

AI driven liquidity provision in OTC financial markets

Álvaro Cartea^{a,b} Patrick Chang^a Mateusz Mroczka^b Roel Oomen^{c,d,*}

^a*Oxford-Man Institute of Quantitative Finance*

^b*Mathematical Institute, University of Oxford*

^c*Deutsche Bank, London*

^d*Department of Statistics, London School of Economics*

May 2022

Abstract

Providing liquidity in over-the-counter markets is a challenging under-taking, in large part because a market maker does not observe where their competitors quote, nor do they typically know how many rivals they compete with or what the trader's overall liquidity demand is. Optimal pricing strategies can be derived in theory assuming full knowledge of the competitive environment, but these results do not translate into practice where information is incomplete and asymmetric. This paper studies whether artificial intelligence, in the form of multi-armed bandit reinforcement learning algorithms, can be used by liquidity providers to dynamically set spreads using only information that is commonly available to them. We also investigate whether collusive effects can arise when competing liquidity providers all employ such algorithms. Our findings are as follows. In a single-agent setup where only one liquidity provider is optimising pricing in an otherwise static environment, all the algorithms considered are able to locate the theoretically optimal pricing policy, albeit they do so quite inefficiently when compared to a model-based approach. In a multi-agent setting where competing liquidity providers simultaneously and independently use algorithms to optimise pricing, we demonstrate that for one class of algorithms (pseudo) collusion can not arise, while for another it can theoretically arise in certain circumstances and we provide examples where it does. The scenarios where collusive effects appear, however, are fragile and sensitive to the specific configuration and exceedingly unlikely to occur in practice. Moreover, with a modest number of competitors, collusive effects that might otherwise arise in some of the most contrived scenarios are largely or entirely eliminated.

* Corresponding author: roel.oomen@db.com. Roel Oomen is employed as a Managing Director of FIC quantitative trading at Deutsche Bank A.G. This paper was prepared within the Sales and Trading function of DB, and was not produced, reviewed or edited by the DB Research Department. The views and opinions rendered in this paper reflect the author's personal views about the subject. No part of the author's compensation was, is, or will be directly related to the views expressed in this paper. See [Disclaimer](#) at end of paper.

1 Introduction

Many of the world's largest financial markets¹ trade in an over-the-counter (OTC) fashion where market makers or liquidity providers (LPs) maintain bi-lateral and private trading relationships with those whose flow they compete for. In optimising their liquidity offering, one of the biggest challenges the LPs face is to navigate a competitive environment characterised by incomplete and asymmetric information. First and foremost, the LPs do not observe the prices of their competitors. They typically do not even know how many LPs they are in competition with. Nor do they know the total liquidity demand of the trader as they only observe the trades they win but not those that are executed with their rivals. Meanwhile, advances in Artificial Intelligence (AI) hold promise to facilitate and improve data-driven decision making in an increasingly digital world. The benefits of AI are numerous and widely recognised, but so are the challenges that come with these often complex and intractable methods with potential opacity at how they arrive at decisions and unpredictable or unintended behaviours arising from interactions with other AIs. For instance, the [Competition & Markets Authority \(2018, 2021\)](#), the [OECD \(2017\)](#), the [European Commission \(2017\)](#), and the Swedish Competition Authority ([Löfström, Ralsmark, and Johansson, 2021](#)) all envisage the risk of independently operated AIs enabling tacit collusion.² Against this background, this paper sets out to provide an in-depth analysis of some established AI methods – i.e. the multi-armed bandit (MAB) reinforcement learning algorithms – when used by LPs to dynamically set pricing in an OTC financial market and whether this can lead to collusive outcomes. While collusion in multi-agent environments is a quickly emerging field of study, to the best of our knowledge this is one of the first studies to focus on financial markets and the first one that considers an OTC market structure.³

Our main findings are as follows. In a single-agent setting where the LP is dynamically optimising pricing in an otherwise static competitive environment, the model-free MAB algorithms are able to identify the optimal pricing strategy but are rather inefficient and slow when compared to a model-based maximum likelihood approach. In a multi-agent setting where LPs are simultaneously but independently optimising their pricing with the use of MAB algorithms, we can mathematically prove for one class of algorithms pseudo collusion (as defined in section 3.3) can not arise whereas for another class of algorithms this cannot be guaranteed and we provide examples where

¹Including the \$6.6tr/day foreign exchange market ([BIS, 2019a](#)), the \$6.5tr/day interest rate derivatives market ([BIS, 2019b](#)), or the \$593bn/day US Treasury and \$34bn/day US corporate credit markets ([SIFMA, 2021](#), page 26). This compares to a largely on-exchange US equity market of \$322bn/day ([SIFMA, 2021](#), page 29). All figures are for 2019.

²Other risks include predatory price differentiation or personalisation, built-in biases and discriminatory effects, and exclusionary practices. See also [The White House \(2015\)](#). Examples where cooperation between AIs can be desirable are discussed in [Dafoe, Hughes, Bachrach, Collins, McKee, Leibo, Larson, and Graepel \(2020\)](#).

³Concurrent papers on the topic are [Xiong and Cont \(2021\)](#) and [Cartea, Chang, and Penalva \(2022\)](#).

independently operated algorithms can give rise to pseudo collusive pricing. The scenarios where this occurs are, however, quite fragile to the specific setup. For instance, collusive effects can only be found for a specific class of algorithms, where all LPs would need to use that same algorithm, with similar configuration and starting point, the number of LPs would need to be very limited, and all this would need to be sustained over extended periods of time to allow the algorithms to converge. In practice – absent an explicit form of collusion to agree to such a specific and coordinated algorithmic setup amongst competitors – this is exceedingly unlikely to occur.

To study the properties of the MAB algorithms when applied to pricing in OTC markets, we adopt the model of [Oomen \(2017a\)](#) because it captures the key characteristics of the trading environment and the incomplete and asymmetric information discussed above. The model assumes there are multiple LPs competing for a trader's order flow. The true price of the asset is unobservable, so both the LPs and the trader use an estimate to determine the price at which they are willing to deal. The trader has a reservation price which needs to be met, but is otherwise uninformed and their liquidity demand is random. They are assumed to trade with the LP that shows the best price. The key decision for the LPs is what spread to charge while being blind to where competitors are pricing. We derive closed form solutions for the spread that LPs charge in a competitive equilibrium as well as the spread that can be sustained in a monopolistic equilibrium. These provide the reference points against which we assess the efficiency of the spreads arrived at by the MAB algorithms. With each trade that the LP wins comes an associated stochastic reward in the form of an effective⁴ spread earned on the deal. If the LP does not win the trade, their reward will be zero. The pairing of action (i.e. selected spread) and resulting reward forms the input to the MAB algorithm, based upon which it determines the next suggested action. The MAB algorithms we consider in this paper are the ϵ -greedy, the EXP3, and the UCB algorithms. Common to all is a fundamental trade-off they seek to balance between – what is often referred to as – “exploration” and “exploitation”. To establish the expected reward associated with each action, the algorithm would want to explore by selecting each candidate spread repeatedly in order to build up an increasingly accurate estimate of the reward mapping. However, each exploration comes with the opportunity cost of not exploiting the seemingly optimal spread, albeit one that is based on an inaccurate and perhaps incorrect estimate of the true expected reward. The MAB algorithms considered here are differentiated by the precise mechanics of how they handle this trade-off.

In a single-agent setting, we find that – given a sufficient amount of trading time – all the MAB algorithms are able to identify the optimal spread to charge or arm to select (we will use “arm” and “spread” interchangeably throughout this paper). While re-assuring, this is also unsurprising in light of well know theoretical results that prove convergence to the optimal arm in a stationary setup (see, e.g., [Auer, Cesa-Bianchi, Freund, and Schapire, 2002](#); [Auer,](#)

⁴Because the true price is unobservable and competitor prices unknown to the LP, the winning trades are subject to adverse selection effects also known as the Winner's curse. The nominal spread charged is therefore not equal to the actual spread that the LP expects to earn.

[Cesa-Bianchi, and Fischer, 2002](#)). We show that the number of arms and the range of values they cover can greatly affect the speed of convergence. While one may be tempted to select many candidate spreads across a wide range of values to ensure it contains the unknown optimal spread, such an approach could come with an unacceptably slow convergence rate. This is one of the trade-offs that needs to be balanced in practice.

One of the key advantages of the MAB algorithms is that they are model-free and therefore apply generically and without modification to any setting where actions are controllable and rewards observed. But this also comes with the downside that they do not incorporate environment specific knowledge which could potentially improve performance. For instance, the trading environment we simulate from has a specific structure, a continuous value function, and it makes distributional assumptions on the noise components. In order to assess the efficiency that is sacrificed with use of these model-free MAB algorithms in exchange for robustness to model misspecification, we derive the maximum likelihood estimator (MLE) for the parameters of our OTC market model. When assessed against this benchmark – which is known to be the most efficient estimator under correct model specification – the MAB algorithms are shown to be very inefficient with a convergence rate to select the optimal arm that is at least an order of magnitude slower than that of the MLE.

In a multi-agent setting, where prices are simultaneously but independently optimised by competitors with each using their own AI algorithm, our primary interest is whether collusive effects can arise. We contribute here to a rapidly emerging literature on the topic.⁵ [Calvano, Calzolari, Denicolò, and Pastorello \(2020\)](#) and [Klein \(2021\)](#) show that when firms independently use Q-learning to set prices, the algorithms can identify strategies that sustain collusion via the threat of a price war when rivals deviate from the collusive state. [Brown and MacKay \(2021\)](#) identifies asymmetries in the speed of pricing as another mechanism through which collusive prices can be supported. These papers assume, however, that rivals' prices are observable which is incompatible with an OTC market structure. Papers that do not make this assumption but still find evidence of collusive effects include [Abada \(2022\)](#) through imperfect exploration, [Aouad and den Boer \(2021\)](#) through targeted algorithm design choices, [Asker, Fershtman, and Pakes \(2021\)](#) through asynchronous Q-learning, and [Hansen, Misra, and Pai \(2021\)](#) through misspecification and correlated exploration cycles. In parallel to the economics literature on the topic, there is an interesting legal debate ongoing regarding the enforceability of current competition law and whether it needs to be adapted to deal with any potential algorithmic collusion, see for instance [Ezrachi and Stucke \(2017\)](#); [Harrington \(2018\)](#); [Schwalbe \(2018\)](#).

On collusive effects arising within an OTC market environment, our results can be summarised as follows. For the EXP3 algorithm, we can show that its pricing policy in a multi-agent setting recovers the replicator dynamics

⁵See [Dorner \(2021\)](#) for an exhaustive and recent overview of the literature.

when the learning rate goes to zero with time. This fully characterises in analytical form the limiting behaviour of the EXP3 algorithm and we use this to prove that in the case of two LPs, the algorithm converges to a pure strategy Nash equilibrium of a one-shot game. We numerically verify that this also holds for the case of more than two LPs. Because the equilibria reached by the EXP3 algorithm are uni-laterally optimal meaning that LPs do not have an incentive to deviate statically we conclude that the EXP3 algorithm is free of pseudo collusion in a multi-agent setting. However, we also show that supra-competitive spread levels can be sustained. Because the MAB algorithms can only search over a finite number of arms, the action state space needs to be discretised. It is this discretisation that can introduce additional Nash equilibria into our setting where no LP has an individual incentive to deviate from charging supra-competitive spreads. Intuitively, the coarser the grid that candidate spreads lie on, the more expensive it becomes for an LP to under-cut the price of their competitor. And so even if that competitor was to charge a spread in excess of the competitive equilibrium value, situations can arise where it is uni-laterally optimal for the LP to charge that same spread when the alternative of under-cutting by a discrete amount would sacrifice more spread capture than what would be gained by an increase in market share. So while the EXP3 algorithm can converge to spread levels higher than the competitive equilibrium, we show that they correspond to Nash equilibria that arise as an artefact of the unavoidable need to discretise the state space. As the number of arms grows, and the candidate spreads lie on an increasingly fine (or asymmetric) grid, the pricing efficiency improves and rapidly converges to the competitive equilibrium derived under continuous pricing.

Turning to the UCB algorithms, we show that in a multi-agent setting, scenarios can arise where supra-competitive spreads can be sustained that are – and crucially different from the EXP3 results – statically and uni-laterally sub-optimal for the individual LPs. We refer to this as pseudo collusion.⁶ While each individual LP could increase their instantaneous expected revenues by undercutting their rivals, the UCB algorithms are not tempted by this and instead continue to quote at levels that prioritise the competitors' collective interests. The scenarios where pseudo collusion arises, however, require specific and often unrealistic model configurations. They are also fragile to the overall environment. For instance, when different LPs use different MAB algorithms to set prices, we find that any pseudo collusion that may otherwise arise amongst competing UCB algorithms largely or completely vanishes. This is indicative of highly non-trivial interactions between different classes of MAB algorithms, further study of which we will defer to another paper. Moreover, if an LP was to restart their UCB algorithm, or a trader would swap out

⁶We introduce this term to avoid conflation or confusion with tacit collusion. The latter typically embodies a punitive retaliation mechanism designed to prioritise the collective competitors' interest by adhering to the collusive strategy over the uni-lateral temptation to deviate. Because for many "black box" AI algorithms it will be unclear whether or not such a mechanism is present, we adopt the term pseudo collusion which is defined by the properties of the outcome rather than the mechanism to get there. Informally, one can think of pseudo collusion as tacit collusion except that the requirement for a retaliation mechanism to be present is dropped.

one existing LP for a new one, again the pseudo collusion that would otherwise occur is largely mitigated. Also, increasing the number of competitors leads to an improved pricing efficiency that rapidly approaches the competitive equilibrium. Even for the most extreme and contrived model parameterisations, any potential collusive effects are largely or entirely eliminated when the trader places only a handful of LPs in competition. In related work (Oomen, 2017a,b; Butz and Oomen, 2019) the authors argue for a limited number of LPs to be placed in competition in order to strike a balance between encouraging competition on the one hand, while avoiding effects like excessive externalisation via a Prisoner's dilemma mechanism that increase execution costs. This informal rule of thumb is now strengthened further by the argument that it also protects against any potential collusive effects that may arise due to use of AI trading technology.

The remainder of the paper is structured as follows. Section 2 introduces the OTC market model and presents closed form solutions for the equilibrium spreads and trade valuation function. Section 3 introduces the MAB algorithms, followed by an assessment of their performance in a single-agent setting, including a comparison to a model-based maximum likelihood approach. It also includes a few examples of multi-agent learning to provide the necessary context for the EXP3 replicator dynamics analysis in Section 4. Section 5 concentrates on the UCB algorithms where examples of pseudo collusion are presented. Section 6 concludes.

2 The OTC Market Model

Let $N \geq 2$ denote the number of liquidity providers that compete for a trader's order flow. At time t , the i^{th} LP shows a bid price (b) and an ask price (a) at which they are willing to buy and sell one unit of the traded asset:

$$b_t^{(i)}(s_i) = p_t^{(i)} - \frac{s_i}{2}, \quad \text{and} \quad a_t^{(i)}(s_i) = p_t^{(i)} + \frac{s_i}{2}, \quad \text{for } i \in \{1, \dots, N\}. \quad (1)$$

Here, $s_i \geq 0$ denotes the LP's quoted spread that is centred around their estimate $p_t^{(i)}$ of the unobserved true price p_t^* , where

$$p_t^{(i)} = p_t^* + m_t^{(i)}, \quad \text{and} \quad p_t^* = p_{t-1}^* + \varepsilon_t, \quad (2)$$

and $m^{(i)} \sim \text{i.i.d. } \mathcal{N}(0, \omega^2)$, $\text{corr}(m^{(i)}, m^{(j)}) = \rho$ for $i \neq j$, $\varepsilon \sim \text{i.i.d. } \mathcal{N}(0, \sigma^2)$.

The trader is assumed to have exogenously driven liquidity demand (D), specified as follows:

$$D_t = \begin{cases} 1, & \text{the trader has demand to buy one unit of the asset,} \\ 0, & \text{the trader has no demand to buy or sell,} \\ -1, & \text{the trader has demand to sell one unit of the asset,} \end{cases} \quad (3)$$

where $D_t = A_t(2B_t - 1)$, $A_t \sim \text{i.i.d. Bin}(1, \delta)$, $B_t \sim \text{i.i.d. Bin}(1, \delta_B)$. So with probability δ the trader has demand for liquidity, they buy with probability δ_B and sell with probability $1 - \delta_B$. The trader only acts on this demand and initiates a trade to buy from (sell to) the LP showing the best ask (bid) price if this improves over their reservation ask (bid) price, i.e. $\min_i a_t^{(i)}(s_i) < a_t^{(T)}(s_T)$ ($\max_i b_t^{(i)}(s_i) > b_t^{(T)}(s_T)$) where

$$b_t^{(T)}(s_T) = p_t^{(T)} - \frac{s_T}{2}, \quad \text{and} \quad a_t^{(T)}(s_T) = p_t^{(T)} + \frac{s_T}{2}. \quad (4)$$

Here, the trader's reservation price s_T determines their appetite to trade at levels that deviate from their estimate $p_t^{(T)}$ of the unobserved true price p_t^* , where

$$p_t^{(T)} = p_t^* + m_t^{(T)}, \quad (5)$$

and $m^{(T)} \sim \text{i.i.d. } \mathcal{N}(0, \omega_T^2)$. To complete the required notation, and analogous to D_t above, we also define $D_t^{(i)}$ as an indicator variable that denotes whether LP- i at time t receives interest to trade on their quoted prices, i.e.

$$D_t^{(i)}(s) = \begin{cases} 1, & \text{when } D_t = 1 \text{ and } a_t^{(i)}(s_i) < \min_{j \neq i} a_t^{(j)}(s_j) \text{ and } a_t^{(i)}(s_i) < a_t^{(T)}(s_T), \\ 0, & \text{otherwise,} \\ -1, & \text{when } D_t = -1 \text{ and } b_t^{(i)}(s_i) > \max_{j \neq i} b_t^{(j)}(s_j) \text{ and } b_t^{(i)}(s_i) > b_t^{(T)}(s_T). \end{cases} \quad (6)$$

The notation makes it explicit that this variable depends not only on the spread charged by LP- i but on the vector of spreads charged by all the respective LPs, i.e. $s \equiv (s_1, \dots, s_N)$.

The model above closely follows [Oomen \(2017a\)](#) and it captures several key features of an over-the-counter trading environment, namely (a) the trader sources liquidity on a bi-lateral basis from a panel of competing LPs (as opposed to trading in a central limit order book that is visible and accessible to all market participants), (b) the LPs do not observe the prices and liquidity that competing LPs show to the trader, and typically do not know how many LPs they are in competition with (i.e. N is determined by the trader and not disclosed), (c) executed trades are only known to the trader and the LP who wins that trade.⁷ The key challenge for the LP is to determine what spread to charge, but given the incomplete and asymmetric information of the environment they operate in, this is a highly non-trivial task. For instance, when an LP finds that they do not win many/any trades for a period of time, it could be a sign that they should tighten their spread, but equally it may be because the trader does not have much liquidity demand (i.e. δ is small), or that there are many competing LPs (i.e. large N), or because the trader's reservation price is restrictive (i.e. small or negative s_T). Before we turn to how the MAB algorithms can be used by

⁷Two features we abstract away from in this paper is the last look trade acceptance process (see, e.g., [Cartea, Jaimungal, and Walton, 2018](#); [Oomen, 2017b](#)) and the LPs' risk management of trade flows via internalisation and externalisation (see, e.g., [Barzykin, Bergault, and Guéant, 2021](#); [Butz and Oomen, 2019](#)).

the LPs to set spreads, we first analyse some properties of the model and present equilibrium results that serve as a reference point throughout the paper.

