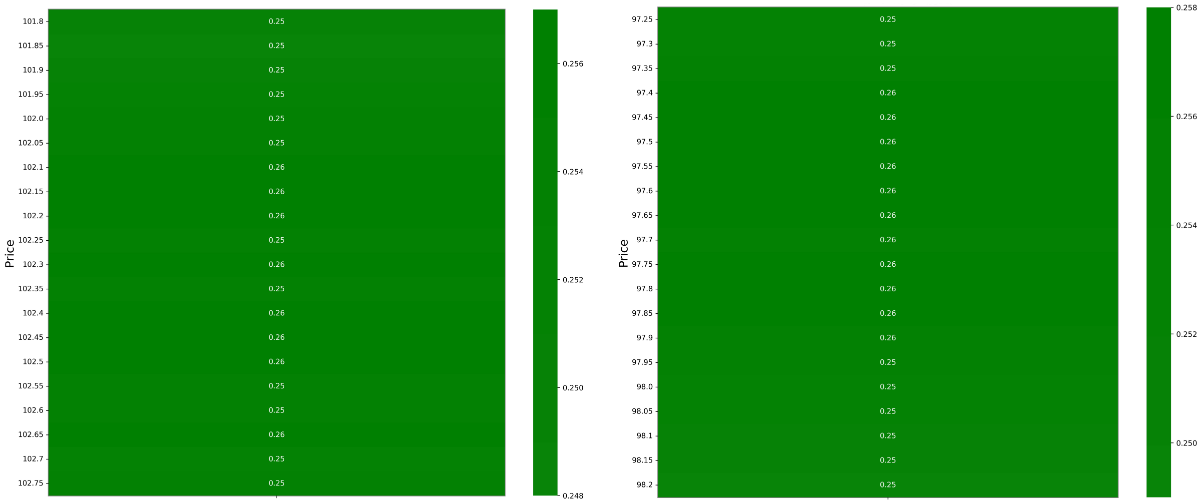
(b) $\lambda_i^d = 0.01$

(c) $\lambda_i^d = 0.05$

(d) $\lambda_i^d = 0.1(*)$

Each panel shows the distribution of the market ask quotes across the 1,000 repetitions in the final day, for a given value of λ_i. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6

Figure B5: λ_i^d Comparative Statics - Distribution of Market Bid Prices in Day $d = 1{,}000{,}000$

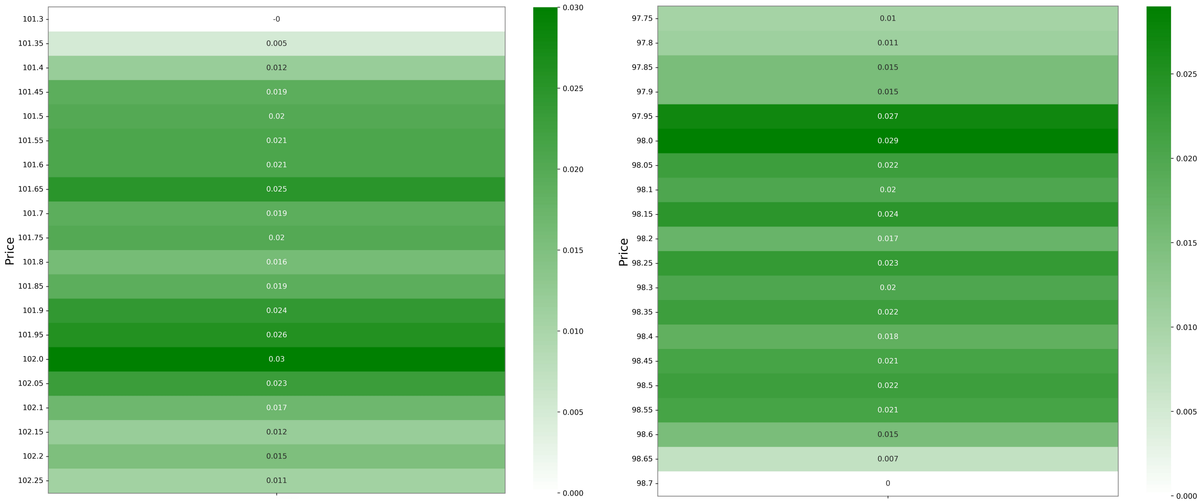


(a) Ask Q-Table

(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup. The graph refers to the baseline Q-learning setup.

Figure B6: $\lambda_i^d = 0.001$ - Final Mean Q-Tables

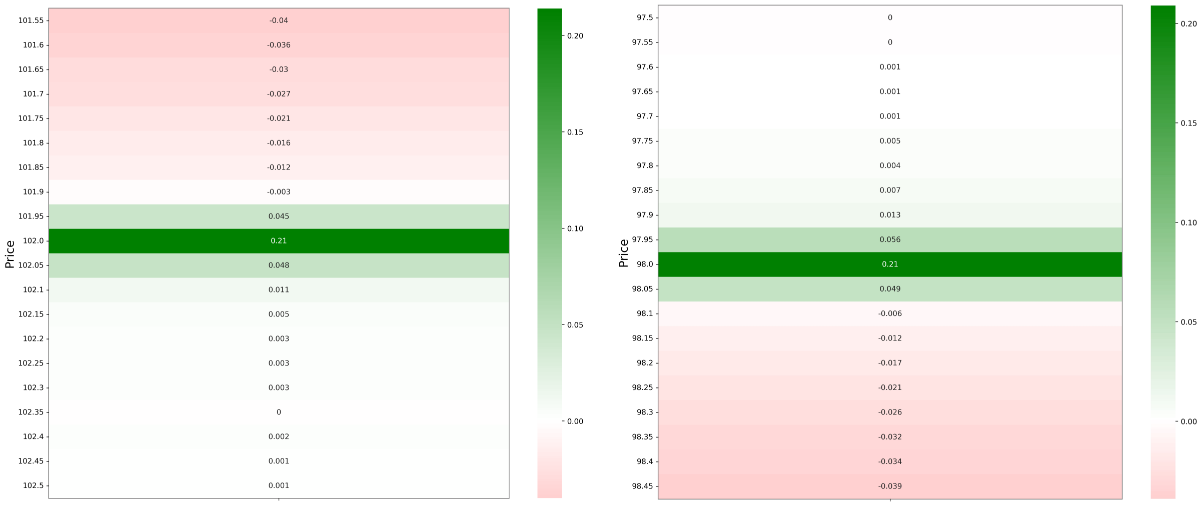


(a) Ask Q-Table

(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure B7: $\lambda_i^d = 0.01$ - Final Mean Q-Tables