The i^{th} LP's effective spread capture or "reward" they receive is given by

$$\pi_t^{(i)}(s) = (a_t^{(i)}(s_i) - p_t^*) \mathbb{1}_{D_t^{(i)}(s)=1} + (p_t^* - b_t^{(i)}(s_i)) \mathbb{1}_{D_t^{(i)}(s)=-1} = \frac{1}{2} s_i |D_t^{(i)}(s)| + m_t^{(i)} D_t^{(i)}(s). \quad (7)$$

Lemma 1 (Trade valuation) *The expected valuation of the trader's flow won by LP- i is given by*

$$\mathbb{V}_i(s) \equiv \mathbb{E}[\pi_t^{(i)}(s)] = \delta \left(\frac{s_i}{2} \theta_i^{(1)}(s) - \omega \sqrt{1-\rho} \mu_i^{(1)}(s) \right), \quad (8)$$

where

$$\theta_i^{(k)}(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} y^{k-1} f_i(x, y, s) \phi(x) \phi(y) dx dy, \quad (9)$$

$$\mu_i^{(k)}(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} y^{k-1} \left(y + \sqrt{\frac{\rho}{1-\rho}} x \right) f_i(x, y, s) \phi(x) \phi(y) dx dy, \quad (10)$$

$$f_i(x, y, s) = \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s_i}{2\omega_T} \right) \prod_{j \neq i}^N \Phi \left(y + \frac{s_j - s_i}{2\omega \sqrt{1-\rho}} \right), \quad (11)$$

for $k \in \mathbb{N}$.

Proof See Appendix D.

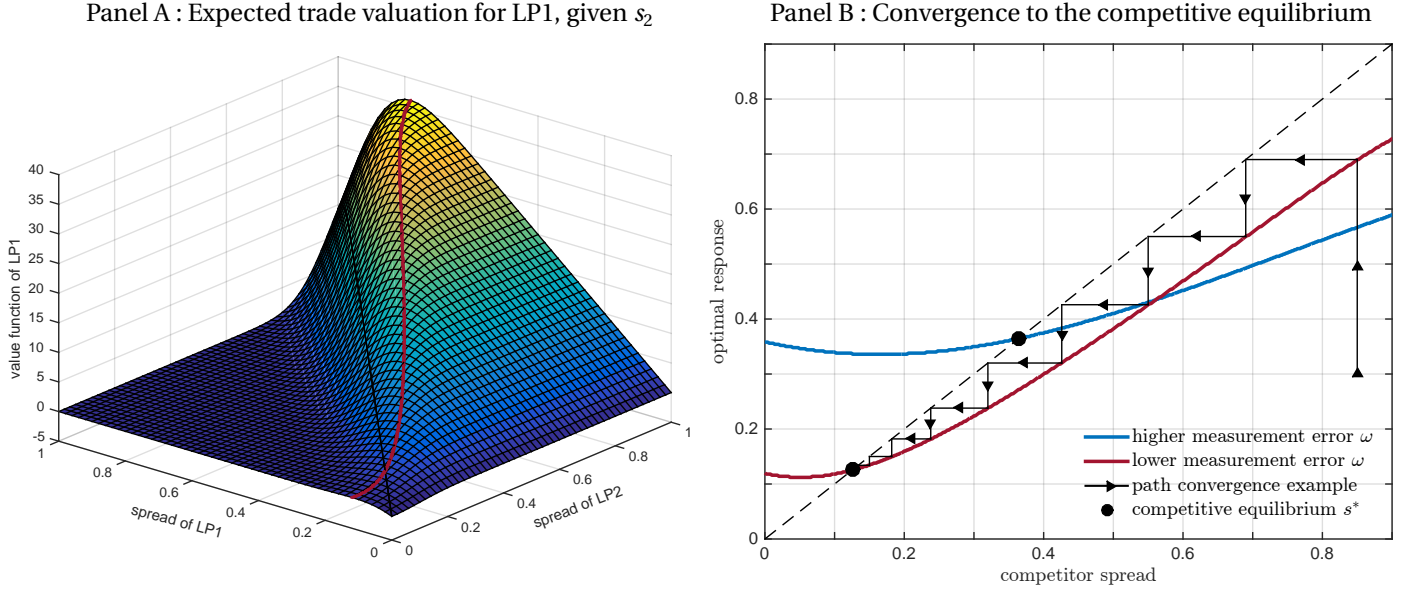
Eq. (8) shows that the trade valuation breaks down into two components. One, the LP's spread capture $s_i/2$ on won trades, weighted by the probability $\theta_i^{(1)}(s)$ that they win the trade given – amongst other factors – their own and their competitors' spreads s as well as the trader's reservation price s_T . The other, an adverse selection or "winner's curse" component which arises due to the unobservability of the prevailing true price and the trader executing on best price (see Oomen, 2017a,b, for further discussion).

The LPs seek to set spreads that maximise their expected trade flow valuations. A symmetric competitive equilibrium is reached when the quoted spread satisfies

$$s^* = \arg \max_{s_i \geq 0} \mathbb{V}_i(s_i, s^*) \quad (12)$$

where $\mathbb{V}_i(s_i, s_{\neq i})$ is a short-hand notation for the expected trade valuation of LP- i when they quote a spread of s_i and their competitors quote a spread of $s_{\neq i}$, i.e. $\mathbb{V}_i(s)$ where the i^{th} element of s equals s_i and all other elements are equal to $s_{\neq i}$.

Figure 1: Illustration of the OTC market model equilibrium



Note. Panel A draws the surface of LP1's expected trade valuation $\mathbb{V}_1(s_1, s_2)$ ($\times 100$) as a function of their own and competitor spread. The superimposed red line draws LP1's optimal spread choice, given s_2 . The black line is where both LPs quote the same spread. The OTC market model parameters are set to $N = 2$, $\rho = 0.5$, $\omega = 0.05$, $s_T = 1$, $\delta = 1$, $\delta_B = 0.5$. Panel B draws the optimal response curve given a competitors spread for two model parameterisations, i.e. lower and higher measurement error where $\omega = 0.05$ and $\omega = 0.15$ respectively (with other parameters set as in Panel A). Convergence to the equilibrium spread through the iterative process where LPs take turns in setting their optimal spread is illustrated by the black arrowed line.

Theorem 1 (Competitive equilibrium spread) *Let $\omega\sqrt{1-\rho} > 0$ and $\omega_T > 0$. A symmetric competitive equilibrium spread satisfies the fixed point problem*

$$s^* = 2\omega\sqrt{1-\rho} \frac{\mu^{(2)}(s^*)}{\theta^{(2)}(s^*)}. \quad (13)$$

Here, $\theta^{(k)}(s) \equiv \theta_i^{(k)}(\iota_N s)$, $\mu^{(k)}(s) \equiv \mu_i^{(k)}(\iota_N s)$ where ι_N is an $N \times 1$ vector of ones.

Proof See Appendix D.

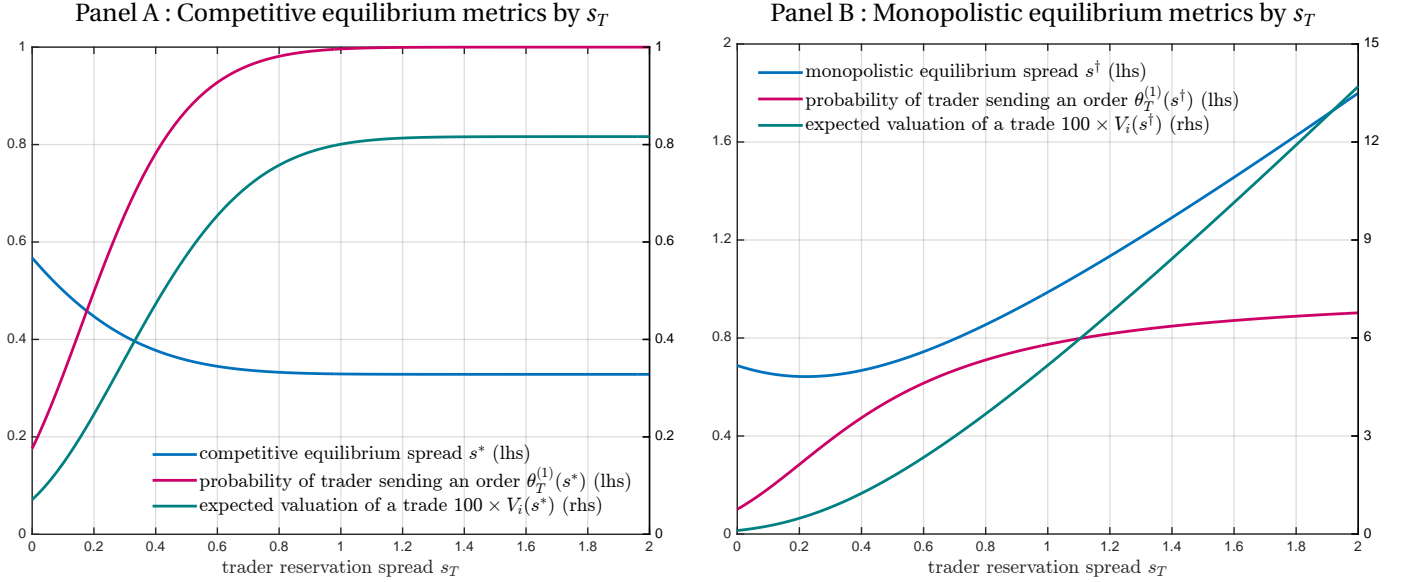
In equilibrium, where all LPs quote the same spread s^* , their probability of winning a trade is equalised at

$$\theta_i^{(1)}(s^*) = \frac{1}{N} \theta_T^{(1)}(s^*), \quad (14)$$

where $\theta_T^{(1)}$ is given by Eq. (36) and denotes the probability that the trader engages any LP at a given point in time. The equilibrium trade valuation is equal to

$$\mathbb{V}_i(s^*) = \delta \left(\frac{s^*}{2N} \theta_T^{(1)}(s^*) - \omega\sqrt{1-\rho} \mu^{(1)}(s^*) \right). \quad (15)$$

Figure 2: Illustration of the OTC market model equilibrium



Note. Panels A and B draw, as a function of the trader's reservation spread s_T , the competitive (s^*) and monopolistic (s^\dagger) equilibrium spreads as in Eqs. (13) and (18) respectively, along with the probability of the LPs' quotes meeting the trader's reservation price (i.e. $\theta_T^{(1)}(s)$ as in Eq. 36), and the LP's expected value of a trade scaled by one hundred (i.e. $100 \times V_i(s)$ as in Eq. 15). The model parameters are set to $N = 5$, $\rho = 0.5$, $\omega = 0.15$, and $\delta = 1$.

Given the focus of this paper, it is instructive to consider what happens when the LPs are not quoting the competitive equilibrium spread. For simplicity, set $N = 2$ and assume that LP-2 is quoting at $s_2 \neq s^*$. LP-1 sets their spread s_1 to maximise the expected trade valuation function in Eq. (8), i.e. $\arg \max_{s_1} \mathbb{V}_1(s_1, s_2)$. See Panel A of Figure 1 for an illustration.⁸ For instance, when LP-2 quotes at $s_2 = 0.85$, then LP-1's optimal response is to undercut their competitor and quote tighter at $s_1 = 0.69$: the sacrifice in spread capture is more than offset by an increased probability of winning the trader's flow. On the other hand, when LP-2 quotes choice at $s_2 = 0$, then LP-1 maximises the expected trade valuation by quoting wider at $s_1 = 0.13$: winning less flow than their competitor but at more favourable prices. Of course, it is natural for LP-2 to react to where LP-1 is quoting and so we can envisage an iterative process where the LPs take turns in setting their optimal spread in response to what their rival is quoting. Panel B of Figure 1 illustrates the "fixed point" solution where the LPs arrive at the competitive equilibrium spread irrespective of their initial quotes.

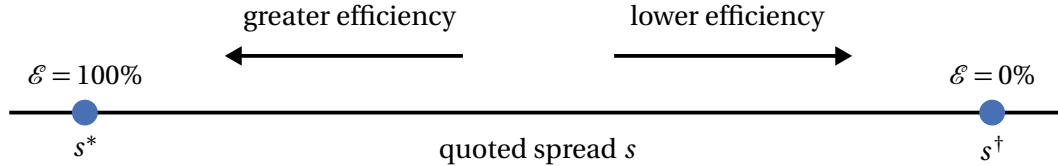
In cases where the LPs quote away from the competitive equilibrium spread, it is helpful to express the relative

⁸Unless otherwise noted, we set $\omega_T = \omega \sqrt{\rho + (1-\rho)/N}$ in all illustrations throughout this paper. This (loosely) corresponds to the case where the trader's estimate of the true price process p^* is simply the average of the N competing LPs' mid-prices.

(in)efficiency of those spreads by normalising them against a “worst case scenario” monopolistic spread s^\dagger . Specifically, we define a pricing efficiency metric as:

$$\mathcal{E}(s) = \frac{\mathbb{V}(s^\dagger) - \mathbb{V}(s)}{\mathbb{V}(s^\dagger) - \mathbb{V}(s^*)}. \quad (16)$$

A value of 100% (0%) means that the expected trade valuation earned by the LP at the selected spread corresponds to that of a competitive (monopolistic) equilibrium.⁹



In the current setting, a symmetric monopolistic equilibrium spread satisfies

$$s^\dagger = \arg \max_{s_i \geq 0} \mathbb{V}_i(\iota_N s_i). \quad (17)$$

Theorem 2 (Monopolistic equilibrium spread) *Let $\omega\sqrt{1-\rho} > 0$ and $\omega_T > 0$. A symmetric monopolistic equilibrium spread satisfies the fixed point problem*

$$s^\dagger = 2\omega\sqrt{1-\rho} \frac{\tilde{\mu}^{(2)}(s^\dagger)}{\tilde{\theta}^{(2)}(s^\dagger)}, \quad (18)$$

where

$$\tilde{\theta}^{(k)}(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^{k-1} f(x, y, s) \phi(x) \phi(y) dx dy, \quad (19)$$

$$\tilde{\mu}^{(k)}(s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^{k-1} \left(y + \sqrt{\frac{\rho}{1-\rho}} x \right) f(x, y, s) \phi(x) \phi(y) dx dy, \quad (20)$$

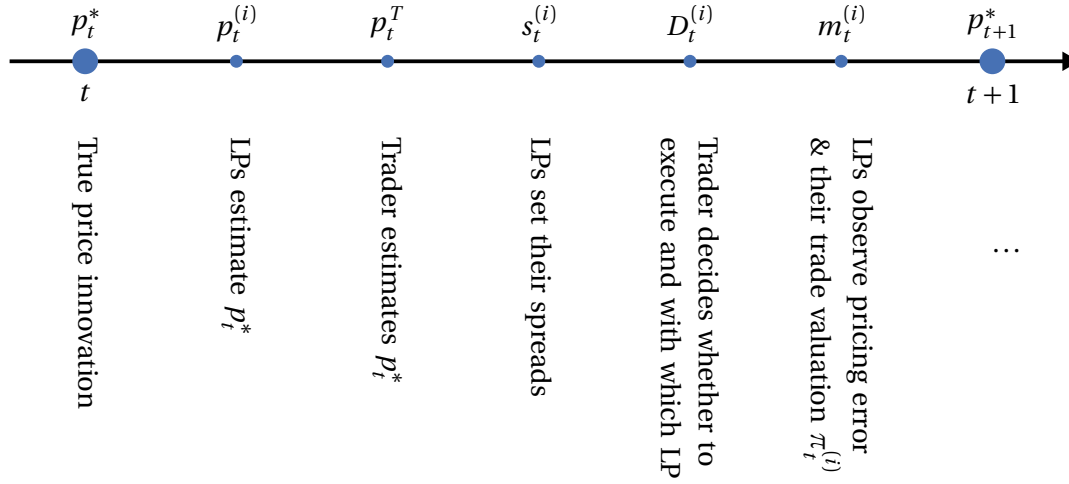
and $f(x, y, s) \equiv f_i(x, y, \iota_N s)$.

Proof See Appendix D.

Figure 2 serves to illustrate a key difference between the two equilibria. In a competitive equilibrium, spreads are naturally constrained by competition even if the trader is willing to trade at any price (i.e. $s_T = \infty$). In a monopolistic equilibrium, on the other hand, the only mechanism that stops LPs from quoting spreads that widen out indefinitely is the traders' control over their reservation price beyond which they do not trade.

⁹When all LPs quote the same spread s , then \mathcal{E} can be larger than one (e.g. when $s < s^*$) but not smaller than zero because the monopolistic equilibrium s^\dagger is the highest revenue point that can be achieved: a spread $s > s^\dagger$ would yield a lower expected trade valuation due to the trader's reservation price constraint. However, when LPs quote different spreads, then the pricing efficiency metric is not bounded by zero either. For instance, it could be negative when one LP quotes, say, s^\dagger and their competitors quote ∞ so that the LP collects all the monopolistic rents without having to share it with their competitors.

Figure 3: Time-line of events



3 AI driven quoting

The model in Section 2 provides a useful framework to analyse the trade-offs faced by the trader and LPs when operating in an OTC market structure. The most profound challenge of all is to navigate the incomplete and asymmetric information: not only is the true price unobserved, the trader's overall liquidity demand concealed, and the LPs blind to their competitors' quoted prices, what is also unknown is the model structure, its parameters, and consequently the trade valuation function and equilibrium spreads. It is precisely this challenge that motivates us to analyse how purely data-driven and model-free algorithms can be used by LPs to set prices. We do this by simulating from the model above so that we can assess the effectiveness of these algorithms by reference to the theoretically optimal spreads derived above.

The class of methods we focus on are so-called multi-armed bandit algorithms (see, e.g., [Sutton and Barto, 2018](#), for an overview) which are designed to address strategic decision making in an unknown environment where an agent chooses an action from a prescribed set of candidate actions and subsequently observes a stochastic reward. The key trade-off faced in such a setup is one between – what is often referred to as – exploration and exploitation. On the one hand, the agent may choose to repeatedly select actions across the full set of candidate actions to build a complete and increasingly accurate estimate of the true reward distribution and its dependence on the action space. On the other hand, the agent may want to avoid the opportunity cost of exploration and instead exploit the rewards of the seemingly optimal arm, albeit one that is based on a noisy – and perhaps incorrect – estimate of the true reward distribution.

Algorithm 1: ϵ –greedy

Parameters: $\epsilon \in [0, 1]$ and a set of candidate spreads $\{s(1), \dots, s(K)\}$.

Initialise: the running average reward $\bar{\pi}_1(s(k)) = 0$ and total number of draws $n_1(k) = 0$ for each arm

$k = 1, \dots, K$.

for $t = 1, 2, \dots$ **do**

 Sample $u \sim \mathcal{U}(0, 1)$.

if $u > \epsilon$ **then**

 Pick the arm with the highest average reward, breaking ties randomly, i.e. $k_t^* = \arg \max_k \bar{\pi}_t(s(k))$.

else

 Randomly pick an arm k_t^* with equal probability $1/K$.

end

Update: propagate previous states for all arms, and increment those for the selected arm as

$\bar{\pi}_{t+1} = \frac{\bar{\pi}_t n_t + \pi_t}{n_t + 1}$, $n_{t+1} \leftarrow +1$ where π_t is the realised reward.

end

In our setting, the above translates to the LP at each time period t , setting a spread by selecting an entry $k_t \in \{1, \dots, K\}$ from a set of candidate spreads $\{s(1), s(2), \dots, s(K)\}$, and subsequently observing the value of any trade done, i.e. $\pi_t(s(k_t))$.¹⁰ The only (weak) assumption we make is that ex-post, the true price p_t^* is observable so that the reward π_t can be established. Figure 3 provides a stylised time-line of events that we adopt throughout, i.e. the value of the trade done in period t is observed immediately before the next time period $t + 1$ so that it can feed in as information used to determine the next spread.

In this paper, we consider four main MAB algorithms that have been proposed in the literature (see, e.g., Sutton and Barto, 2018). The first is the so-called ϵ –greedy method described by Algorithm 1. Its design is simple: with probability ϵ it “explores” by randomly selecting a spread from the set of allowable spreads, and otherwise it “exploits” by selecting the spread with the highest running average reward. A variation of the ϵ –greedy algorithm has been proposed by Auer, Cesa-Bianchi, Freund, and Schapire (2002) in the form of the “Exponential-weighted algorithm for Exploration and Exploitation” or EXP3 as described by Algorithm 2. Again, a fixed exploration rate γ is set which ensures every arm always has a chance of being selected, but different from the ϵ –greedy algorithm, the exploitation stage selects arms probabilistically with arms that have seen higher rewards exponentially achieving a higher probability of being drawn. One obvious drawback of both these algorithms is that the probability of select-

¹⁰For notational convenience, we drop the LP index identifier i and also the reward dependence on competitor spreads s and the market model parameters.

Algorithm 2: EXP3

Parameter: $\gamma \in (0, 1]$ and a set of candidate spreads $\{s(1), \dots, s(K)\}$.

Initialise: weights $w_t(k) = 1$ for each arm $k = 1, \dots, K$.

for $t = 1, 2, \dots$ **do**

Randomly draw an arm k_t^* from a distribution where

$$\Pr(k) = (1 - \gamma) \frac{w_t(k)}{\sum_k w_t(k)} + \gamma \frac{1}{K}. \quad (21)$$

When the reward π_t is realised, update the weights as follows:

$$w_{t+1}(k) = \begin{cases} w_t(k) \exp \frac{\gamma \pi_t(s(k))}{K \Pr(k)}, & k = k_t^* \\ w_t(k), & k \neq k_t^* \end{cases} \quad (22)$$

end

ing the optimal arm, even in the limit, never reaches 1 but is instead capped by $1 - \epsilon + \epsilon/K < 1$ or $1 - \gamma + \gamma/K$ for the respective algorithms.¹¹

An alternative class of methods are the Upper Confidence Bound algorithms, i.e. the UCB-V by [Audibert, Munos, and Szepesvári \(2007\)](#) and the UCB-Tuned by [Auer, Cesa-Bianchi, and Fischer \(2002\)](#) as described in Algorithms 3 & 4. They have two key distinguishing features. Firstly, the term $\ln(t)/n_t(k)$ ensures that all candidate arms get drawn but that over time the exploration rate goes to zero so that the optimal arm is selected consistently in the limit. Secondly, the exploitation takes into account the measurement uncertainty of the running average reward. For instance, if one arm has a running average reward of 10, but an upper confidence bound of say 30 due to it being a very noisy estimate, while another arm has a higher average reward but with far greater certainty, say 15 with an upper confidence bound of 17, then the ϵ -greedy algorithm would exploit the latter arm while the UCB algorithms would draw the former one considering that it still has a good chance of having a superior true reward distribution. Of course, with more observations, the empirical reward estimates associated with each arm become increasingly accurate and so in the limit the UCB bounds converge to the same reward estimates used by the ϵ -greedy algorithm.

Another way in which the MAB algorithms are distinguished from one another, is by the environment they are designed to operate in. The UCB algorithms – members of the so-called “stochastic” class of algorithms – assumes that each arm yields independent and identically distributed rewards, i.e. a stochastic but stationary environment.

¹¹It is, of course, straightforward to modify the algorithms to incorporate a time-dependent and decaying exploration rate. See Figure 15 in Appendix A for an example.

Algorithms 3 & 4: UCB-V & UCB-Tuned

Parameters: a set of candidate spreads $\{s(1), \dots, s(K)\}$.

Initialise: the running average reward $\bar{\pi}_1(s(k)) = 0$, average squared reward $\bar{\pi}_1^2(s(k)) = 0$ and total number of draws $n_1(k) = 0$ for each arm $k = 1, \dots, K$.

for $t = 1, 2, \dots$ **do**

 Pick the arm associated with the highest value of the UCB-V objective function:

$$k_t^* = \arg \max_k \bar{\pi}_t(s(k)) + 3 \frac{\ln(t)}{n_t(k)} + \sqrt{2 \frac{\ln(t)}{n_t(k)} v_t(k)} \quad \text{where} \quad v_t(k) = \bar{\pi}_t^2(s(k)) - (\bar{\pi}_t(s(k)))^2, \quad (23)$$

 or the UCB-Tuned objective function:

$$k_t^* = \arg \max_k \bar{\pi}_t(s(k)) + \sqrt{\frac{\ln(t)}{n_t(k)} \min\left(\frac{1}{4}, v_t(k)\right)} \quad \text{where} \quad v_t(k) = \bar{\pi}_t^2(s(k)) - (\bar{\pi}_t(s(k)))^2 + \sqrt{2 \frac{\ln(t)}{n_t(k)}}. \quad (24)$$

 Any arms with $n_t(k) = 0$ are drawn with priority and equal probability.

 Any ties are broken randomly.

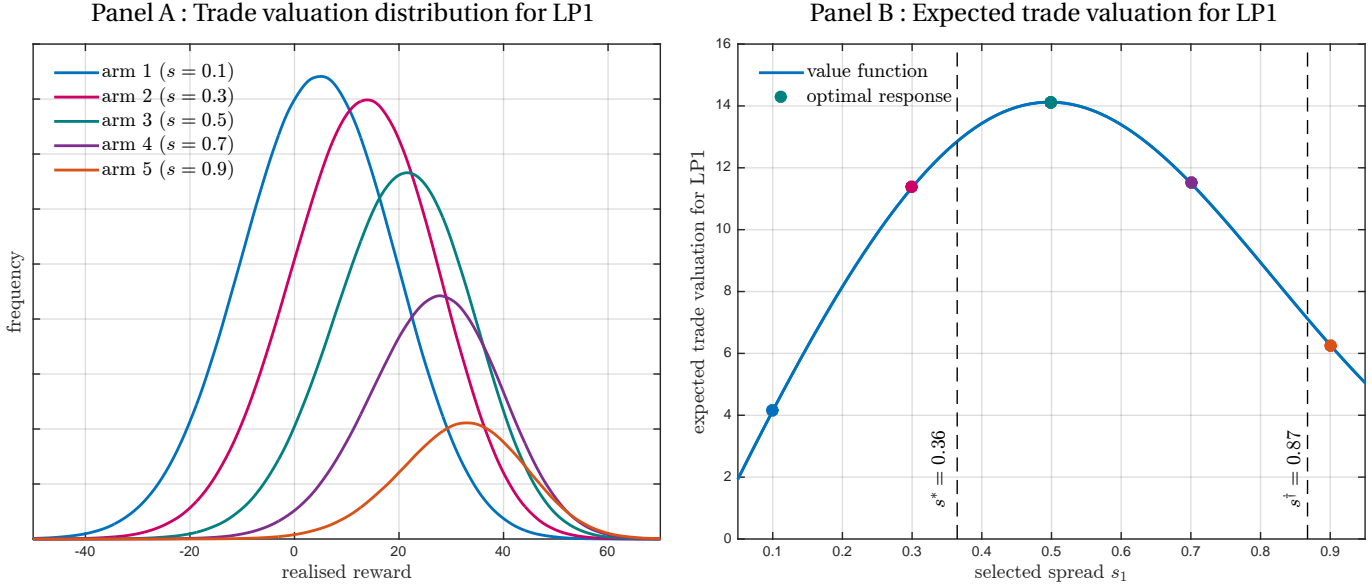
Update: propagate previous states for all arms, and increment those for the selected arm as

$$\bar{\pi}_{t+1} = \frac{\bar{\pi}_t n_t + \pi_t}{n_t + 1}, \quad \bar{\pi}_{t+1}^2 = \frac{\bar{\pi}_t^2 n_t + \pi_t^2}{n_t + 1}, \quad n_{t+1} \leftarrow n_t + 1 \text{ where } \pi_t \text{ is the realised reward.}$$

end

And the arm it selects is deterministic in that the one with the highest upper confidence bound will always be chosen. In contrast, the EXP3 algorithm – member of the so-called “non-stochastic” class of algorithms – is designed for non-stationary or adversarial environments. The algorithm relies on sampling randomised strategies which are harder to exploit by an adversary than the deterministic strategies used in the UCB algorithms. In this classification, the ϵ -greedy algorithm falls somewhere in between: it exploits by deterministically selecting the arm with the highest average reward, but explores in a randomised fashion. The OTC market model clearly suits the EXP3 algorithm in that it's a non-stationary and adversarial environment where the LPs' rewards depend on the spreads chosen by their competitors. The rationale for including the UCB (and ϵ -greedy) algorithm in this analysis is twofold. First, nothing stops LPs from deploying algorithms to environments they are not designed for, and their performance in such scenarios provides a useful contrast to that of the EXP3 algorithm. Second, [Hansen, Misra, and Pai \(2021\)](#) find that in situations where the UCB-Tuned algorithm is misspecified to the environment it operates in, collusive effects can arise. We aim to test this finding in the context of the OTC market model.

Figure 4: Illustration of rewards for given arm draws



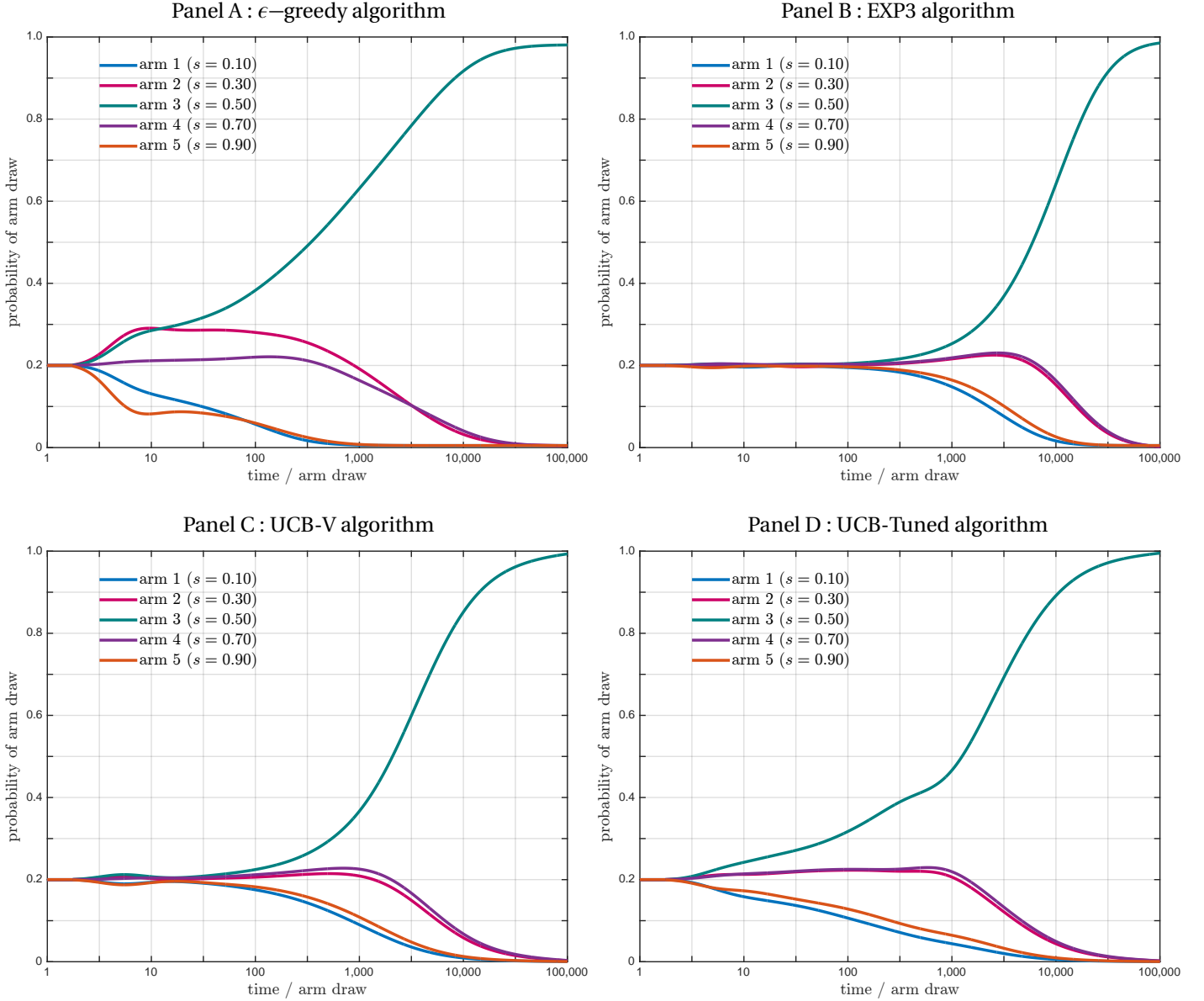
Note. Panel A displays the distribution of trade valuations $\pi_t^{(1)}(s)$ in Eq. (7) obtained by LP-1 for each candidate spread, when competitor LP-2 is quoting a fixed spread $s_2 = 0.7$. Panel B draws the expected trade valuation function $V_1(s)$ in Eq. (8) as a function of LP-1's spread s_1 and for fixed competitor spread $s_2 = 0.7$. The OTC market model parameters are set to baseline values.

3.1 Stationary environment and single-agent learning

We first analyse the case where only one LP is learning where to quote through the use of the MAB algorithms described above while the other LPs quote a fixed spread. The OTC market model we simulate from is described in Section 2. Unless stated otherwise, our baseline parameterisation takes $N = 2$, $\rho = 0.5$, $\omega = 0.15$, $\omega_T = \omega \sqrt{\rho + (1 - \rho)/N}$, $s_T = 1$, $\delta = 1$, $\delta_B = 1/2$ together with a set of $K = 5$ candidate spreads $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. For ease of presentation, we also multiply the trade valuations by a factor of one hundred in the Figures and Tables throughout the paper. LP-2 is assumed to quote a constant spread of $s_2 = 0.7$, and in this setting the optimal response is for LP-1 to draw the third arm and set a spread $s_1 = 0.5$. The reward distribution for each candidate spread and the trade valuation function is illustrated in Figure 4. Table 6 in Appendix A provides further metrics for the different arm choices: the optimal spread finds a balance between showing a tighter spread to increase the probability of both meeting the trader's reservation price and then to winning the trade against showing a wider spread to increase the per-trade revenues. Controlling adverse selection is another factor, albeit of secondary importance here.

With the model configured, we now assess the ability of the various MAB algorithms to learn about the trading environment and whether – and if so, how quickly – they will select the optimal spread. For this purpose, we simulate

Figure 5: Speed of convergence to optimal arm draw



Note. This chart draws the probability of selecting each candidate spread as a function of time when LP-1 is using an MAB algorithm to set spreads while their competitor LP-2 is quoting a fixed spread $s_2 = 0.7$. In this scenario, the optimal spread to quote for LP-1 is $s_1 = 0.5$ or arm 3. Panels A – D report results for the MAB algorithms 1, 2, 3 & 4 respectively. The OTC market model parameters are set to baseline values.

1,000 independent runs of 100,000 time steps each where at every point a specific arm k_t is chosen and an associated reward $\pi_t(s(k_t))$ is received. This then feeds into the algorithm's logic to determine the next arm to pull. Figure 5 reports the probability of the algorithms selecting a particular candidate spread at each time step.¹² Given $K = 5$

¹²All probability curves reported in this paper, are calculated as smoothed averages across 1,000 independent simulation replications,

and the initial arm draw is random, all probability curves start off at 20% but then evolve as rewards are observed and arm choices are adjusted following the prescribed logic. For all algorithms, the third and theoretically optimal arm becomes the dominant choice as time progresses, with a caveat that the ϵ -greedy and EXP3 algorithms never reach the point where they draw the optimal arm with probably one due to the fixed exploration rate. The observed convergence to the optimal arm is re-assuring but unsurprising given the theoretical guarantees that exist.¹³ The speed of convergence does appear to vary somewhat, with the ϵ -greedy algorithm reaching a 90% optimal arm selection after about 8,000 time steps, while the UCB can take up to 14,000 and the EXP3 up to 28,000 time steps. We should note, however, that convergence aside, the absolute and relative speeds vary quite substantially depending on the specific parameterisation of the trading environment as well as the choice of MAB hyper-parameters such as the set of candidate spreads and the exploration rates ϵ and γ . Table 6 in Appendix A illustrates this point.¹⁴ If the number of arms is reduced to $K = 3$, either via a narrower grid of candidate spreads where $\{0.3, 0.5, 0.7\}$ or a coarser grid of candidate spreads $\{0.1, 0.5, 0.7\}$ the speed of convergence increases substantially, especially for the latter case. Conversely, with a larger number of $K = 9$ arms and a finer grid of candidate spreads $\{0.1, 0.2, \dots, 0.9\}$, convergence is substantially slower¹⁵ by a factor of between 7 and 25. The fourth scenario in the table considers a finer grid ($K = 9$) combined with more competitors ($N = 4$). Compared to the $N = 2$ case, convergence is slightly faster, which at first sight is counter-intuitive, but this is simply because we are not comparing like-for-like: here, the competitor spreads are set to 1, and the value function is now more peaked making it easier to identify the optimal response. The final scenario with $\delta = 1/2$ simply leads to about half the trade observations and – unsurprisingly – roughly halves the speed of convergence for all algorithms.

The results show that the MAB algorithms can cope with the trading environment considered here, and are able to successfully discover the optimal spread to charge via the process of exploration and the evaluation of observed rewards. In terms of speed of convergence, there is no clear ranking between the MAB algorithms and their relative performance varies with the specific scenarios and model configurations.

The model-free nature of the algorithms does mean they cannot incorporate environment specific information and this can limit their efficiency.¹⁶ For instance, in our application, it would seem reasonable to assume that the

using the local linear regression smoother of Fan and Gijbels (1992) with a bandwidth of 0.25 in logarithmic time (i.e. $\log_{10}(t)$).

¹³The literature in this area tends to study convergence in terms of minimising the regret associated with pulling a sub-optimal arm. See, for instance, Auer, Cesa-Bianchi, Freund, and Schapire (2002) on the convergence rate of the EXP3 algorithm, or Auer, Cesa-Bianchi, and Fischer (2002) for the ϵ -greedy and UCB variants.

¹⁴See also Figure 15 in Appendix A for sensitivity of the ϵ -greedy and EXP3 algorithm to exploration rate parameters ϵ and γ .

¹⁵This suggests that rather than starting off with an overly fine set of arm choices and large K , it may be more efficient to iterate through a series of coarser grids with smaller K , and on convergence retain the optimal arm supplemented with new previously unexplored candidate arms.

¹⁶This limitation can be addressed via the use of so-called contextual bandit algorithms, but this is beyond the scope of the paper. Interested

expected reward as a function of spread is continuous. So observed rewards should contain information not just at the specific point sampled, but in a local neighbourhood around it. This is precisely the reason why the convergence of the MAB algorithms is so sensitive to the resolution of the state space and increasing the number of arms K carries a large penalty in terms of speed. Motivated by this observation, we benchmark the performance of the model-free MAB algorithms against a fully parametric approach where the model structure is assumed to be known but the model parameters are not. This serves to assess the efficiency that is sacrificed in exchange for robustness to potential model mis-specification. To ease exposition, we require a slightly modified notation where $s_{1:T}$ denotes the time-series of selected spreads $\{s_1, s_2, \dots, s_T\}$ by the optimising LP, and similarly $D_{1:T}$ and $m_{1:T}$ denote the time-series of trade indicators and measurement errors observed by the optimising LP as per Figure 3. Also, we denote the vector of (unknown) model parameters as $\Theta = (\delta, \delta_B, \omega, \rho, s_2, \omega_T, s_T)$. In what follows, we assume that at the end of period T the LP observes m_T, s_T, D_T and has kept a record of $m_{1:T-1}, s_{1:T-1}, D_{1:T-1}$. Because the assumed observability of π_t in the MAB algorithms requires p_t^* , the same information can be used to establish m_t , and so comparable information is used by the MLE. The log-likelihood function associated with the market model for $N = 2$ can now be explicitly written down as:

$$\mathcal{L}(m_{1:T}, D_{1:T} | \Theta, s_{1:T}) = \sum_{t=1}^T \ln L(m_t, D_t | \Theta, s_t), \quad (25)$$

where $L : \mathbb{R} \times \{-1, 0, 1\} \rightarrow \mathbb{R}$ is given by

$$L(m, 1 | \Theta, s) = \frac{\delta \delta_B}{\omega} \phi\left(\frac{m}{\omega}\right) \left(1 - \Phi\left(\frac{(1-\rho)m + \frac{1}{2}(s-s_2)}{\omega \sqrt{1-\rho^2}}\right)\right) \left(1 - \Phi\left(\frac{m + \frac{1}{2}(s-s_T)}{\omega_T}\right)\right), \quad (26)$$

$$L(m, -1 | \Theta, s) = \frac{\delta(1-\delta_B)}{\omega_i} \phi\left(\frac{m}{\omega}\right) \Phi\left(\frac{(1-\rho)m - \frac{1}{2}(s-s_2)}{\omega \sqrt{1-\rho^2}}\right) \Phi\left(\frac{m - \frac{1}{2}(s-s_T)}{\omega_T}\right), \quad (27)$$

$$L(m, 0 | \Theta, s) = \frac{1}{\omega} \phi\left(\frac{m}{\omega}\right) - L(m, 1 | \Theta, s) - L(m, -1 | \Theta, s). \quad (28)$$

See Appendix D for a derivation of Eq. (25). The above enables straightforward maximum likelihood estimation (MLE) of the model parameters Θ by numerically maximising the likelihood function \mathcal{L} over the allowable parameter space for a given data sample, i.e.

$$\hat{\Theta}_T = \arg \max_{\Theta} \mathcal{L}(m_{1:T}, D_{1:T} | \Theta, s_{1:T}). \quad (29)$$

With estimated model parameters in hand, the expected trade valuation function in Eq. (8) can be evaluated at each candidate spread to determine the optimal one.

reader can refer to, e.g., [Lattimore and Szepesvári \(2020, Chapter 18\)](#). However, the maximum likelihood estimator can be viewed as a limiting case of a contextual bandit algorithm that optimally incorporates the full model structure.