(a) Ask Q-Table

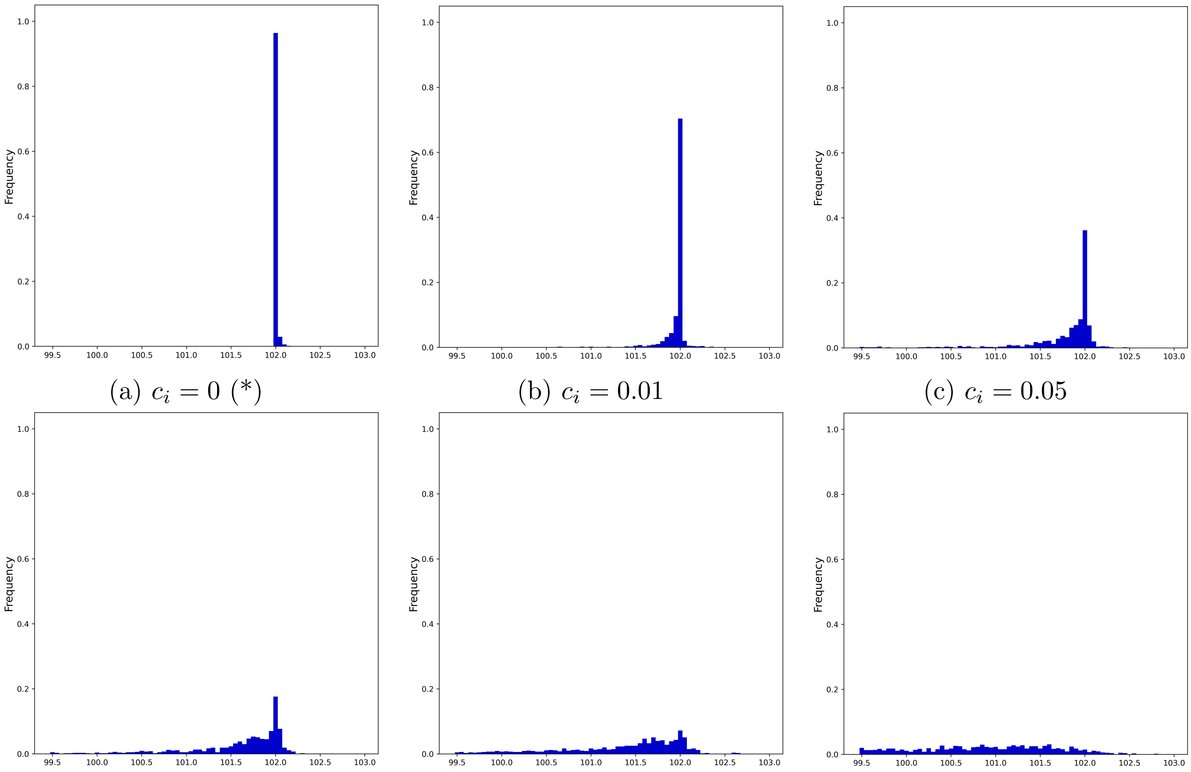
(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure B8: $\lambda_i^d = 0.05$ - Final Mean Q-Tables

B.3 Varying the Minimum Exploration Probability

Below, we present the market ask and bid distributions in the final day for the values of c_i considered in Section 7.2.



(a) $c_i = 0$ (*)

(b) $c_i = 0.01$

(c) $c_i = 0.05$

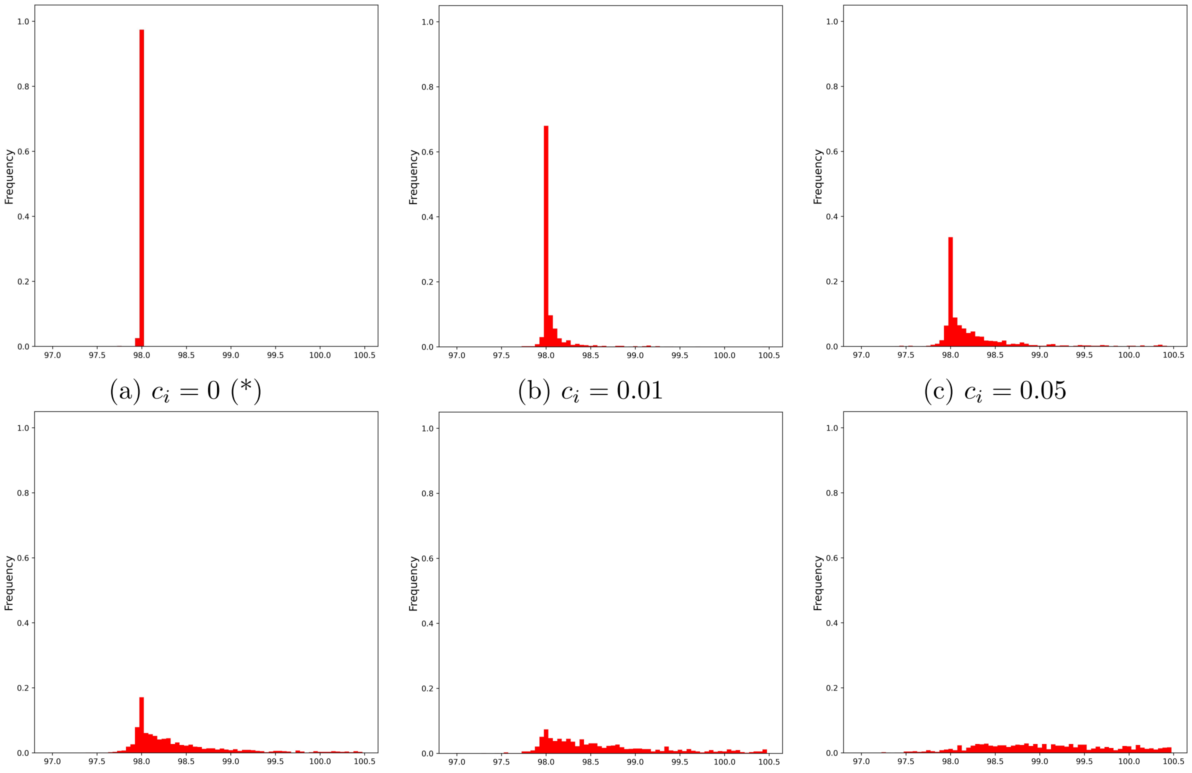
(d) $c_i = 0.10$

(e) $c_i = 0.20$

(f) $c_i = 0.50$

Each panel shows the distribution of the market ask quotes across the 1,000 repetitions in the final day, for a given value of c_i. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6.

Figure B9: c_i Comparative Statics - Distribution of Market Ask Prices in Day $d = 1{,}000{,}000$



(a) $c_i = 0$ (*)

(b) $c_i = 0.01$

(c) $c_i = 0.05$

(d) $c_i = 0.10$

(e) $c_i = 0.20$

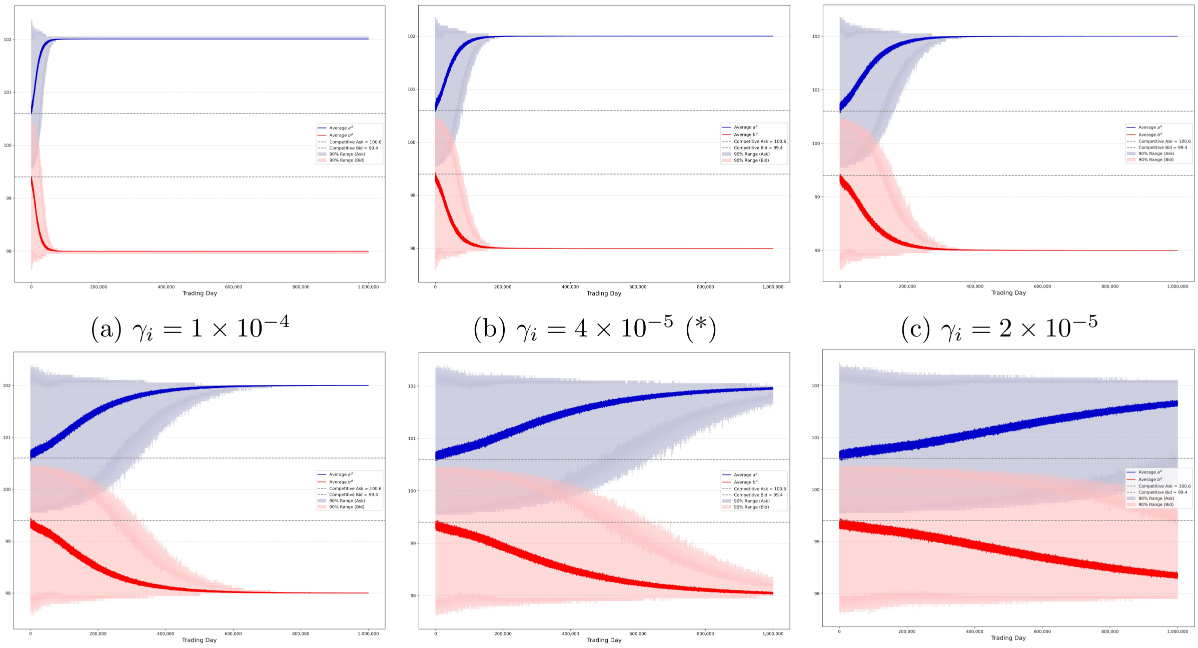
(f) $c_i = 0.50$

Each panel shows the distribution of the market bid quotes across the 1,000 repetitions in the final day, for a given value of c_i. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6.

Figure B10: c_i Comparative Statics - Distribution of Market Bid Prices in Day $d = 1{,}000{,}000$

B.4 Varying the Exploration Decay Rate

Figure B11 shows the market quotes over days, with each subplot representing a different value of γ_i. For a larger γ_i, we see that the simulations reach the result at an accelerated speed; as we decrease γ_i towards zero, we see that the time taken for the simulations to reach ask prices of 102 and bid prices of 98 increases. At the extreme of our considered values ($\gamma_i = 2.5 \times 10^{-6}$), we see that, whilst the final ask and bid quotes are 101.68 and 98.33, respectively, no convergence has been reached and the prices are still trending upwards. Figures B12 and B13 plot market ask and bid distributions in the final day for the various values of γ_i.



(a) $\gamma_i = 1 \times 10^{-4}$

(b) $\gamma_i = 4 \times 10^{-5}$ (*)

(c) $\gamma_i = 2 \times 10^{-5}$

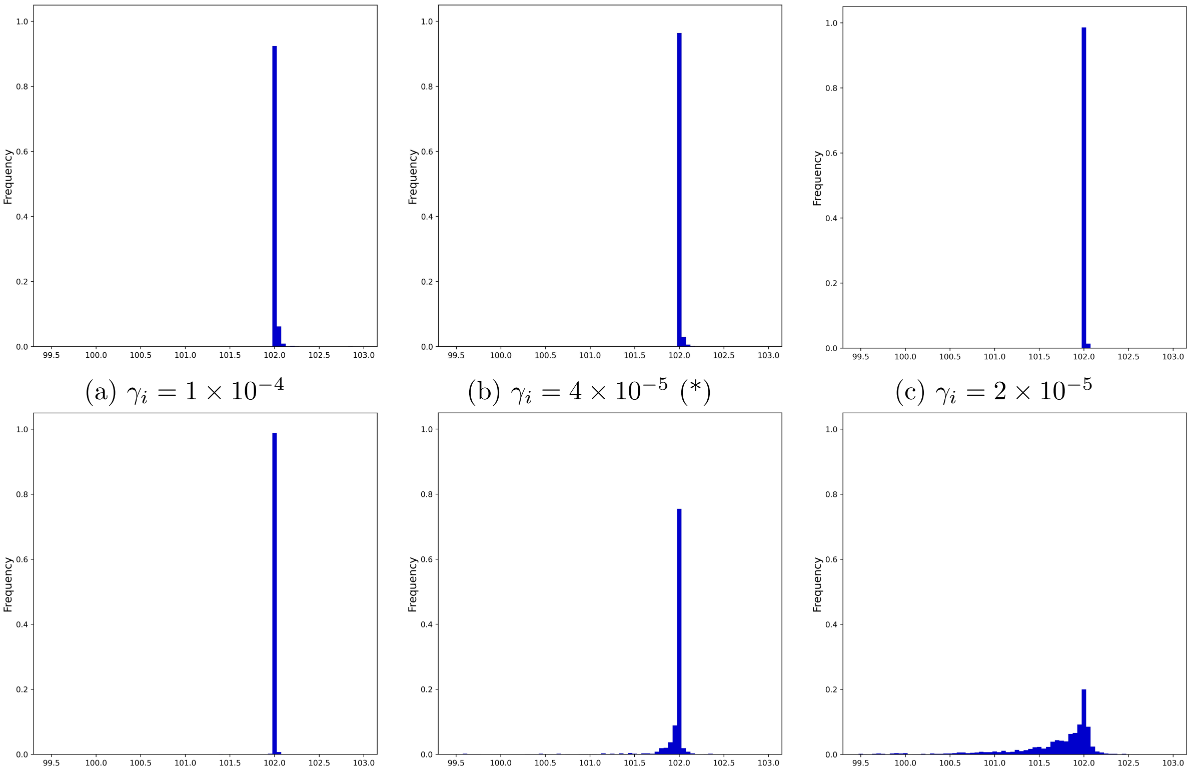
(d) $\gamma_i = 1 \times 10^{-5}$

(e) $\gamma_i = 5 \times 10^{-6}$

(f) $\gamma_i = 2.5 \times 10^{-6}$

Each panel, for a given value of γ_i, plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The "(*)" refers to the baseline parameterization that we considered in Section 6. The mean market ask (bid) quotes in the final day are as follows: $\gamma_i = 1 \times 10^{-4}$ — 102 (98); $\gamma_i = 1 \times 10^{-5}$ — 102 (98); $\gamma_i = 2 \times 10^{-5}$ — 102 (98); $\gamma_i = 4 \times 10^{-5}$ — 102 (98); $\gamma_i = 5 \times 10^{-6}$ — 101.95 (98.05); $\gamma_i = 2.5 \times 10^{-6}$ — 101.68 (98.33).