Table 1: Parametric approach – maximum likelihood estimation

sample size	Panel A : frequency of arm draws					Panel B : MLEs of model parameters						
	$s = 0.1$	$s = 0.3$	$s = 0.5$	$s = 0.7$	$s = 0.9$	$\delta = 1$	$\delta_B = \frac{1}{2}$	$\omega = 0.15$	$\rho = \frac{1}{2}$	$s_2 = 0.7$	$\omega_T = 0.13$	$s_T = 1$
<i>Scenario 1 : correct model specification</i>												
$T = 10$	0.2%	14.6%	63.0%	16.9%	5.3%	0.97 (0.73,1.00)	0.50 (0.14,0.90)	0.14 (0.08,0.21)	0.63 (0.30,0.95)	1.75 (0.43,2.92)	0.08 (0.00,0.39)	1.02 (0.36,3.50)
$T = 100$	0.0%	0.0%	91.3%	8.7%	0.0%	0.99 (0.91,1.00)	0.50 (0.38,0.62)	0.15 (0.13,0.17)	0.59 (0.17,0.95)	1.33 (0.60,3.17)	0.09 (0.00,0.22)	0.88 (0.66,1.22)
$T = 1,000$	0.0%	0.0%	99.5%	0.5%	0.0%	1.00 (0.98,1.00)	0.50 (0.46,0.54)	0.15 (0.14,0.16)	0.55 (0.39,0.77)	0.88 (0.65,2.58)	0.12 (0.01,0.18)	0.95 (0.71,1.15)
$T = 10,000$	0.0%	0.0%	100.0%	0.0%	0.0%	1.00 (0.99,1.00)	0.50 (0.49,0.51)	0.15 (0.15,0.15)	0.51 (0.45,0.57)	0.70 (0.67,0.74)	0.13 (0.10,0.15)	1.00 (0.92,1.06)
$T = 100,000$	0.0%	0.0%	100.0%	0.0%	0.0%	1.00 (1.00,1.00)	0.50 (0.50,0.50)	0.15 (0.15,0.15)	0.50 (0.49,0.52)	0.70 (0.69,0.71)	0.13 (0.12,0.14)	1.00 (0.98,1.02)
sample size	$s = 0.1$	$s = 0.3$	$s = 0.5$	$s = 0.7$	$s = 0.9$	$\delta = 1$	$\delta_B = \frac{1}{2}$	$\omega = 0.15$	$\rho = \frac{1}{2}$	$s_{2:8} = 1$	$\omega_T = 0.11$	$s_T = 1$
<i>Scenario 2 : model mis-specification (assume 2 LPs when there are 8)</i>												
$T = 10$	0.0%	12.6%	64.2%	19.0%	4.2%	0.99 (0.80,1.00)	0.50 (0.12,0.88)	0.14 (0.09,0.21)	0.69 (0.44,0.95)	1.57 (0.49,2.81)	0.07 (0.00,0.30)	1.17 (0.40,3.51)
$T = 100$	0.0%	0.0%	96.9%	3.1%	0.0%	1.00 (0.95,1.00)	0.50 (0.38,0.62)	0.15 (0.13,0.17)	0.68 (0.31,0.95)	0.90 (0.64,2.53)	0.06 (0.00,0.18)	0.91 (0.70,1.16)
$T = 1,000$	0.0%	0.0%	100.0%	0.0%	0.0%	1.00 (0.99,1.00)	0.50 (0.46,0.54)	0.15 (0.14,0.16)	0.66 (0.49,0.79)	0.79 (0.70,0.85)	0.10 (0.05,0.13)	0.90 (0.81,1.01)
$T = 10,000$	0.0%	0.0%	100.0%	0.0%	0.0%	1.00 (1.00,1.00)	0.50 (0.49,0.51)	0.15 (0.15,0.15)	0.64 (0.60,0.69)	0.76 (0.74,0.78)	0.10 (0.09,0.11)	0.91 (0.88,0.93)
$T = 100,000$	0.0%	0.0%	100.0%	0.0%	0.0%	1.00 (1.00,1.00)	0.50 (0.50,0.50)	0.15 (0.15,0.15)	0.64 (0.63,0.65)	0.76 (0.75,0.76)	0.10 (0.10,0.10)	0.90 (0.89,0.91)

Note. Panel A reports the frequency of arm draws across 1,000 independent simulation runs, where the selected arm is the one that maximises the expected trade valuation function that is parametrised using the MLEs of the model parameters. Panel B reports the average MLEs of the model parameters, together with the 2.5– and 97.5–percentiles in parenthesis below. The model parameters are estimated using the likelihood function in Eq. (25) which assumes that $N = 2$. In scenario 1 at the top, the likelihood corresponds to the true model, whereas in scenario 2 the same likelihood is used with $N = 8$ instead of the assumed $N = 2$. In scenario 2, the competitor spreads are adjusted to $s_i = 1$ for $i = 2, \dots, N$ to ensure the optimal arm remains the same.

To simulate the data $(m_{1:T}, s_{1:T}, D_{1:T})$, we use the same model parameters as before to facilitate comparison with the MAB analysis. Because the MLE approach does not dictate how spreads should be set by the optimising LP, we assume – for simplicity and computational speed – that for the first T periods they select spreads randomly from the same set of candidate spreads used by the MAB algorithms, and subsequently conduct the maximum likelihood estimation of the model parameters, i.e. $\hat{\Theta}_T$. In other words, full exploration for the first T periods. The estimated market model parameters can then be plugged into Eq. (8) from Lemma 1 to determine the optimal (trade valuation maximising) spread to quote thereafter. Table 1 reports the results of this exercise where the sample size T used

in the maximum likelihood estimation is varied between 10 and 100,000 observations. The top panel shows the superior efficiency of the parametric MLE approach when compared to the model-free MAB approach, and the difference is substantial with very fast convergence and the optimal spread being selected more than 90% of the time based on only 100 observations. Unreported results show that on a finer candidate spread grid with $K = 9$, the probability of selecting the optimal arm is over 90% following this MLE approach after 1,000 observations. This suggests a convergence rate that is at least one order of magnitude faster than that of the best performing MAB algorithms.

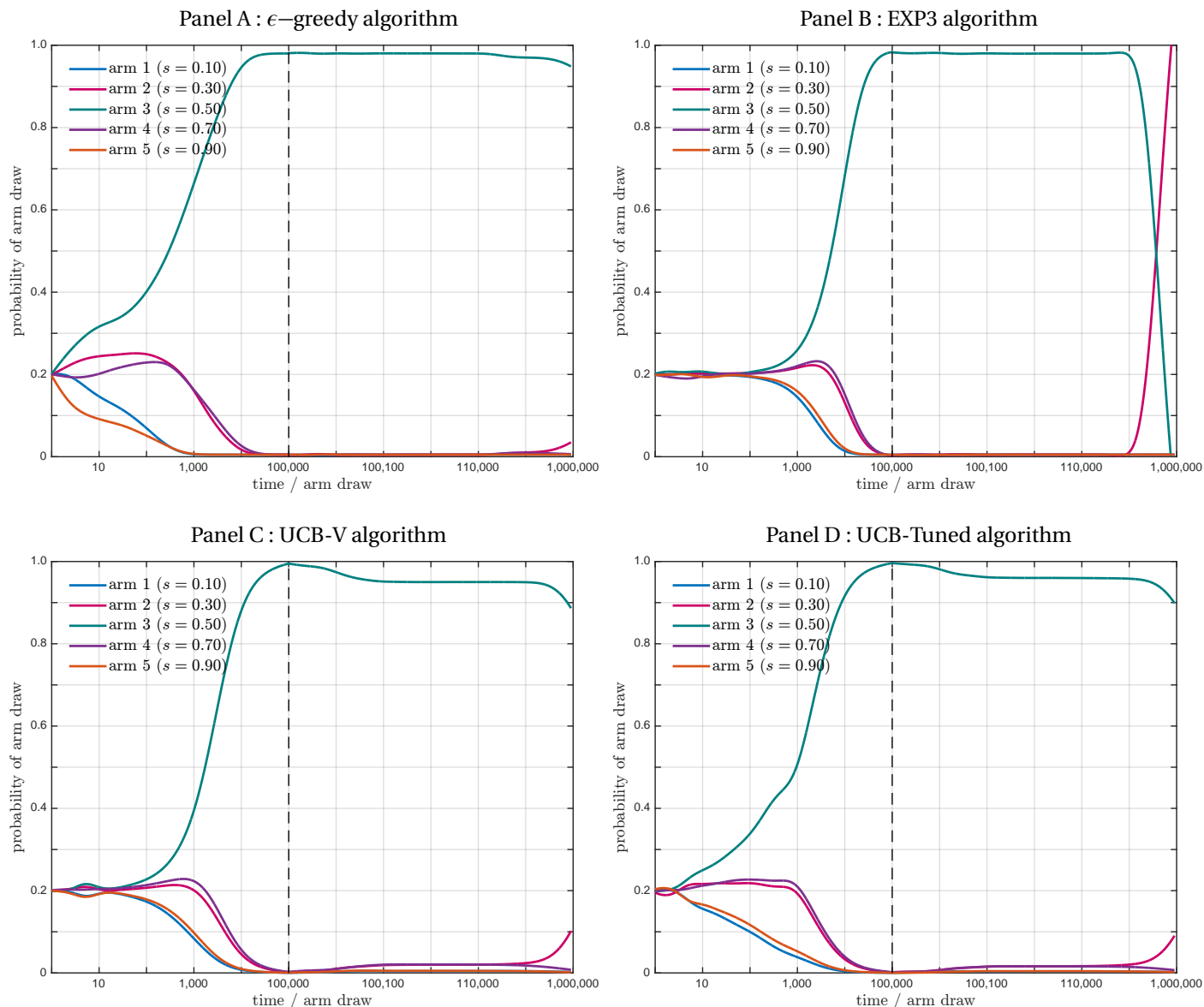
As already noted, an obvious drawback of the MLE approach is that it is prone to model mis-specification. The lower panel in Table 1 reports results where we simulate from a model with eight LPs instead of two, but still use the same likelihood function as above. The likelihood is therefore mis-specified and we can see from Panel B that certain parameter estimates are now severely biased, even in large samples. In particular, while the seven competitors are quoting at a spread of 1, the MLE optimising LP believes they are in competition with only one LP that is quoting a spread of 0.76 and that the correlation of their measurement errors m is 64% instead of the true correlation of 50%. Interestingly, however, even with this mis-specification and biased MLE parameter estimates, the optimal arm is selected and the convergence rate is equally, if not more, impressive. The intuition for this is provided in Figure 16 in Appendix A. Convergence to the correct arm is achieved because – as it turns out – the correctly specified expected trade valuation function is closely approximated by the mis-specified one when evaluated at the biased parameter estimates. And the convergence is faster, because the value function is more peaked than in the first scenario. While this analysis is by no means exhaustive, it does provide a useful perspective on the considerations that need to be weighed by the LP when deciding on a data-driven approach to their pricing optimisation.

3.2 Non-stationary environment and multi-agent learning

The results presented thus far assume that a single LP is optimising their quoting strategy within an otherwise static environment. The remainder of this paper focuses on the case where multiple LPs are simultaneously but independently optimising their liquidity provision to the trader. We start with a couple of simple examples to illustrate that the MAB performance can change radically when competitor LPs also change their quoting strategy, either at a discrete point in time or via a concurrent and dynamic optimisation routine. This so-called multi-agent learning is a topic of substantial and growing interest in the literature (see, e.g., [Zhang, Yang, and Başar, 2021](#), for a recent overview). The next section provides an in-depth analysis using replicator dynamics.

The first example considers the case where $N = 4$ LPs compete for a trader's flow. LP-1 is setting spreads using the MAB algorithms discussed above, while their three competitors quote a constant spread of 0.9 for the first

Figure 6: Non-stationary environment: competitors tighten spread and optimal arm changes from 3 to 2



Note. This chart draws the probability of arm choice for the optimising LP when their competitors are quoting a fixed spread of $s_{2:4} = 0.9$ for the first 100,000 periods and then independently tighten spreads to $s_{2:4} = 0.4$ for the remaining 900,000 time periods. The optimal spread shifts from 0.5 (arm 3) to 0.3 (arm 2) in this scenario. $N = 4$ and the remaining OTC market model parameters are set to baseline values. The x-axis is on a \log_{10} scale that is reset when the competitors' quotes change as indicated by the dashed line.

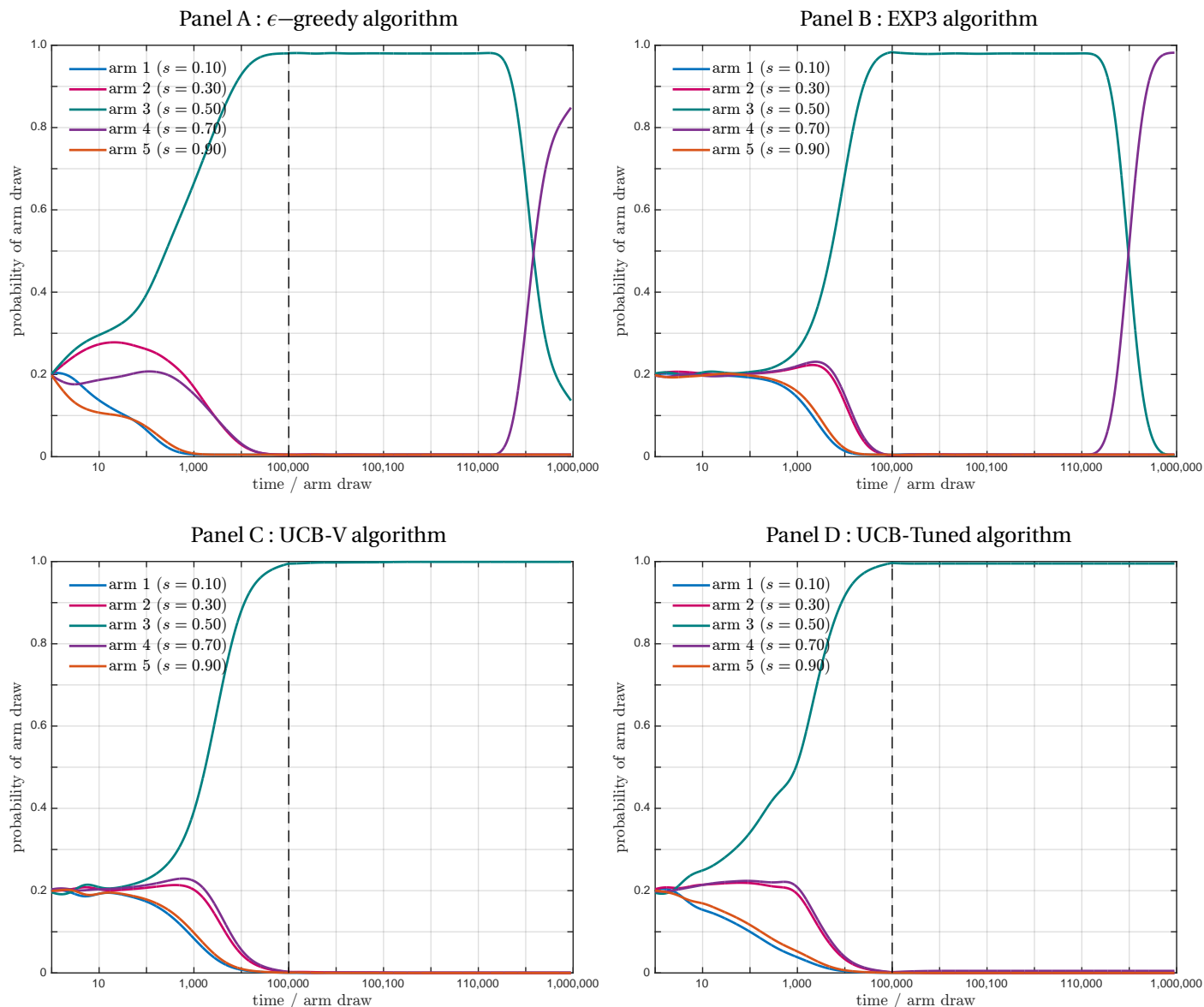
100,000 time periods, and then independently tighten (widen) their spreads to 0.4 (1.35). The optimal spread for LP-1 to quote is 0.5 (arm 3) while their competitors quote at 0.9, but this changes to 0.3 (arm 2) as they tighten their spreads and 0.7 (arm 4) as they widen.

First consider the results of the spread tightening scenario in Figure 6. As expected, in the initial phase where competitor spreads are static, all the MAB algorithms converge to the optimal arm choice, but as competitors tighten their spreads several of the MAB algorithms struggle to adapt to this discrete change of environment. Notably, the ϵ -greedy and UCB algorithms continue to quote at the previously established but now sub-optimal spread for extended periods of time and even after nearly one million time steps, the probability of quoting the optimal arm 2 or a spread of 0.3 is only slightly elevated at between 5% and 10%. In other words, the adjustment is very sluggish and time-consuming. This can be explained as follows. After competitors independently tighten spreads, the expected rewards associated with all arms – including the new optimal arm – decrease substantially (see Figure 17 in Appendix A). This starts to bring down the measured average rewards $\bar{\pi}_t(s(k))$ across all arms, but does so much slower for the previously optimal arm 3 as it is an average based on many more observations than any of the other arms. In other words, the objective function needs to be reshaped and degraded at the previously optimal arm before the algorithm starts to sample new arms again, including the optimal arm 2 which now produces the highest rewards albeit still lower than before the spread tightening.¹⁷ The UCB algorithms face the added challenge that the upper confidence bounds (i.e. the terms involving $\ln(t)/n_t(k)$ in Eqs. 23 and 24) shrink quicker for the newly selected arm 2 than they do for previously frequently drawn arm 3. The EXP3 algorithm, on the other hand, adjusts more rapidly to a change in environment. This is because the rewards are weighted by their inverse arm probabilities and so a large reward observed for a low probability arm leads to a rapid upward revision of that arm probability.

Next, consider the results of the spread widening scenario in Figure 7. The rewards for every arm draw now increase and the optimal arm choice changes from 3 to 4 or a spread of 0.7. The most notable difference in MAB performance is that the UCB algorithms are now unable to adjust to this new environment over the time scales considered here. Intuitively, the rewards associated with the now sub-optimal arm 3 do still increase and are actually higher compared to where they were previously. Consequently, the algorithm continues to draw this arm as the distance in objective function between arm 3 and the other arms further increases. The fact that arm 4 has a higher payoff goes unnoticed in this case and converges to arm 3 continues to get reinforced by the higher rewards preventing exploration of other arms including the optimal arm 2. The ϵ -greedy algorithm, on the other hand, can adjust

¹⁷Table 7 in the Appendix A provides some numerical illustration. At the end of the first phase, both the ϵ -greedy and UCB algorithms measure an average reward of 15.5 associated with drawing arm 3 based on about 95,000 draws and an average reward of 12.2 associated with drawing arm 2 based on about 2,000 draws. After competitors tighten their spreads, the expected reward for arm 3 goes down to 1.6 and the now optimal arm 2 yields a higher but still modest expected reward of 2.4. So the challenge is for the established average of arm 3 to be brought down from 15.5 to below that of arm 2, but each time arm 2 is selected the average drops, and does so quicker because it is based on fewer observations and the new reward of 2.4 is substantially below 12.2. Overcoming this adjustment phase is naturally very time consuming.