Figure B11: γ_i Comparative Statics - Mean Market Quotes Over Days



(a) $\gamma_i = 1 \times 10^{-4}$

(b) $\gamma_i = 4 \times 10^{-5}$ (*)

(c) $\gamma_i = 2 \times 10^{-5}$

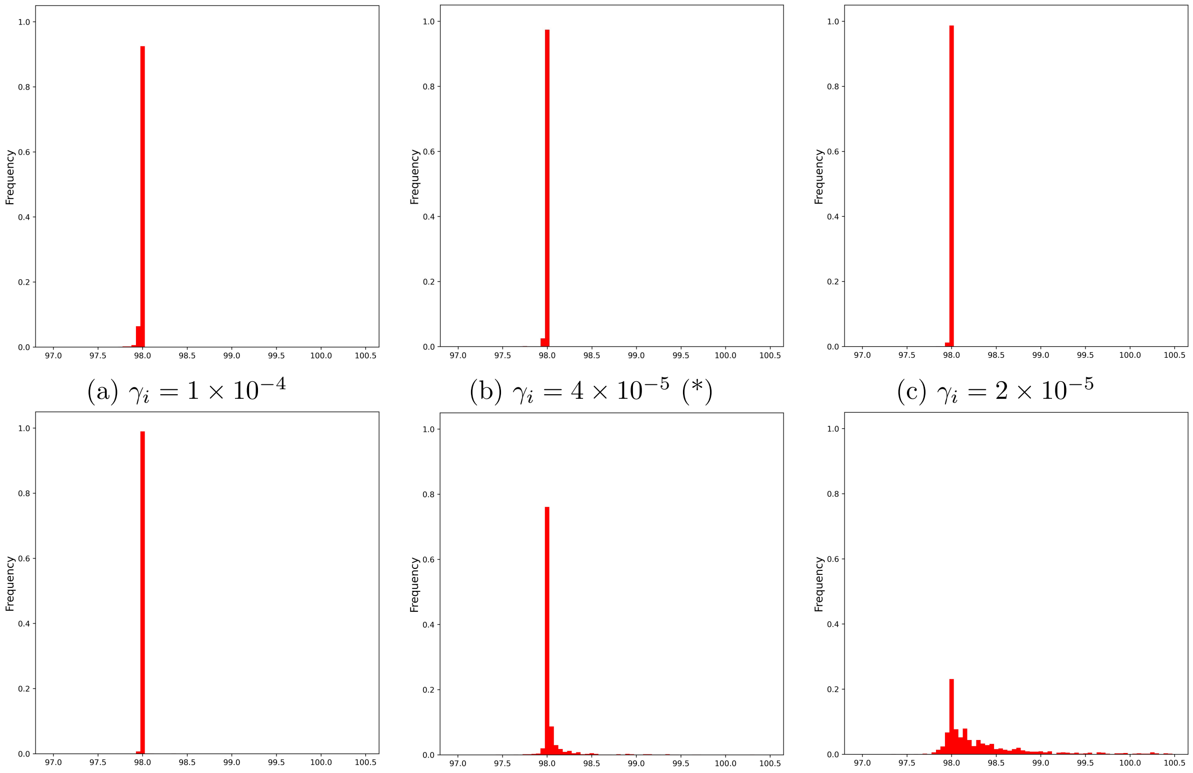
(d) $\gamma_i = 1 \times 10^{-5}$

(e) $\gamma_i = 5 \times 10^{-6}$

(f) $\gamma_i = 2.5 \times 10^{-6}$

Each panel shows the distribution of the market ask quotes across the 1,000 repetitions in the final day, for a given value of γ_i. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6

Figure B12: γ_i Comparative Statics - Distribution of Market Ask Prices in Day $d = 1{,}000{,}000$



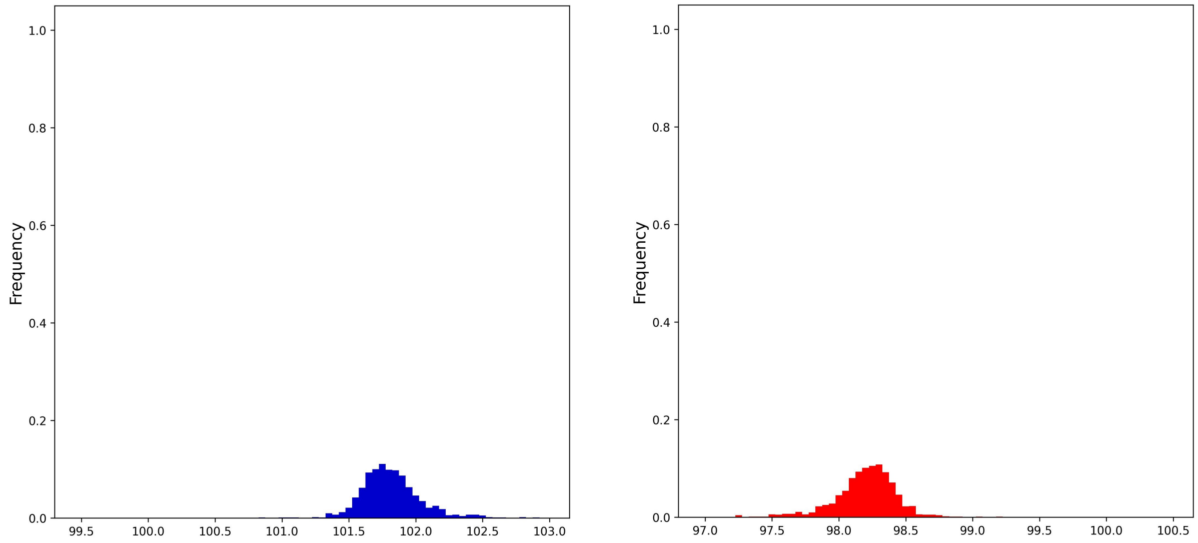
(a) $\gamma_i = 1 \times 10^{-4}$ (b) $\gamma_i = 4 \times 10^{-5}$ (*) (c) $\gamma_i = 2 \times 10^{-5}$

(d) $\gamma_i = 1 \times 10^{-5}$ (e) $\gamma_i = 5 \times 10^{-6}$ (f) $\gamma_i = 2.5 \times 10^{-6}$

Each panel shows the distribution of the market bid quotes across the 1,000 repetitions in the final day, for a given value of γ_i. The graph refers to the baseline Q-learning setup. The "(*)" refers to the baseline parameterization that we considered in Section 6