Figure 7: Non-stationary environment: competitors widen spread and optimal arm changes from 3 to 4

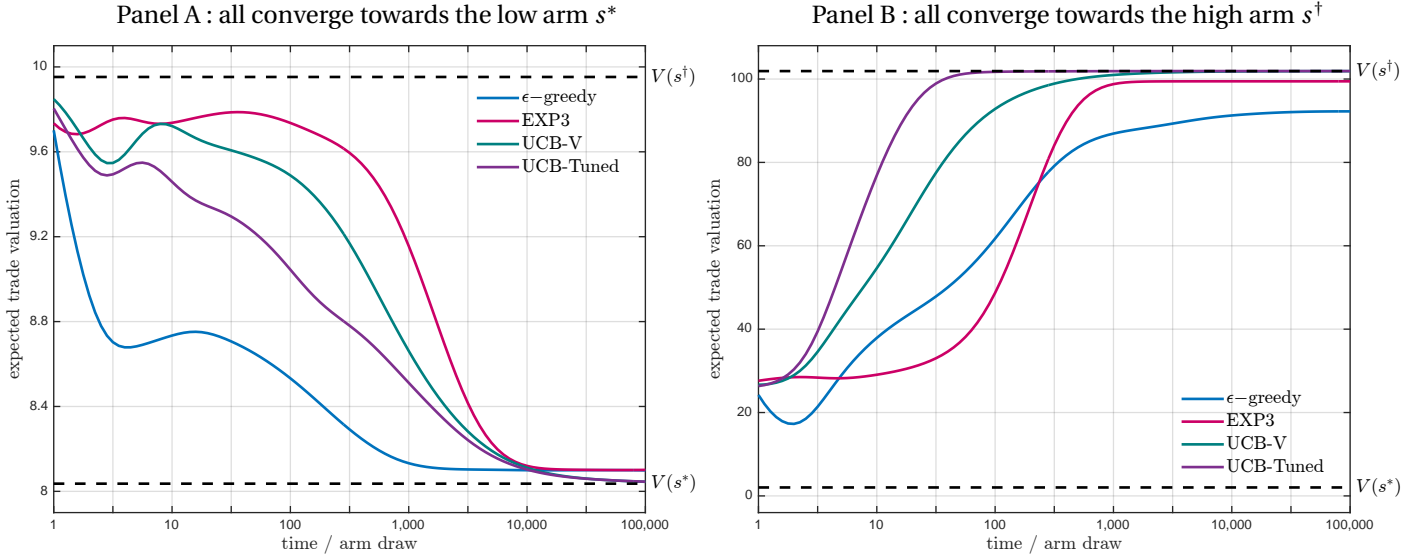


Note. This chart draws the probability of arm choice for the optimising LP when their competitors are quoting a fixed spread of $s_{2,4} = 0.9$ for the first 100,000 periods and then widen spreads to $s_{2,4} = 1.35$ for the remaining 900,000 time periods. The optimal spread shifts from 0.5 (arm 3) to 0.7 (arm 4) in this scenario. $N = 4$ and the remaining OTC market model parameters are set to baseline values. The x-axis is on a \log_{10} scale that is reset when the competitors' quotes change as indicated by the dashed line.

and does so quicker than in the spread tightening case. The EXP3 algorithm is quickest to adapt in both scenarios.

As a final remark, it is clear that the MAB algorithms' limited ability to adapt to a change in environment is simply due to it needing to overcome the "legacy" information compounded into its objective function from a previous but

Figure 8: Non-stationary environment: multi-agent learning scenarios



Note. This chart displays the expected trade valuation associated with the LPs' arm choices when they all use the same MAB algorithm concurrently to set their spreads. It is calculated over 1,000 independent simulation runs. In Panel A, $\rho = 0$, $s_T = 1$ with associated spread levels $s^* = 0.51$ and $s^\dagger = 1.20$. In Panel B, $\rho = 0.95$, $s_T = 5$ with associated spread levels $s^* = 0.12$ and $s^\dagger = 4.30$. For the ϵ -greedy and EXP3 algorithms, $\epsilon = \gamma = 2.5\%$. The remaining OTC market model parameters are set to baseline values.

now superseded environment. The MLE approach would face the same challenge. One obvious way to resolve this is to restart the algorithm or estimation procedure at the point where the environment has materially changed. In practice, however, it is often unknown whether and if so when a change has occurred. This motivates a literature on structural break testing and the same could be applied in this context, but is beyond the scope of this paper.

We now turn to the second example of a non-stationary environment where every LP simultaneously but independently sets spreads based on an MAB algorithm. Because this mainly serves to provide the motivating context for the next section, we keep the setup here as simple as possible and the discussion brief. We assume $N = 2$ competing LPs set spreads by drawing at every time-point either the “low” arm s^* or the “high” arm s^\dagger , as determined by the objective function of their chosen MAB algorithm. We assume both LPs use the same algorithm, start at the same time, and run it for 100,000 time periods. Figure 8 reports the expected trade valuation calculated across 1,000 independent replications for two different model configurations where ρ and s_T are varied. Panel A shows an example where both LPs end up consistently drawing the low arm and set spreads at s^* , irrespective of the specific MAB algorithm they deploy. Panel B, on the other hand, is an example where both LPs converge towards drawing the high arm and set their spread at s^\dagger .

3.3 Pseudo collusion

Thus far we have avoided labelling convergence to the high arm as collusive: the mere observation that a “high” price is being charged does not – by itself – constitute evidence of collusion via independently run MAB algorithms. We first need to be clear and precise about the terminology and concepts involved. Collusion involves explicit coordination and agreement amongst competitors in order to enable anti-competitive conduct. A key feature of a collusive outcome is that the collective interests of competitors are prioritised over the uni-lateral interests of individual competitors who would – in a competitive environment – be incentivised to deviate from the collusive strategy and undercut their rivals were it not for an agreement that prevents them from doing so. Collusion is illegal and not in scope of this paper: we consider an environment where all competitors act independently.¹⁸ Tacit collusion refers to the case where collusive outcomes are achieved but without explicit coordination or agreement amongst competitors. It requires a repeated game which embodies a retaliation mechanism that is sufficiently punitive and designed to prevent competitors deviating from the collusive strategy in pursuit of their individual interests (see, e.g. Ivaldi, Jullien, Rey, Seabright, and Tirole, 2003). However, AI algorithms are often intractable “black boxes” and it may be unclear whether or not they embody any form of retaliatory mechanism. For this reason, we introduce the novel concept of pseudo collusion. Informally, a pseudo collusive state is one where supra-competitive prices are sustained while competitors are individually and statically incentivised to deviate. Thus, pseudo collusion can lead to outcomes that are observationally equivalent to a collusive or tacitly collusive state but it doesn’t require there to be explicit agreement nor the existence of a retaliation mechanism to sustain it.

Definition 1 (Pseudo collusion with continuous pricing) *An equilibrium spread \bar{s} is supra-competitive when $\bar{s} > s^*$. It is pseudo collusive when $\bar{s} > s^*$ and there exists a spread s where $\mathbb{V}_i(s, \bar{s}) > \mathbb{V}_i(\bar{s}, \bar{s})$.*

Because the MAB algorithms in this paper – and financial markets in general – operate with discrete prices and spreads, we require a modified definition of pseudo collusion to account for this.

Definition 2 (Pseudo collusion with discrete pricing) *An equilibrium spread \bar{s} is pseudo collusive when $\bar{s} > s^*$ and there exists a spread s in the allowable set of candidate spreads where $\mathbb{V}_i(s, \bar{s}) > \mathbb{V}_i(\bar{s}, \bar{s})$.*

Note that because a static Nash equilibrium requires $\mathbb{V}_i(s, \bar{s}) < \mathbb{V}_i(\bar{s}, \bar{s})$ for all s , it can be supra-competitive but never pseudo collusive. In a static Nash equilibrium the individual LP is – at any point in time, and without regard for the future – inherently incentivised not to deviate and so this equilibrium does not require any form of collusive effects to be sustained. Throughout the remainder of this paper, we will adopt the above definition of pseudo collusion.

¹⁸Also, competitors do not observe their rivals’ prices, and the information that is available to them is limited to the commercials of their own private and bi-lateral dealings with the trader and initiated by the trader.

4 MAB convergence properties under replicator dynamics

A limitation of the analysis above is that it relies on extensive and time-consuming simulations for specific model configurations which, in turn, is not amenable to comprehensively characterise the algorithms' behaviours and properties. This section sets out to address this limitation. In particular, we use stochastic approximation techniques to show that trajectories of the policies (probabilities of playing each action) generated by the EXP3 algorithm follow the trajectories from the replicator dynamics. Importantly, the replicator dynamics allow us to prove some key properties of the algorithm, and facilitate analysis for an arbitrary number of competitors N and number of candidate spreads K .

Lemma 2 *Let $\mathcal{P}_{i,t}(k)$ be the discrete-time stochastic process that describes the probability of LP- i drawing arm k at time t via the EXP3 Algorithm 2. For a sufficiently small value of γ , the increment of $\mathcal{P}_{i,t}(k)$ is given by*

$$\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) = \begin{cases} \frac{\gamma \pi_{i,t}(s(k))}{K} (1 - \mathcal{P}_{i,t}(k)) + O(\gamma^2), & k = k_{i,t}^*, \\ -\frac{\gamma \pi_{i,t}(s(\ell))}{K} \mathcal{P}_{i,t}(k) + O(\gamma^2), & k \neq k_{i,t}^* = \ell. \end{cases} \quad (30)$$

Proof See Appendix D.

Theorem 3 *Let $\mathcal{P}_{i,t}(k)$ be the discrete-time stochastic process that describes the probability of LP- i drawing arm k at time t for the EXP3 Algorithm 2. The expected increment of \mathcal{P} is described by the continuous globally integrable vector field $\dot{\mathcal{P}} : [0, 1]^{N \times K} \rightarrow [0, 1]^{N \times K}$ with component functions*

$$\dot{\mathcal{P}}_{i,t}(k) = \frac{\mathcal{P}_{i,t}(k)}{K} \left[\bar{\nabla}_{i,t}(s(k)) - \sum_m \mathcal{P}_{i,t}(m) \bar{\nabla}_{i,t}(s(m)) \right], \quad \text{for } m, k \in \{1, 2, \dots, K\}, \quad (31)$$

where the dot denotes the derivative with respect to time and $\bar{\nabla}_{i,t}(s(k)) = \sum_{m, j \neq i} \nabla_i(s(k), s(m), s) \mathcal{P}_{j,t}(m)$, i.e. the expected reward of LP- i when selecting action k , calculated as the expected trade valuation in Eq. (8) weighted by the probability of competitor actions.

Set $\tau_t = \gamma t$ and let $\hat{\mathcal{P}}$ be the continuous-time affine interpolated processes of policies for N independent learning agents using the EXP3 algorithm with entries given by

$$\hat{\mathcal{P}}_{i,\tau_t+\Delta}^\gamma(k) = \mathcal{P}_{i,t}(k) + \Delta \frac{\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k)}{\gamma}, \quad (32)$$

for all $t \in \mathbb{N}$ and $0 \leq \Delta < \gamma$. Then, for a sufficiently small value of γ , $\hat{\mathcal{P}}^\gamma$ is a pseudo-trajectory of the flow induced by $\dot{\mathcal{P}}$ with probability one.

Proof See Appendix D.

Theorem 3 establishes that for a sufficiently small value of the learning rate γ , the evolution of the EXP3 algorithm converges to a trajectory of the replicator dynamics in Eq. (31). Intuitively, the learning rate γ is inversely proportional to the number of observations that are locally available to drive incremental updates to the probabilities of drawing the respective arms. Therefore, as the learning rate falls ($\gamma \rightarrow 0$), the law of large numbers applies and this ensures that the actual trajectory (a realisation) of a single run of the algorithm converges to its expected trajectory which are described by the replicator dynamics (see Börgers and Sarin, 1997).¹⁹

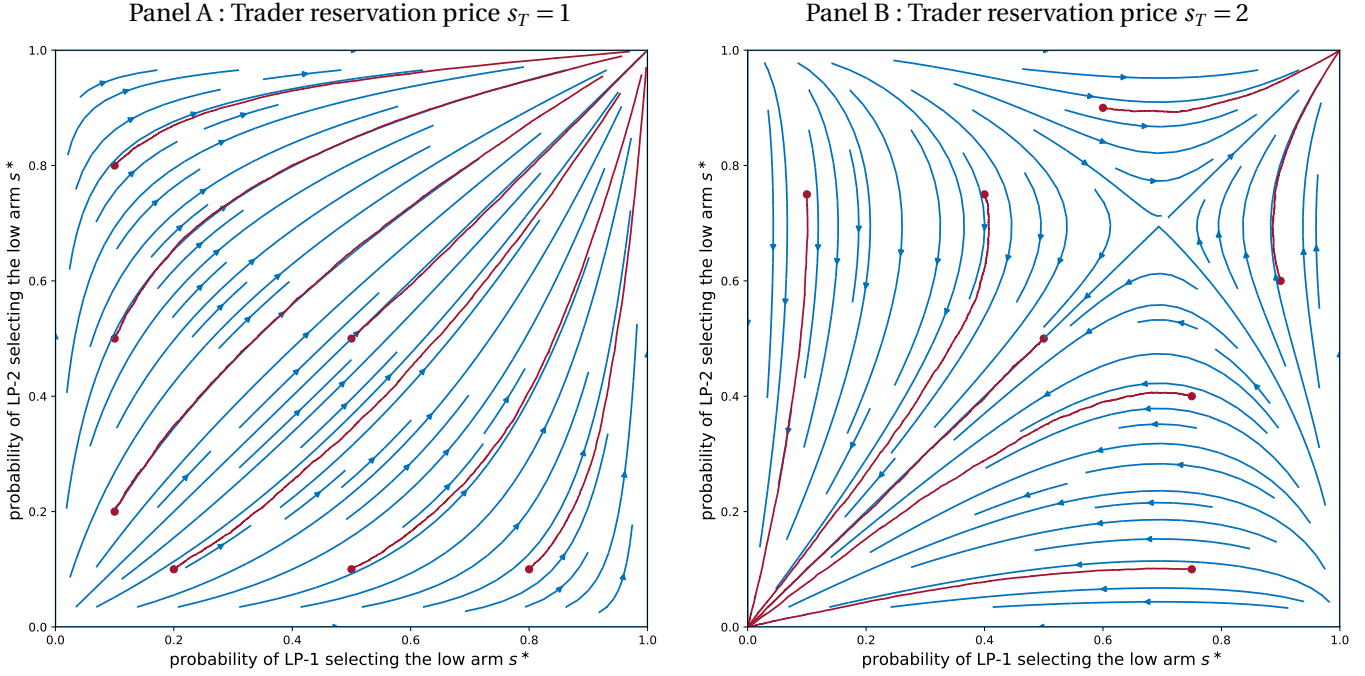
4.1 Path convergence

To illustrate the path convergence of the EXP3 algorithm to the replicator dynamics, consider the simplest setup with $N = 2$ LPs competing for a trader's flow, both LPs are simultaneously and independently optimising spread selection using the EXP3 algorithm in 2, and both select between $K = 2$ arms, one being the “low” arm s^* and the other the “high” arm s^\dagger . We now compare the replicator dynamics in Eq. (31) of Theorem 3 to the probability of arm selection in Eq. (21) and track these over time.

Figure 9 presents the results of this exercise in the form of field plots, for a learning rate $\gamma = 0.0001$ and when varying the initial arm selection probabilities across the allowable domain $[0, 1]$. The blue arrowed lines represent the trajectories of the replicator dynamics and the superimposed red lines are single runs of the actual trajectories from the EXP3 dynamics. First and foremost, we note that the blue and red lines are closely aligned, illustrating the convergence of the single path arm probabilities to the replicator dynamics. Figure 18 in Appendix A provides further examples with $N = 3$ LPs or $K = 3$ arms. Comparing Panels A and B of Figure 9 we note convergence to different arms depending on configuration and starting point. For instance, when the trader has a reservation price $s_T = 1$, irrespective of starting point, all paths converge to the point where both LPs consistently select the low arm s^* . However, as the trader loosens the reservation price to $s_T = 2$, convergence can be to either the low arm s^* or the high arm s^\dagger depending on initial conditions. For instance, if both LPs start off with a uniform prior over arm probabilities (i.e. the centre of the chart) then both converge to charge the high arm s^\dagger . However, if one or both start off with a high probability of charging the low arm s^* then convergence for both is to that point. In other words, convergence is determined by the basin of attraction that the starting point falls into. Panel D of Figure 18 in Appendix A illustrates the basins of attraction for $N = 3$ LPs: even though the model parameters are comparable, the basin of attraction towards the low arm s^* has grown in volume compared to the $N = 2$ LP case. We come back to this observation later on.

¹⁹When γ is not sufficiently small, not only does the realised path of a single run not converge to the replicator dynamics, but nor does the path averaged across many replications.

Figure 9: Replicator dynamics of the EXP3 algorithm



Note. For $N = 2$ LPs and $K = 2$ arm choices, these field plots draw in blue the theoretical gradient of arm selection probabilities as predicted by the replicator dynamics and in red single run trajectories of length 1,000,000 starting off at different points in the probability space. The learning rate $\gamma = 0.0001$ and the set of candidate spreads is (s^*, s^\dagger) . The OTC market model parameters are set to baseline values, and only the trader's reservation price s_T is varied between Panels A and B.

It is worth noting that the point at the centre of the field plots is of primary interest because it corresponds to the case where LPs adopt a uniform prior over arm selection probabilities, in line with the specification of the EXP3 Algorithm 2. But points away from the centre are also of interest because they provide insight into the convergence of the EXP3 algorithm for any arbitrary prior on the arm probabilities. In practice, the LP may have an informed prior on where to quote (e.g. based on market conditions, or spread levels offered to comparable traders, or trader feedback on market share) but equally would want this validated in a data-driven manner without pre-judging the outcome. Setting a wide range of spreads with a strong prior imposed could achieve this. The field plots are also useful to assess situations where new LPs are introduced. The existing set of LPs may already have formed strong views on which arms to draw based on their activity thus far, but the new LP might start off with a uniform prior. Any such scenarios can be located and understood within the field plots.

To conclude, we analyse what happens for alternative choices of the learning rate, namely a decaying $\gamma = 0.05t^{-1/3}$ or a “large” $\gamma = 2.5\%$ used earlier on in the paper. The results can be found in Figure 19 in Appendix A. They show –

Table 2: Expected trade valuation by arm choice and competitor action

LP1 \ LP2	Panel A : low arm convergence		Panel B : high arm convergence	
	L	H	L	H
L	8.0 8.0	19.8 1.4	2.0 2.0	5.9 0.0
H	1.4 19.8	10.0 10.0	0.0 5.9	102 102

Note. This table reports the expected trade valuations for $N = 2$ LPs when they select between $K = 2$ arms, i.e. a low arm s^* and a high arm s^\dagger . The first (second) entry in each cell is the expected reward for LP1 (LP2). In Panel A, $\rho = 0$, $s_T = 1$ which results in an equilibrium spread $s^* = 0.51$ ($s^\dagger = 1.20$) and an associated probability of trading $\theta_T = 98.1\%$ ($\theta_T = 45.9\%$). In Panel B, $\rho = 0.95$, $s_T = 5$ which results in an equilibrium spread $s^* = 0.12$ ($s^\dagger = 4.30$) and an associated probability of trading $\theta_T = 100\%$ ($\theta_T = 96.0\%$). The other model parameters are set to the baseline specification.

as expected – that with a decaying γ the EXP3 paths start off somewhat erratically but then as the learning rate drops with the passage of time they increasingly align with the replicator dynamics. For a fast learning rate, the convergence conditions are not satisfied, and we see that the individual paths vary wildly but if they start off well inside the basin of attraction, they still converge to the corresponding arm. For the paths that straddle the boundaries of neighbouring basins of attraction, convergence is unpredictable. Nevertheless, when averaging over many path simulations, Panel C of Figure 19 shows a reasonable, though not perfect, alignment with the replicator dynamics. This suggests that the replicator dynamics can still be used to approximate what convergence would look like – on average – when the learning rate is fast.²⁰

4.2 Collusion or competition in a multi-agent environment?

Let us return to the scenarios discussed in Figure 8 where, depending on model configuration, we found convergence to the low or the high arm. To provide insight to the uni-laterally optimal actions, Table 2 reports the expected trade valuations (calculated using Eq. 8 in Lemma 1) for both LPs depending on their respective static actions. Starting with Panel A, where we have convergence to the low arm, the reward matrix indeed identifies (L,L) as the Nash equilibrium, i.e. irrespective of the competitor's arm choice, the low arm is the optimal choice. We therefore have

²⁰To provide an additional perspective on this we calculate, for varying learning rates, the probability of individual EXP3 sample paths converging to the equilibrium predicted by the replicator dynamics given a random starting point in the field plot of Panel B in Figure 9. While for $\gamma = 0.0001$ we unsurprisingly obtain 100% convergence, even for the large $\gamma = 2.5\%$ the “noisy” sample paths still converge to the correct equilibrium as implied by the replicator dynamics in over 9 of 10 cases. The probability of convergence to the correct equilibrium is 97% for $\gamma = 1\%$, 93% for $\gamma = 2.5\%$ and 87% for $\gamma = 10\%$.