Figure B13: γ_i Comparative Statics - Distribution of Market Bid Prices in Day $d =$ 1,000,000

B.5 Soft-max Exploration

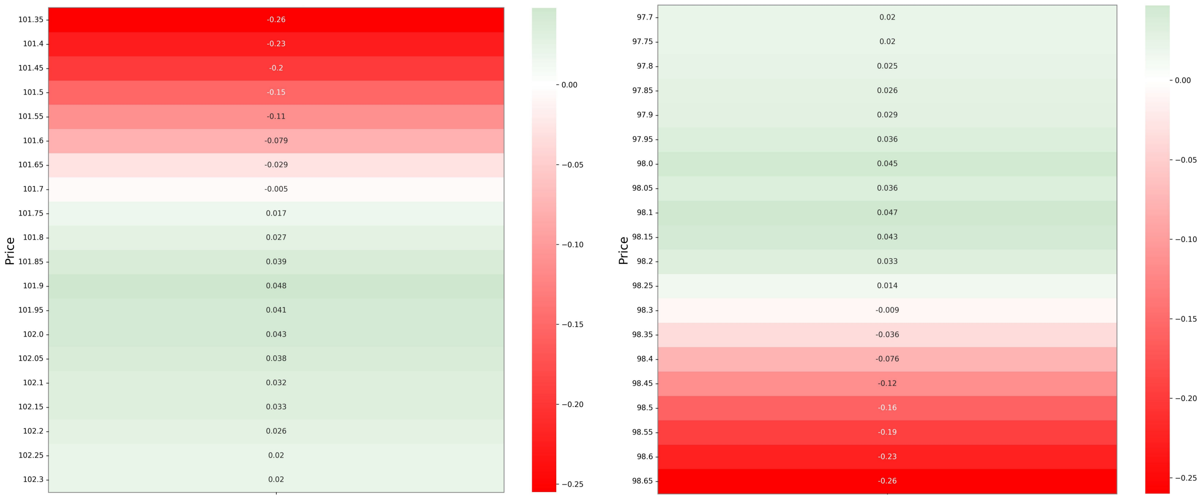


(a) Ask

(b) Bid

The graph shows the distribution of best ask quotes across the 1,000 repetitions in the final day for both the ask and bid quotes. The graph refers to the baseline Q-learning setup.

Figure B14: Soft-max Exploration - Distribution of AMM Quotes in $d = D = 1,000,000$



(a) Ask Q-Table

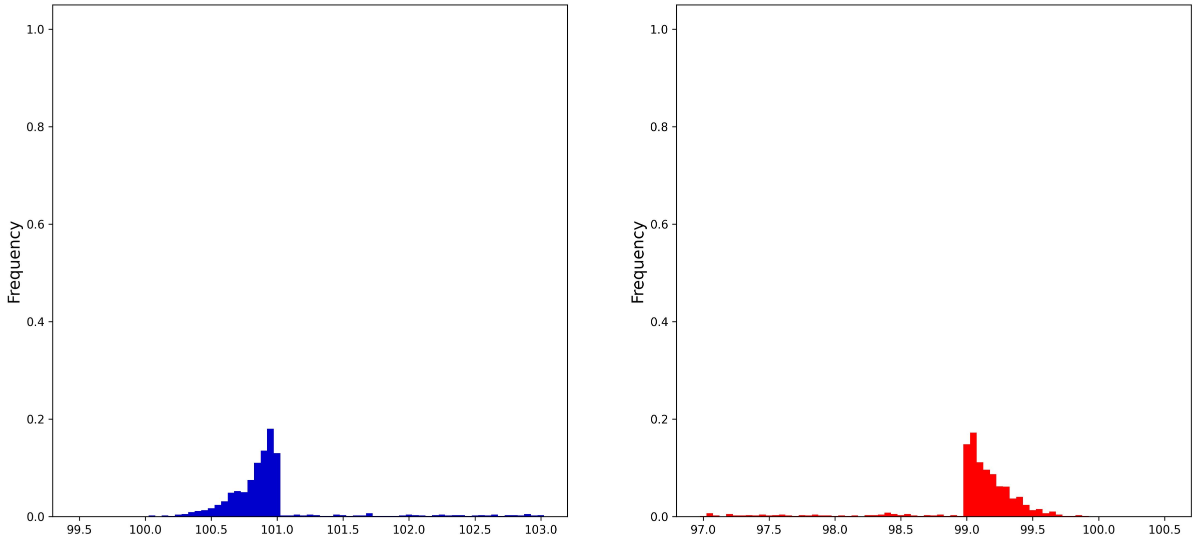
(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure B15: Soft-max Exploration - Final Mean Q-Tables

B.6 Counterfactual Updating: Fixed-price Market Maker

In this section, we consider how a single AMM, now using counterfactual updating, responds to a “human” market maker quoting fixed ask and bid prices of 101 and 99, respectively.

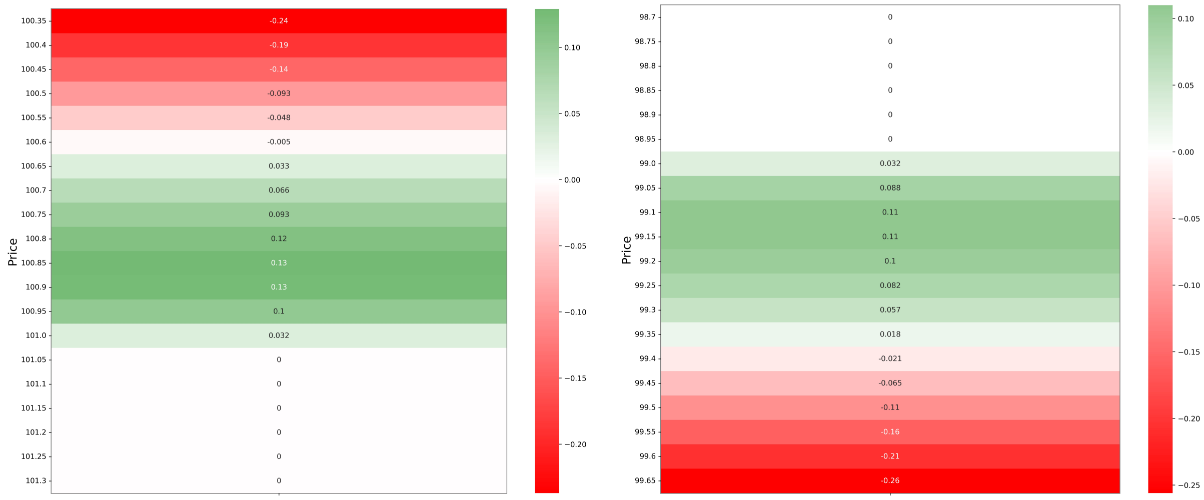


(a) Ask

(b) Bid

The graph shows the distribution of the ask and bid quotes set by the AMM in the final day, across the 1,000 repetitions. The graph refers to the counterfactual-updating Q-learning setup.