Table 3: LP-1 expected trade valuation by arm choice and competitor action

		arm / spread choice LP-2						
		0.06	0.12 (s^*)	1.17	2.21	3.26	4.30 (s^\dagger)	6.46
arm / spread choice LP-1	0.06	0.5	1.4	3.0	3.0	3.0	3.0	3.0
	0.12 (s^*)	0.8	2.0	5.9	5.9	5.9	5.9	5.9
	1.17	0.0	0.0	28.2 ⁷	58.3 ⁶	58.3	58.3	58.3
	2.21	0.0	0.0	0.0	54.4 ⁵	111 ⁴	111	111
	3.26	0.0	0.0	0.0	0.0	80.5 ³	163 ²	163
	4.30 (s^\dagger)	0.0	0.0	0.0	0.0	0.0	102 ¹	204
	6.46	0.0	0.0	0.0	0.0	0.0	0.0	0.1

Note. This table reports the expected trade valuations for LP-1 when there are $N = 2$ LPs that select between $K = 7$ arms on the grid indicated above. The baseline model parameters are used, except $\rho = 0.95$, $s_T = 5$. The **green cell** indicates the competitive equilibrium s^* , the **red cell** indicates the monopolistic equilibrium s^\dagger , the **blue circles** indicate the step-wise convergence to the competitive equilibrium on the discretised grid indicated by the **filled blue circle**.

convergence to the competitive equilibrium s^* . Turning to Panel B, here we see that both (L,L) and (H,H) are Nash equilibria in that once both LPs find themselves in either state, neither is individually incentivised to deviate. Because the (H,H) equilibrium is less efficient than the competitive equilibrium with continuous pricing (s^*), it is supra-competitive but by virtue of it being a Nash equilibrium it is not pseudo collusive. Note that because the algorithms play on a discretised action space, additional Nash equilibria can be introduced that may be supra- and/or super-competitive compared to s^* derived under continuous pricing.²¹

Now consider the case where we have $N = 2$ LPs, but instead of selecting from 2 arms, they can now both choose from $K = 7$ arms, i.e. $s \in \{\frac{1}{2}s^*, s^*, \dots, s^\dagger, \frac{3}{2}s^\dagger\}$ with an equidistant grid between s^* and s^\dagger . Table 3 reports the corresponding reward matrix for LP-1 (due to symmetry, the reward matrix for LP-2 is the transpose). We see that there are again two Nash equilibria – but crucially, and different from the previous case – the second one is not s^\dagger but instead $s = 1.17$. This is also the point that the replicator dynamics for the EXP3 algorithm converge to. So while the algorithm does not converge to the competitive spread s^* , the finer grid of candidate spreads leads to an equilibrium that is substantially more efficient than s^\dagger and not pseudo collusive. Table 8 in Appendix A reports the results when we increase the number of arm choices to $K = 40$. Now there is only one static Nash equilibrium, and it coincides with the competitive equilibrium s^* under continuous pricing. This is the one that is reached by the

²¹If the set of candidate spreads does not include s^* , a super-competitive Nash equilibrium may exist in the discrete game.

EXP3 algorithm.

The observed convergence of the EXP3 algorithm to a static Nash equilibrium is a finding we can prove and formalise for two LPs.

Proposition 1 *If two LPs use the EXP3 algorithm with the same finite set of candidate spreads, then the LPs' strategies converge to a pure strategy Nash equilibrium of the one-shot game for a sufficiently small value of the learning rate γ .*

Proof See Appendix D for a sketch of the proof. See [Cartea, Chang, and Penalva \(2022\)](#) for further details.

Consequently, the EXP3 algorithm does not lead to psuedo collusion amongst the LPs. While we cannot currently generalise this result to $N > 2$ LPs, if any collusive effects were to exist then they would certainly be diminished with an increase in competitors. The $N = 2$ LP case is thus the stronger result and is expected to extend to the case with more than two LPs.

Using the replicator dynamics, it is now straightforward to assess the sensitivity of the spread convergence to the model parameters and EXP3 algorithm configuration. It is convenient to express the results here in terms of the pricing efficiency metric \mathcal{E} (see Eq. 16). This effectively measures how close the EXP3 equilibrium spread on a discretised grid is to the competitive equilibrium spread s^* derived under continuous pricing. Table 4 reports this metric for varying choices of number of competing LPs N and number of candidate spreads or arm choices K . The figures in blue indicate the scenarios where the competitive spread s^* is achieved, those in red are the scenarios where the spread efficiency is below 75%, and those in orange are the scenarios in between. From here it is obvious that increasing the candidate spread granularity as well as increasing number of competing LPs increases the equilibrium spread efficiency. It should be noted that the model parameters used here are quite extreme, i.e. the model is configured to encourage supra-competitive outcomes as much as possible with a competitor price correlation of 95% and a trader reservation price that is hardly constraining and about 50 times the competitive equilibrium spread s^* . For more realistic model configurations, say with $\rho = 50\%$ and $s_T = 1$ a perfect spread efficiency is reached substantially quicker.

Thus far we have only considered symmetric configurations where both LPs choose the same grid of candidate spreads. In practice, it is exceedingly unlikely that competing LPs without any form of direct coordination would select the exact same grid (let alone use the same MAB algorithm, and start them at the exact same time with the same prior distribution). To provide some insights into this case, we consider the $N = 2$ LP case but now have LP-1 select from $K = 3$ arms while LP-2 selects from $K = 4$ arms. Figure 10 reports the evolution of arm probabilities as predicted by the replicator dynamics. We see that the LPs iteratively undercut each other until they both arrive at the competitive spread s^* . This outcome is also consistent with the reward matrices, as reported in Table 9 in Appendix

Table 4: EXP3 pricing efficiency \mathcal{E}

	Number of competing LPs					
	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N = 6$	$N \geq 7$
Number of arms	$K = 2$	0%	0%	0%	0%	100%
	$K = 3$	48%	48%	48%	48%	100%
	$K = 5$	74%	74%	74%	74%	100%
	$K = 10$	88%	88%	88%	100%	100%
	$K = 15$	93%	93%	100%	100%	100%
	$K = 20$	94%	100%	100%	100%	100%
	$K = 30$	96%	100%	100%	100%	100%
	$K \geq 40$	100%	100%	100%	100%	100%

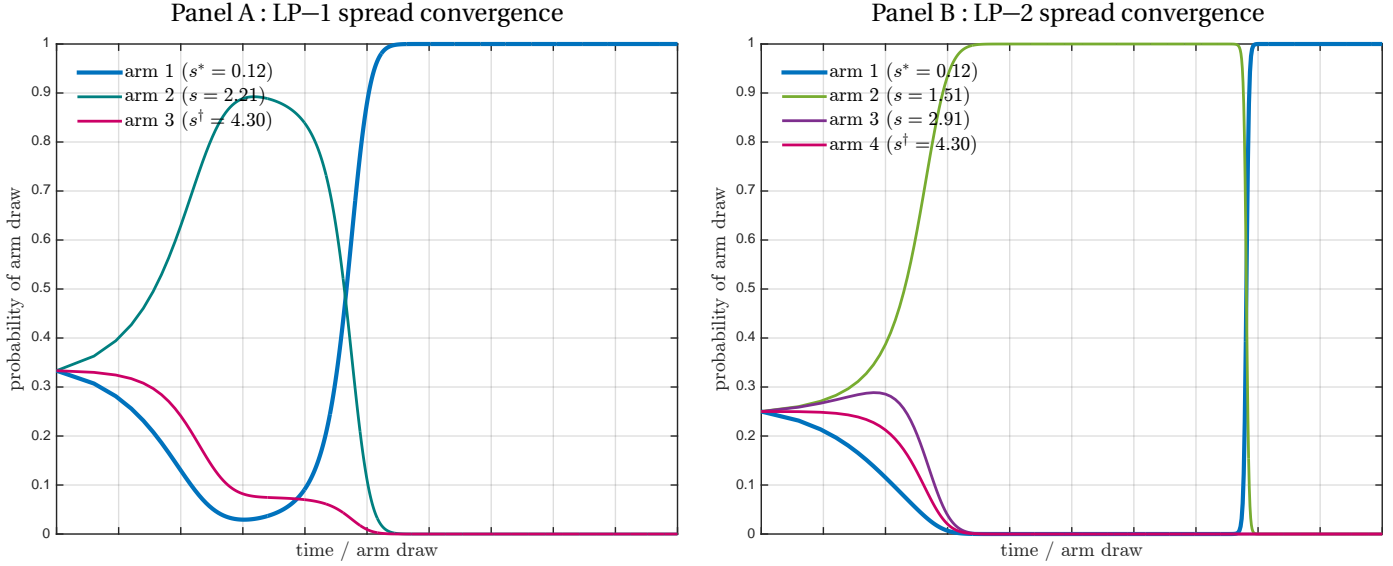
Note. This table reports the pricing efficiency (as measured by \mathcal{E} in Eq. 16) that is reached in steady state by the EXP3 algorithm as a function of number of competitors N and number of candidate spreads K . The candidate spreads are spaced equidistantly between the competitive and monopolistic spread s^* and s^\dagger . The remaining market model parameters are set to $\omega = 0.15$, $\rho = 0.95$, $s_T = 5$.

A. Recall that when both LPs use the same candidate spreads, it requires a very granular grid of 40 candidates to ensure the competitive equilibrium spread s^* is reached. But here, with only 3 and 4 arms respectively, the same outcome is arrived at.

Taken together, the results demonstrate that the supra-competitive outcomes are merely an artefact of the discretisation of the state space. Intuitively, if the grid of candidate spreads is sufficient fine, then until the competitive equilibrium s^* is reached it will always benefit the LPs to undercut their competitors: on a fine grid, the LP can undercut with a (small) sacrifice in spread capture more than offset by the gain in market share as graphically illustrated in Panel B of Figure 1. If, on the other hand, the grid of candidate spreads is coarse, then this trade-off may prematurely level out and allow supra-competitive outcomes to be sustained. Another important observation is that the equilibrium the EXP3 algorithm converges to in a multi-agent setting is a static Nash equilibrium with a pricing efficiency that improves rapidly with an increase in N (and K). Even in some of the most contrived configurations, a competitive equilibrium is reached with only a handful of competitor LPs.

As a final remark, we want to emphasise that our conclusions regarding whether or not there are collusive effects is based on the adopted definition of pseudo collusion. In related work, [Cartea, Chang, and Penalva \(2022\)](#) provide an argument that certain regions of the field plots where multiple Nash equilibria exist (e.g. Panel B of Figure 9) can be interpreted as embodying a tacitly collusive retaliatory mechanism which may drive the equilibrium selection

Figure 10: EXP3 convergence with asymmetric spread options



Note. This chart reports the evolution of arm probabilities of the EXP3 algorithm when LP-1 selects from $K = 3$ candidate spreads while LP-2 selects from $K = 4$ candidate spreads, all equally spaced between s^* and s^\dagger . The market model parameters are set to $\omega = 0.15$, $\rho = 0.95$, $s_T = 5$, while remaining ones are baseline values.

towards the less competitive one of the two. Both views have their merit and it highlights that – aside from the challenges of characterising highly complex dynamics of interacting AIs – the conclusions drawn are sensitive to the precise definition of what is meant with collusion (tacit, pseudo, or otherwise).

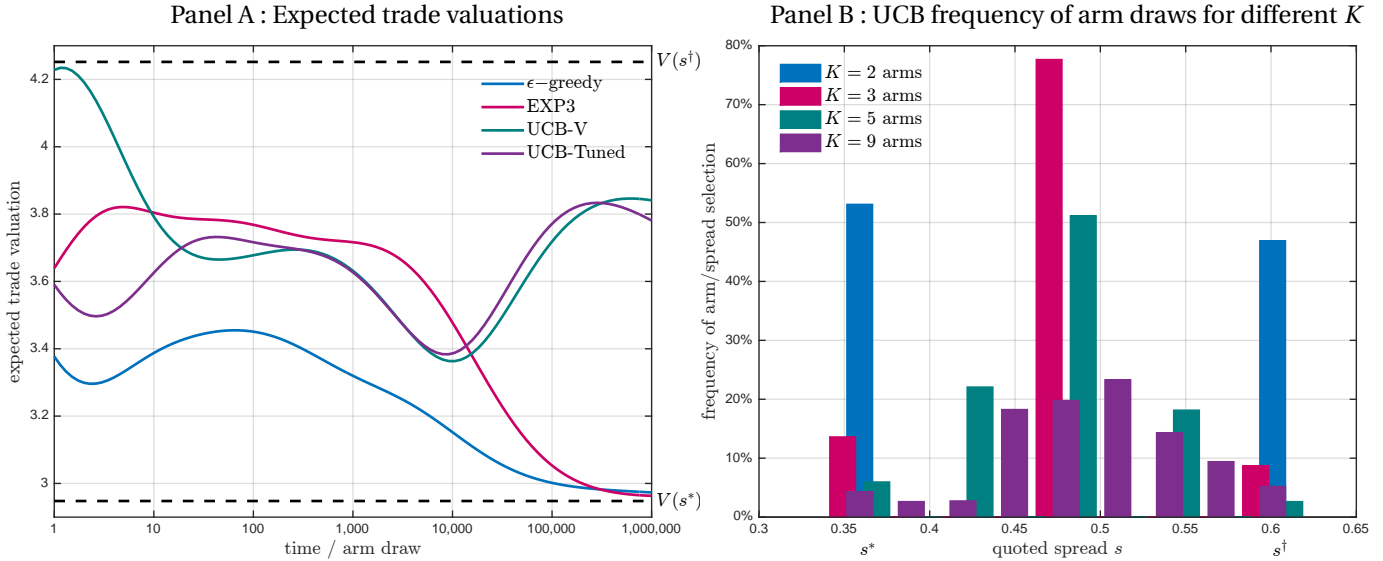
5 Discussion

The stochastic approximation techniques used in the previous section cannot be applied to any of the other MABs because their design is not compatible. Specifically, the stochastic approximation works for discrete-time stochastic processes $\{\mathcal{P}_t\}_{t \in \mathbb{N}}$ (generated by the algorithms) with increments of the form

$$\mathcal{P}_{t+1} - \mathcal{P}_t = \gamma f(\mathcal{P}_t, \pi_t), \quad (33)$$

where $\gamma > 0$ is the learning rate, $f : \mathbb{R}^{N \times K} \times E \rightarrow \mathbb{R}^{N \times K}$ is the stochastic update rule of the learning algorithms, and $\pi_t \in E$ is the random reward vector for the N agents. Because the ϵ -greedy and UCB algorithms draw arms in a deterministic manner, they do not adhere to Eq. (33). See [Cartea, Chang, and Penalva \(2022\)](#) for a detailed discussion around applying the stochastic approximation techniques to learning algorithms.

Figure 11: MAB algorithm behaviour in multi-agent system



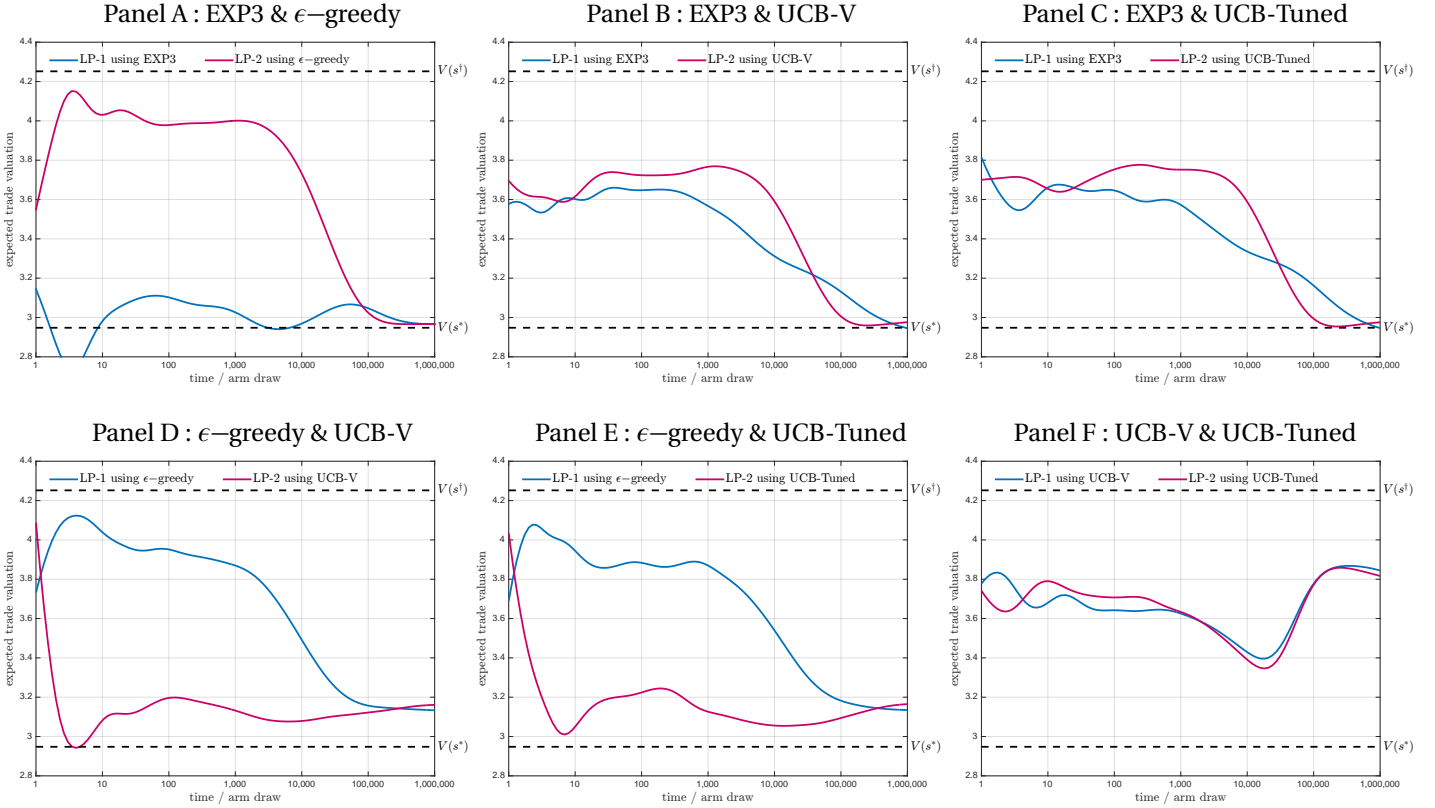
Note. Panel A displays the expected trade valuation when the LPs all use the same MAB algorithm concurrently to set their spreads. The values are calculated as a smoothed average across 1,000 independent simulation runs. The OTC market model parameters are set to baseline values, except $s_T = 0.5$. For the ϵ -greedy and EXP3 algorithms, $\epsilon = \gamma = 2.5\%$. Panel B reports the probability of arm selection in steady state for the UCB-V algorithm, as a function of candidate spread discretisation K .

Consequently, any convergence guarantees for the other MAB algorithms to a static Nash equilibria (and thus, the absence of pseudo collusion) cannot be obtained. Quite the contrary in fact. We can identify examples where the UCB algorithms converge to a supra-competitive state that is not a Nash equilibrium and is therefore evidence that genuinely pseudo collusive effects can arise when used simultaneously but independently by all LPs. Panel A of Figure 11 contains such an example. Here, we see that while the EXP3 (and the ϵ -greedy) algorithms converge to the competitive equilibrium s^* , the UCB algorithms converges to a supra-competitive state where they consistently select the middle arm of the three candidates. The expected trade valuations by arm draw and competitor action are as below, and they confirm that drawing the middle arm is uni-laterally sub-optimal for both LPs in a one-shot game.

LP1 \ LP2			
	L	M	H
L	2.9 2.9	4.3 2.6	5.5 1.8
M	2.6 4.3	3.9 3.9	5.3 2.9
H	1.8 5.5	2.9 5.3	4.3 4.3

While this example is sufficient to show pseudo collusion can arise with the UCB algorithms, we do want to verify

Figure 12: MAB algorithm behaviour in multi-agent system



Note. This chart displays the expected trade valuations when two LPs use different MAB algorithms to dynamically set spreads. The values are calculated as a smoothed average across 1,000 independent simulation runs. The OTC market model parameters are set to baseline values, except $s_T = 0.5$.

that this is not simply a rare edge case due to an unfortunate or limited choice of arms. Panel B of Figure 11 shows the frequency of arm draws in steady state as the number of arms – and hence the granularity of the spread grid – is varied. From here it is clear that the supra-competitive state the UCB algorithms select is not due to a constrained set of candidate spreads.

The finding that use of UCB algorithms can result in pseudo collusion is consistent with a recent paper by Hansen, Misra, and Pai (2021). They consider an underlying Bertrand duopoly model with two firms independently running UCB-tuned algorithms to determine what prices to quote. Profits are communicated to the firms after being perturbed by an additive noise term. The authors empirically demonstrate how the algorithms can end up quoting prices close to the monopolistic price. This is shown to happen when the noise term has small variance because it leads to a firm's rewards being more sensitive to the prices quoted by the other firm. This in turn leads to the algorithms producing strongly correlated sequences of prices, which implicitly means that a firm optimises not

only its own price, but also to some extent, its competitors' price, which clearly leads to supra- competitive prices. In the extreme case of a firm completely controlling both its own and its competitor's prices, we arrive at a collusive equilibrium. While the setup in this paper differs from that in [Hansen, Misra, and Pai \(2021\)](#), their intuition for the mechanism at play is also insightful in our model.

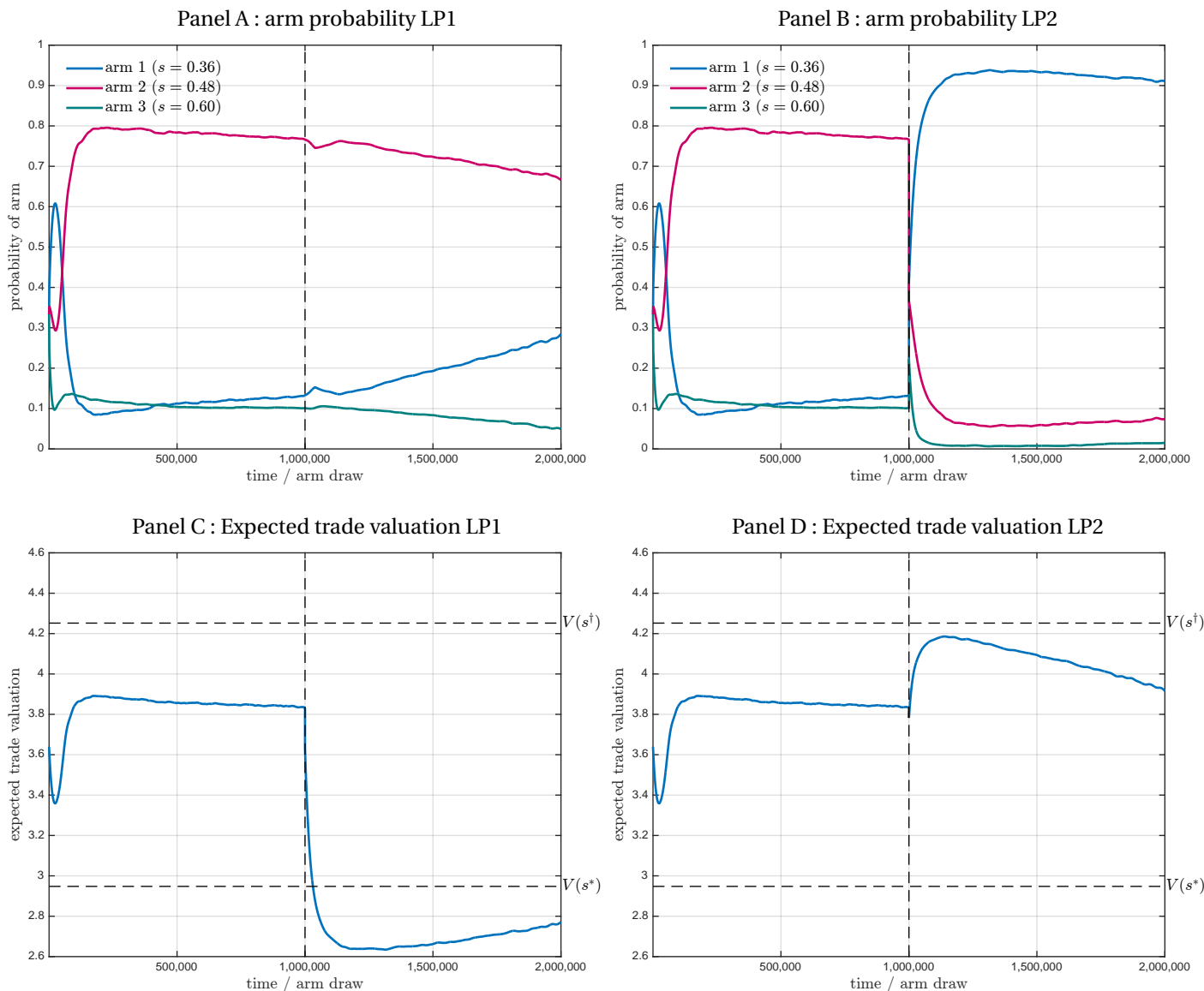
We conclude this section by analysing whether pseudo collusion is a robust feature of the UCB algorithms or a fragile one. To begin, consider what happens when competing LPs each use a different MAB algorithm to set spreads. This is of interest because the proof that the EXP3 algorithm converges to a Nash equilibrium – and similarly, the example that collusive effects can arise for the UCB algorithms – is all predicated on the assumption that all LPs use the same algorithm. The results of this analysis are reported in Figure 12. Interestingly, Panels A – C show that when any MAB algorithm competes with an EXP3 algorithm, both converge to the competitive equilibrium s^* whereas the UCB would otherwise converge to a supra-competitive and/or pseudo collusive state. Mixing the UCB algorithms with the ϵ -greedy algorithm (Panels D and E) leads to converge to a supra-competitive state that is more efficient than what the UCB algorithms would otherwise achieve but less efficient for the ϵ -greedy algorithm. Finally, when different versions of the UCB algorithms are ran simultaneously and independently by competing LPs, the convergence is similar to when the same version is used by all.

Next, consider what happens when one of the LPs resets their UCB algorithm in the middle of a run by flushing its state. Or equivalently, when the trader replaces one of the existing LPs mid-way through with a new one that uses the same MAB algorithm to optimise pricing. The results are in Figure 13. At the end of the first episode, both LPs consistently draw the pseudo collusive arm 2 (with about 80% probability), but when LP-2 resets their UCB algorithm they undercut LP-1 whose objective function is entrenched and struggles to adjust their quoted spread. Interestingly, LP-1 now earns below competitive profits at supra-competitive spreads and LP-2 on the other hand earns close to monopolistic rents at the competitive spread. Similar to the scenarios in Figures 6 and 7, adjustment to a new equilibrium is very sluggish.

Finally, we consider the sensitivity of pseudo collusion for the UCB algorithms to the number of LPs that are placed in competition. Panel A of Figure 14 draws the expected trade valuations attained in steady state by the UCB-V algorithm²² as a function of the number of LPs N . The finding is intuitive: with more LPs competing the expected trade valuation falls but – more importantly – so does the collusive effect as quantified by the pricing efficiency metric. With only two LPs, $\mathcal{E} = 33\%$ meaning that the profits attained are two-thirds towards the monopolistic equilibrium, but the efficiency rapidly improves with the addition of only a small number of additional competitors: with $N = 4$ the efficiency is already around 90% and with $N \geq 5$ the equilibrium reached is competitive and s^* is

²²The results for the UCB-Tuned algorithm are qualitatively identical and numerically very similar to those of the UCB-V algorithm, and hence left out to save space.

Figure 13: UCB algorithms' behaviour with a state reset by LP–2



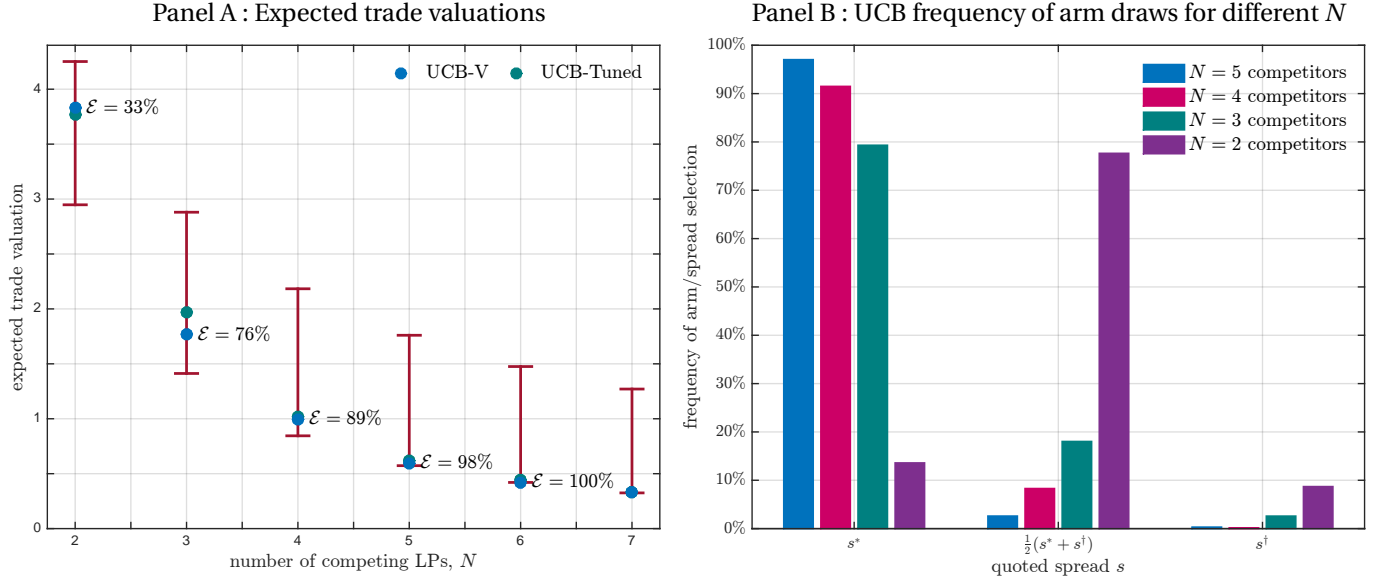
Note. This chart draws the evolution of arm probabilities and expected trade valuations for two LPs when LP–2 resets their UCB-Tuned algorithm in the middle of the sample. The OTC market model parameters are set to baseline values, except $s_T = 0.5$.

consistently selected (see Panel B) by all LPs despite their use of the UCB algorithms.²³

The finding that only a handful of competitors materially mitigates – if not, entirely removes – any potential collusive effects is important. Most of the academic literature in this field is focussed on oligopolies with a very

²³Even for the very extreme and contrived configuration of Panel B in Figure 2, with a near-perfect $\rho = 0.95$ correlation of pricing errors between LPs and a very large trader reservation value $s_T = 5$, the UCB algorithms attain over 90% pricing efficiency with 7 LPs, qualitatively similar to the EXP3 results in Table 4.

Figure 14: UCB algorithms' behaviour with increasing number of competitors



Note. Panel A draws the expected trade valuations (as blue dots •) achieved by the UCB-V and UCB-Tuned algorithm in steady state, as a function of the number of competing LPs N . The vertical red lines indicate the range of trade valuations between $V(s^*)$ at the bottom \perp and $V(s^\dagger)$ at the top \top . The pricing efficiency metric \mathcal{E} in Eq. (16) is also reported for the UCB-V algorithm. Panel B draws the frequency of UCB-V arm draws in steady state as a function of the number of competing LPs N . The OTC market model parameters are set to baseline values, except $s_T = 0.5$.

limited number – often two – competitors. While one can find examples where a single or a very small number of firms have dominant market power, this certainly does not reflect the status quo in financial markets where in virtually all segments there are many competitors that are readily accessible by traders. Advancements in electronic trading technology have lowered the barriers for market participants to connect and trade with one another to a minimum. This means that the choice of N is fully at the discretion of the trader. In related work (Oomen, 2017a,b; Butz and Oomen, 2019) we have argued for a limited number, say a handful, of LPs to be placed in competition in order to strike a balance between encouraging competition on the one hand, but avoiding effects like excessive externalisation via a Prisoner's dilemma mechanism that increase execution costs. This informal rule of thumb is now strengthened further by the argument that it also protects against any potential collusive effects that may (or may not) arise due to use of AI trading technology.

6 Conclusion

Collusion via independently but simultaneously operated AI algorithms is the subject of a quickly emerging literature and heightened regulatory interest across multiple industries. To the best of our knowledge, this is the first paper to provide an analysis of the topic as it relates to OTC financial markets. Using a simple theoretical model where liquidity providers compete for a trader's order flow, we show that the use of multi-armed bandit algorithms can lead to equilibria where supra-competitive spreads can be sustained. In some cases this is a rather mechanical consequence of the unavoidable need in practice to operate on a discrete grid of prices rather than the continuous one assumed in theory. But in other cases, it is indicative that pseudo collusive states can be sustained.

The scenarios where pseudo collusion arises are, however, quite specific and fragile and unlikely to materialise in practice for a number of reasons.

1. It is widely acknowledged that the observability of competitor prices is one of the most important enablers for collusion, tacit or otherwise (see, e.g., [Harrington, 2018](#)). In an OTC market structure, LPs do not observe their competitor prices and the risks are thereby already materially mitigated.
2. We show that the collusive effects – even if they were to exist – quickly vanish with only a moderate number of competing LPs (e.g. see Figure 14). In practice, traders have easy access to a large number of LPs and rarely limit their interactions to only one or two.²⁴ In fact, best execution regulations and/or a trader's internal policies often require a minimum of three or more competitive quotes. While public data on LP selection is scarce, [Siikanen, Nögel, and Kanninen \(2019\)](#) analyse a proprietary dataset of a large FX aggregator and find that a typical trader has an average of 5.4 LPs competing for their flow. This is already sufficient to eradicate most if not all of the collusive effects that might otherwise arise.
3. We show that pseudo collusion only arises for a certain class of MAB algorithms and to sustain it all LPs would need to use that same algorithm. Again, in practice, the chances that each LP independently selects the same algorithm to optimise their pricing are remote.
4. Even if all LPs were using the same specific algorithm, they would need to start it roughly at the same time for their exploration cycles to align. Absent that, an LP whose algorithm has settled on a spread would quickly be

²⁴For instance, in the FX market there are many dozens of LPs. One of the main trading platforms writes “With over 2,300 buy-side customers and more than 200 liquidity providers across 75 different countries, today 360T is uniquely positioned to help connect the global FX industry...” (see <https://www.360t.com>). Similarly, in the US corporate bond space “MarketAxess expanded the liquidity pool to include over 90 major and regional dealers.” (see Page 7 MarketAxess Investor Presentation 1st Quarter 2022, <https://investor.marketaxess.com>).

superseded by another LP that has just started their algorithm and through exploration will find that undercutting is the best policy (e.g. see Figure 13). Similarly, if a trader would periodically rotate selected LPs in and out of their panel, the competitor MAB algorithms would quickly run “out-of-sync” and again make any potentially collusive effects less likely to arise or sustain.

5. Pricing in the OTC market tends to be bespoke by client and instrument (and equally, traders may select different LPs for different instruments). So the level at which the MAB algorithms operate would need to be quite granular which, in turn, means only limited information is available to them. To illustrate, the MAB algorithms often take – say – 50,000 trades to converge to an equilibrium state (e.g. see Figure 13). Appendix B estimates that a representative trader in the world’s largest financial market of foreign exchange and in the most liquid EURUSD instrument still only executes about 2-10 trades per day on average. So that implies, optimistically, that it would take the MAB algorithms about 5,000 days (or 20 years) to converge if everything else remains unchanged over this period. For other, less liquid, OTC markets this would be even more extreme.
6. The trader has a material information advantage over the LPs because they observe all their prices. Also, they are in full control of the execution environment and select the LPs that can compete for their flow at any point in time. This means that collusive actions are a lot easier for the trader to identify and disrupt than it is for the LPs to effectuate.

Pseudo collusion requires favourable conditions on all these points simultaneously in order to enable it. With independently operating LPs, it would seem to require an “alignment of the stars” type scenario for it to occur. This observation is, of course, predicated on the specifics of an OTC market structure, the MAB algorithms considered here, and the assumptions made throughout. For other market structures, and alternative AI algorithms, the findings may differ.

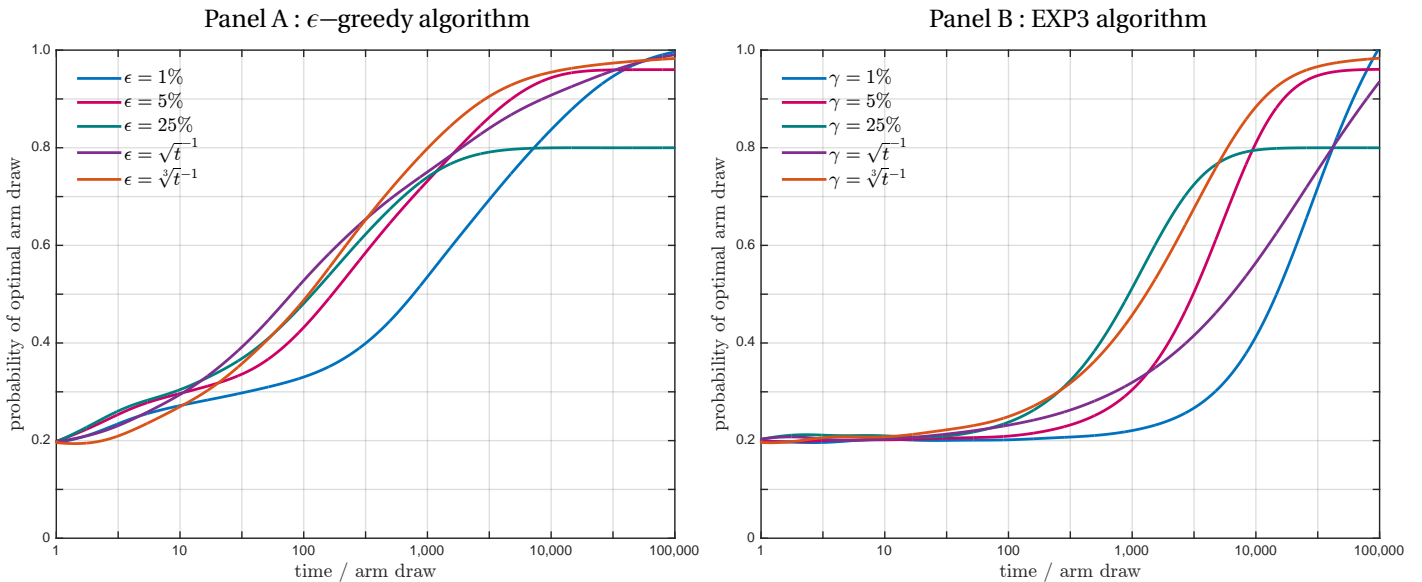
A Additional tables and figures

Table 5: Decomposition of expected trade valuation components

LP1	Expected trade valuation by arm draw						prob of trade	market share
	half-spread		prob of win		adverse selection	trade value		
	$s_1/2$		$\theta_1^{(1)}$		$\omega\sqrt{1-\rho}\mu_1^{(1)}$	\mathbb{V}_1	θ_T	$\theta_1^{(1)}/\theta_T$
arm 1	5.0	×	96.7%	+	−0.69	= 4.1	98.9%	97.8%
arm 2	15.0	×	88.1%	+	−1.85	= 11.4	96.7%	91.2%
arm 3	25.0	×	69.5%	+	−3.25	= 14.1	92.3%	75.3%
arm 4	35.0	×	43.3%	+	−3.66	= 11.5	86.6%	50.0%
arm 5	45.0	×	19.7%	+	−2.59	= 6.3	81.7%	24.1%

Note. This table provides a decomposition of the components that relate to the expected trade valuation $\mathbb{V}_i(\mathbf{s})$ in Eq. (8) by arm draw for the market model with baseline parameterisation and competitor LP quoting a spread of $s_2 = 0.7$. All figures are multiplied by 100, except for the percentages.