Figure B16: Counterfactual Updating, Fixed-price Market Maker - Distribution of AMM Quotes in $d = D = 1,000,000$



(a) Ask Q-Table

(b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the counterfactual-updating Q-learning setup.

Figure B17: Counterfactual Updating, Fixed-price Market Maker - Final Mean Q-Tables

B.7 Decoupled Updating

In this section, we consider a variation of the updating rule in which the Q-values are only updated when there is a trade on that side of the market. That is, the ask Q-values are only updated if the trader buys the asset from the AMM; the bid Q-values are only updated if the trader sells the asset to the AMM.

B.7.1 Standard

If market maker i chooses ask $a_i^d = \alpha_j$ and bid price $b_i^d = \beta_k$, then these are updated as follows:

$$q_i^{d+1}(\alpha_j) = \lambda_i^d \left(\left(\alpha_j - v^d \right) \mathbf{1}_{\{\Phi_i^d = -1\}} \right) + \left(1 - \lambda_i^d \right) q_i^d(\alpha_j), \qquad \text{if } X^d = 1, \tag{36}$$

$$q_i^{d+1}(\alpha_j) = q_i^d(\alpha_j), \qquad \text{otherwise}, \tag{37}$$

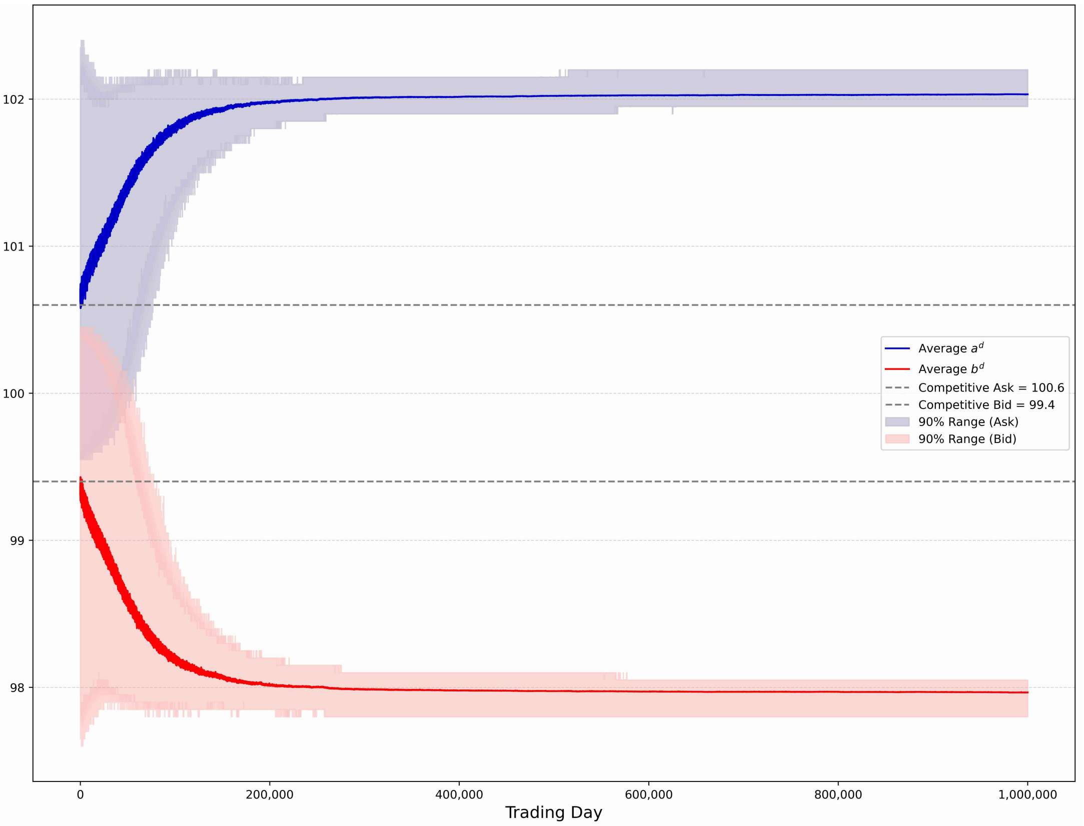
$$\hat{q}_i^{d+1}(\beta_k) = \lambda_i^d \left(\left(v^d - \beta_k \right) \mathbf{1}_{\{\Phi_i^d = 1\}} \right) + \left(1 - \lambda_i^d \right) \hat{q}_i^d(\beta_k), \qquad \text{if } X^d = -1, \tag{38}$$

$$\hat{q}_i^{d+1}(\beta_k) = \hat{q}_i^d(\beta_k), \qquad \text{otherwise.} \tag{39}$$

For all other (i.e., unchosen) ask prices α_l $(l \neq j)$ and bid prices β_m $(m \neq k)$, the Q-values are unchanged:

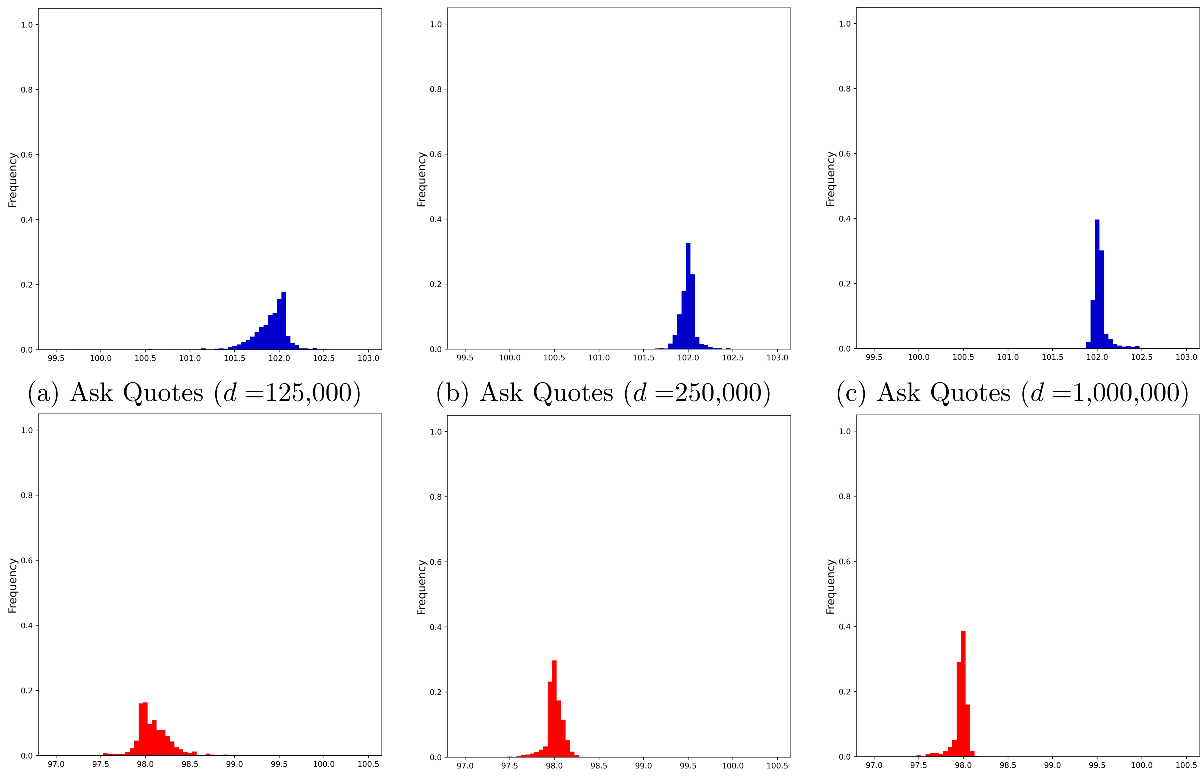
$$q_i^{d+1}(\alpha_l) = q_i^d(\alpha_l), \tag{40}$$

$$\hat{q}_i^{d+1}(\beta_m) = \hat{q}_i^d(\beta_m). \tag{41}$$



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the baseline Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 102.03 and 97.97, respectively.

Figure B18: Decoupled, Baseline - Mean Market Quotes Over Days

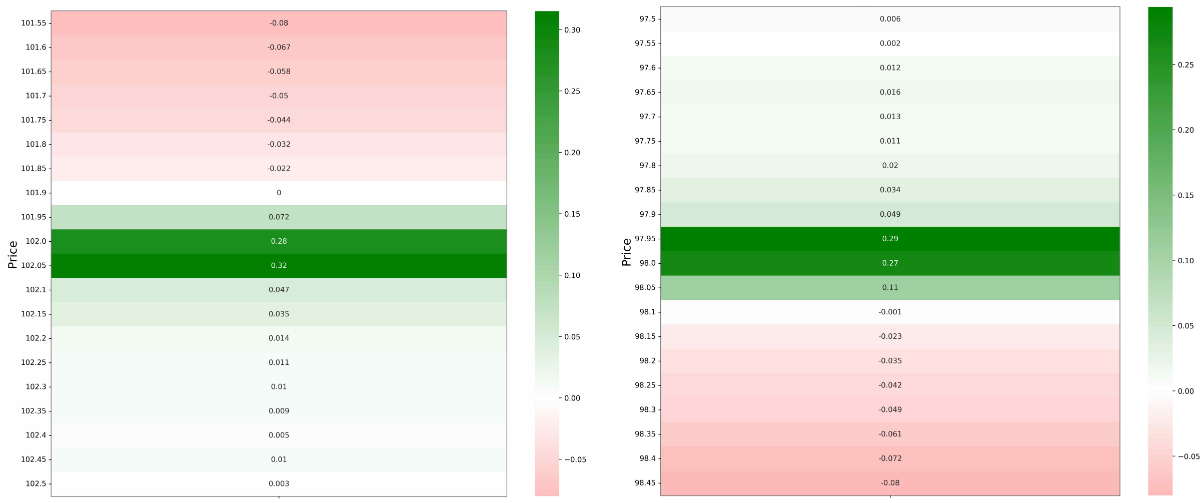


(a) Ask Quotes (d =125,000) (b) Ask Quotes (d =250,000) (c) Ask Quotes (d =1,000,000)

(d) Bid Quotes (d =125,000) (e) Bid Quotes (d =250,000) (f) Bid Quotes (d =1,000,000)

The graph plots the distributions of market ask and bid prices across the 1,000 repetitions at different points in the simulation: days 125,000, 250,000, and 1,000,000. The graph refers to the baseline Q-learning setup.

Figure B19: Decoupled, Baseline – Distributions of Market Ask and Bid Quotes over Time



(a) Ask Q-Table (b) Bid Q-Table

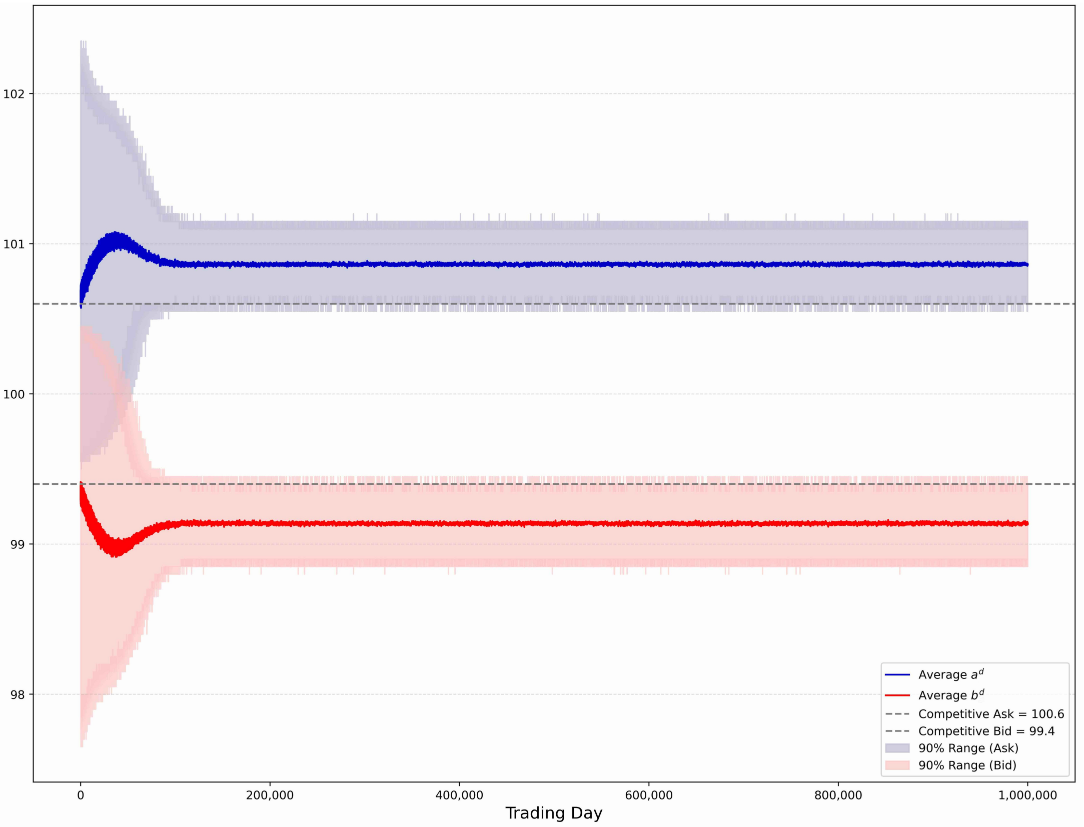
The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the baseline Q-learning setup.

Figure B20: Decoupled, Baseline - Final Mean Q-Tables

B.7.2 Counterfactual Updating

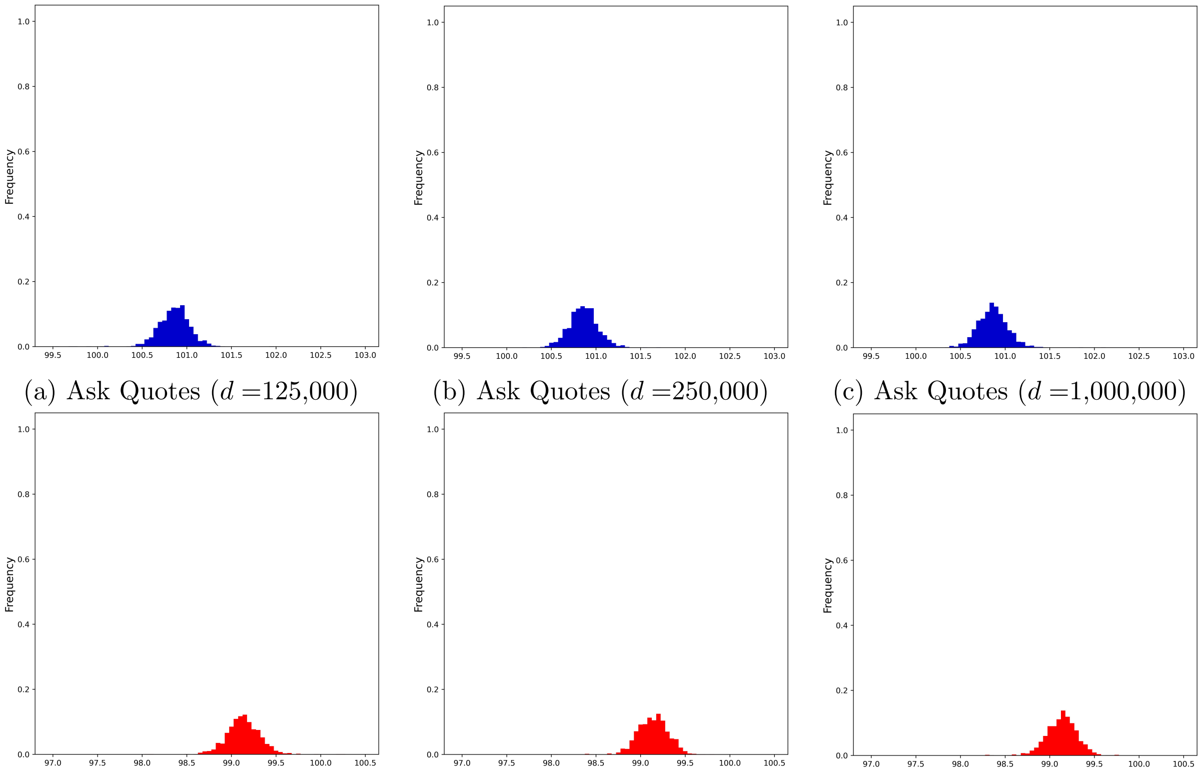
The counterfactual updating in the decoupled version follows the same logic as described in Section 8, but with the same conditions on the updating as described in B.7.1. The ask Q-values are only updated when $X^d = 1$ and the bid Q-values are only updated when $X^d = -1$, otherwise the Q-values on that side of the market are left unchanged.

Below we present the results for the counterfactual weighting of $\rho = 1$.



The graph plots the mean market quotes over days, averaging across the 1,000 repetitions. The graph refers to the counterfactual-updating Q-learning setup. The mean ask (bid) is the dark blue (red) line; the blue and red shaded bands represent the 90% range across the simulations for the ask and bid, respectively. The dashed lines plot the competitive ask and bid prices, as discussed in Section 3. The mean market ask and bid quotes in the final day are 100.86 and 99.13, respectively.

Figure B21: Decoupled, Counterfactual Updating - Mean Market Quotes Over Days

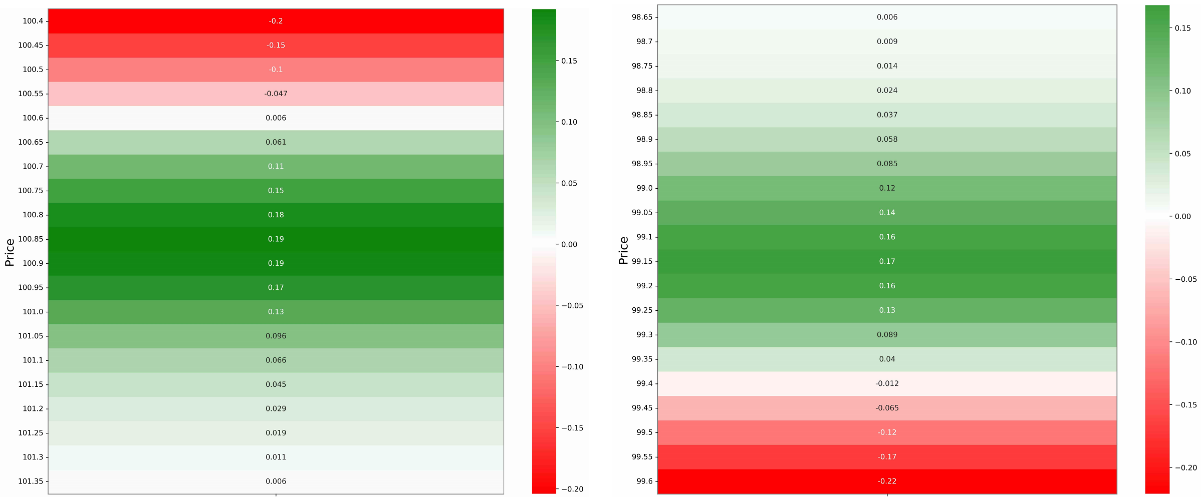


(a) Ask Quotes (d =125,000) (b) Ask Quotes (d =250,000) (c) Ask Quotes (d =1,000,000)

(d) Bid Quotes (d =125,000) (e) Bid Quotes (d =250,000) (f) Bid Quotes (d =1,000,000)

The graph plots the distributions of market ask and bid prices across the 1,000 repetitions at different points in the simulation: days 125,000, 250,000, and 1,000,000. The graph refers to the counterfactual-updating Q-learning setup.

Figure B22: Decoupled, Counterfactual Updating – Distributions of Market Ask and Bid Quotes over Time



(a) Ask Q-Table (b) Bid Q-Table

The graph plots the mean ask and bid Q-tables at the end of the simulations, averaging across the 1,000 repetitions. It shows the Q-tables for only a subset of prices, centered around the final mean prices. The graph refers to the counterfactual-updating Q-learning setup.

Figure B23: Decoupled, Counterfactual Updating - Final Mean Q-Tables