Figure 15: Sensitivity to hyper-parameters



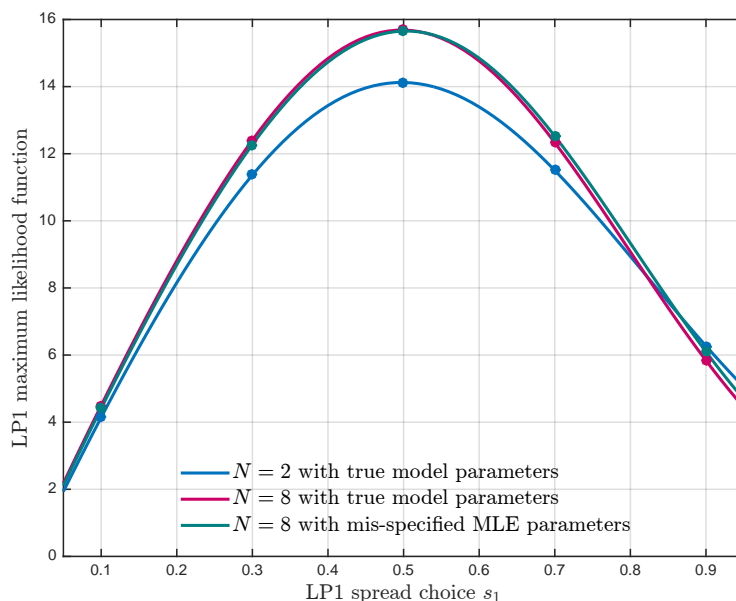
Note. This chart is a companion to Table 5 and reports the probability of drawing the optimal arm for different values of the ϵ -greedy and EXP3 algorithms' hyper-parameters ϵ and γ respectively.

Table 6: Average number of periods until convergence to the optimal arm is reached

scenario	ϵ -greedy	EXP3	UCB-V	UCB-Tuned
baseline	8,222	28,510	14,125	10,715
narrower grid ($K = 3$)	4,467	16,788	11,482	8,810
coarser grid ($K = 3$)	501	4,955	2,265	1,349
finer grid ($K = 9$)	209,894	207,014	106,660	85,507
finer grid & more competitors ($K = 9, N = 4$)	182,810	201,372	96,828	83,176
lower trade intensity ($\delta = \frac{1}{2}$)	20,184	54,325	31,989	29,174
<i>Expressed as a ratio to the baseline scenario</i>				
narrower grid ($K = 3$)	1/1.8	1/1.7	1/1.2	1/1.2
coarser grid ($K = 3$)	1/16.4	1/5.8	1/6.2	1/7.9
finer grid ($K = 9$)	25.5	7.3	7.6	8.0
finer grid & more competitors ($K = 9, N = 4$)	22.2	7.1	6.9	7.8
lower trade intensity ($\delta = \frac{1}{2}$)	2.5	1.9	2.3	2.7

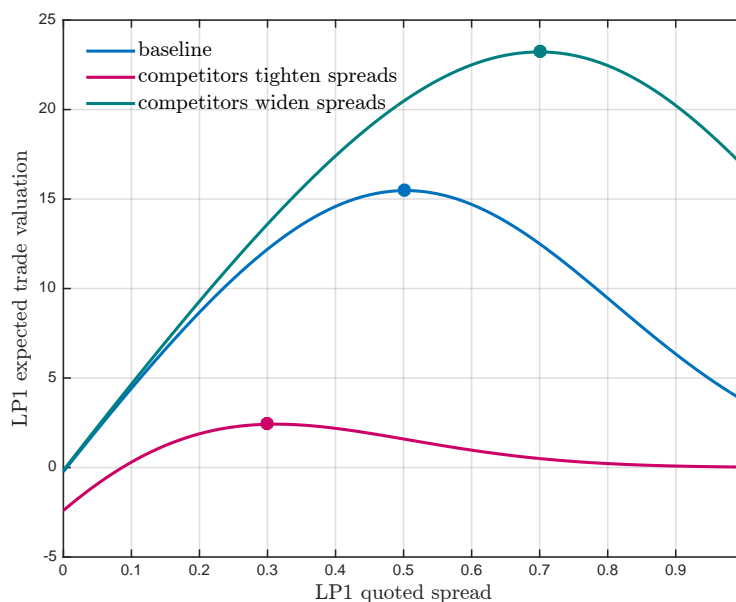
Note. This table reports the average number of periods until the MAB algorithm convergence, defined here as the first point where the probability of drawing the optimal arm is consistently over 90%. Other than K and N , the market model parameters are set to baseline values.

Figure 16: MLE under correct and incorrect model specification



Note. This chart draws the maximum likelihood function in Eq. (25) under correct model specification when $N = 2$ and when the model is mis-specified by assuming $N = 2$ whereas in reality $N = 8$. The likelihood function is evaluated at the true model parameters, as well as the estimated model parameters from Table 1 using the mis-specified likelihood function and using 100,000 observations. The OTC market model parameters are set to baseline values.

Figure 17: Trade valuation function when competitors tighten or widen spreads



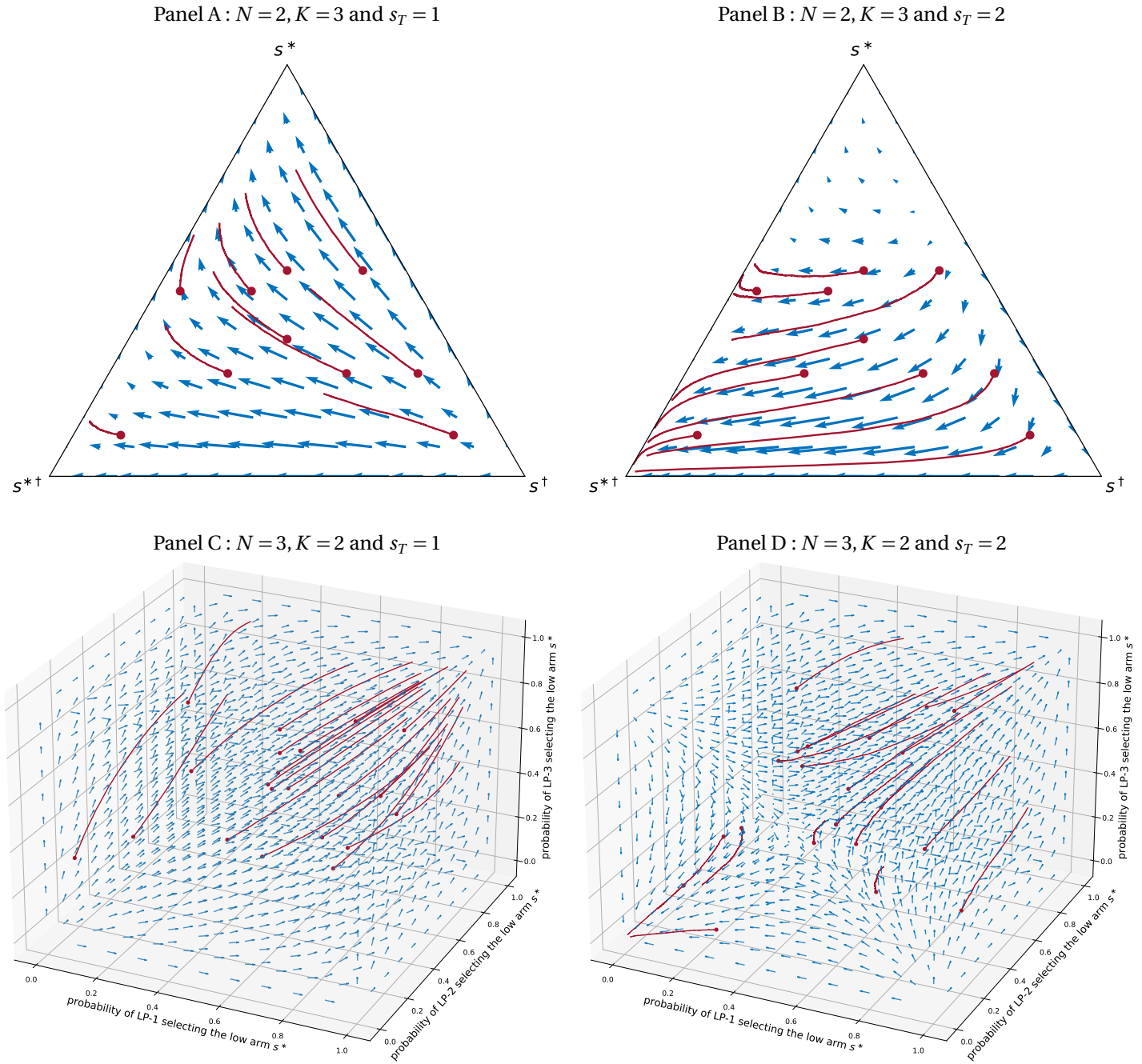
Note. This chart draws the expected trade valuation for LP1 as a function of their spread, where their competitors quote a fixed spread of 0.9 (baseline), 1.35 (widen), or 0.7 (tighten). $N = 4$ and the remaining OTC market model parameters are set to baseline values.

Table 7: Expected trade valuations and UCB objective function components

		arm draw				
		$s = 0.1$	$s = 0.3$	$s = 0.5$	$s = 0.7$	$s = 0.9$
<i>Panel A : expected trade valuations \mathbb{V}_1 ($\times 100$) for LP-1</i>						
	when competitors quote $s = 0.9$	4.421	12.198	15.473	12.500	6.341
	when competitors tighten to $s = 0.4$	0.300	2.417	1.594	0.501	0.083
	when competitors widen $s = 1.35$	4.658	13.600	20.470	23.220	20.231
<i>Panel B : UCB objective function components at $t = 100,000$ and competitors quoting $s = 0.9$</i>						
(a)	estimated average of rewards $\bar{\pi}$	4.420	12.163	15.473	12.453	6.293
	number of arm draws $n(k)$	435	1,774	95,118	2,147	527
	estimated variance of rewards $\overline{\pi^2} - \bar{\pi}^2$	2.091	1.925	2.094	2.449	1.819
(b)	$3\ln(T)/n(k)$	7.977	1.975	0.036	1.643	6.607
(c)	$\sqrt{2\ln(T)/n(k)(\overline{\pi^2} - \bar{\pi}^2)}$	3.331	1.589	0.225	1.632	2.819
	UCB-V objective function (a+b+c)	15.709	15.727	15.734	15.728	15.719

Note. This table reports a decomposition of the UCB objective function values in Eq. (23) at the end of a run of 100,000 observations, and averaged across 1,000 independent simulation runs. The market model parameters are set to baseline values.

Figure 18: Replicator dynamics for the EXP3 algorithm



Note. This chart displays the replicator dynamics field plots for the EXP3 algorithm. It is a companion to Figure 9, but for different choices of K and N . In Panels A and B, $s_1 = s_2$.

Figure 19: Convergence to replicator dynamics for alternative learning rates γ

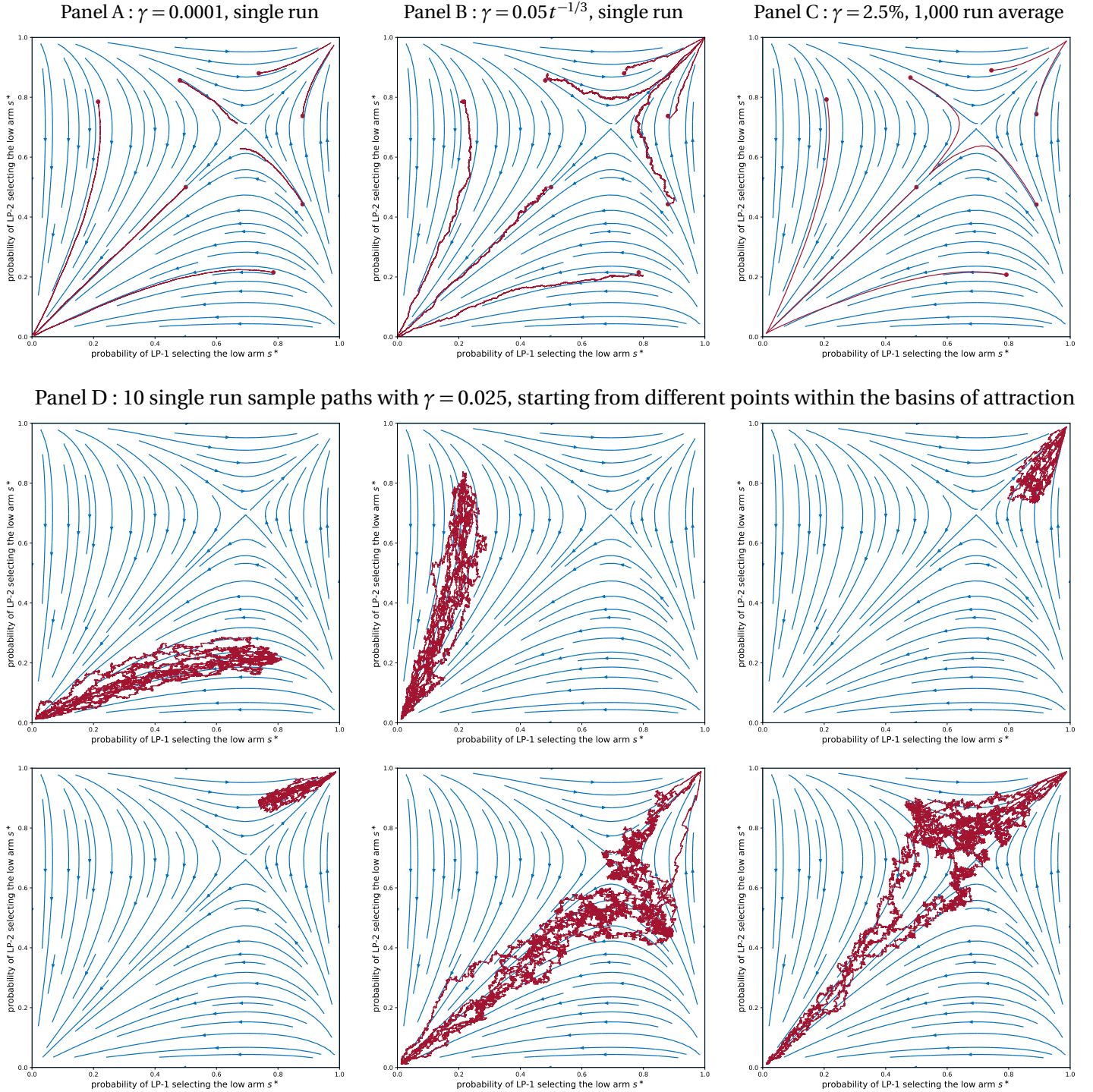


Table 8: LP-1 expected trade valuation by arm choice and competitor action

		s^*		arm / spread choice LP-2										s^\dagger	
		0.12	0.21	0.31	0.40	0.50	0.59	...	3.83	3.92	4.02	4.11	4.21	4.30	
arm / spread choice LP-1	s^* 0.12	2.0 ⁴⁹	4.43 ⁴⁸	5.7	5.9	5.9	5.9	...	5.9	5.9	5.9	5.9	5.9	5.9	
	0.21	1.1	4.40 ⁴⁷	8.4 ⁴⁶	10.3	10.7	10.7	...	10.7	10.7	10.7	10.7	10.7	10.7	
	0.31	0.2	1.9	6.8 ⁴⁵	12.4 ⁴⁴	15.0	15.4	...	15.5	15.5	15.5	15.5	15.5	15.5	
	0.40	0.0	0.3	2.6	9.2 ⁴³	16.4 ⁴²	19.6	...	20.2	20.2	20.2	20.2	20.2	20.2	
	0.50	0.0	0.0	0.4	3.4	11.5 ⁴¹	20.5 ⁴⁰	...	25.0	25.0	25.0	25.0	25.0	25.0	
	0.59	0.0	0.0	0.0	0.5	4.1	13.9 ³⁹	...	29.7	29.7	29.7	29.7	29.7	29.7	
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
	3.83	0.0	0.0	0.0	0.0	0.0	0.0	...	94.6	160	186 ⁴	191	191	191	
	3.92	0.0	0.0	0.0	0.0	0.0	0.0	...	30.3	96.7	164	191	195	195	
	4.02	0.0	0.0	0.0	0.0	0.0	0.0	...	4.4	31.0	98.7 ³	167	194	198 ²	
	4.11	0.0	0.0	0.0	0.0	0.0	0.0	...	0.3	4.4	31.5	100.3	169	197	
	4.21	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.3	4.5	32.0	102	171	
s^\dagger 4.30	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.3	4.6	32.2	102 ¹		

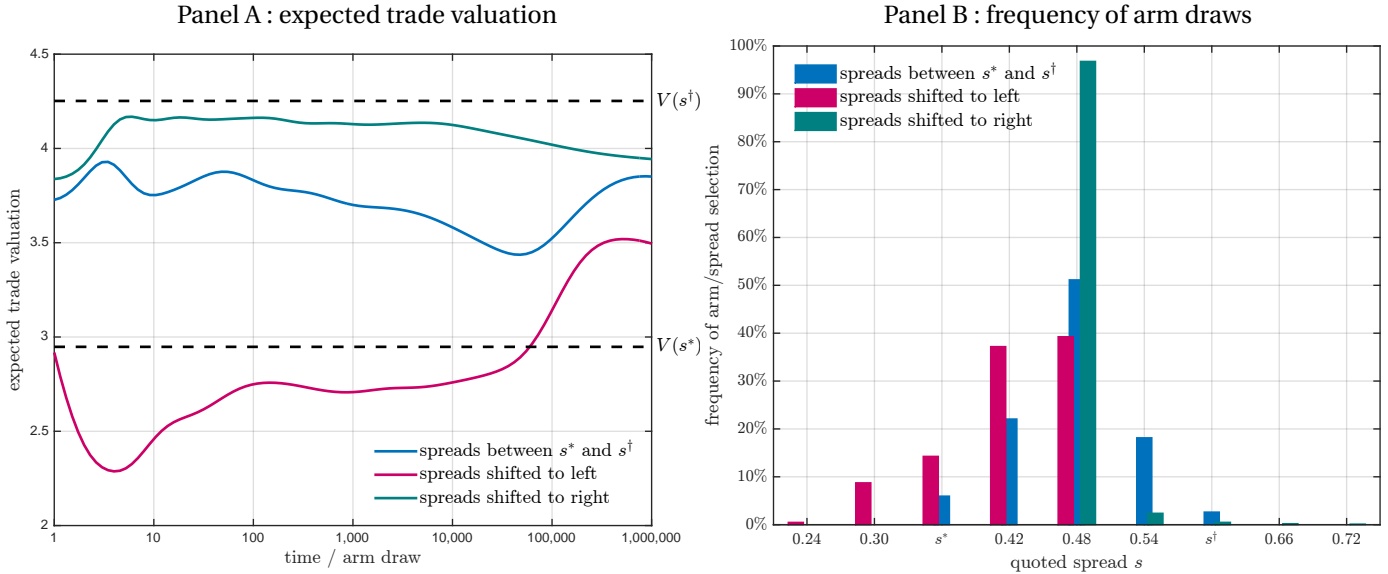
Note. This table reports the expected trade valuations for LP-1 when there are $N = 2$ LPs that select between $K = 40$ arms on the grid indicated above. The baseline model parameters are used, except $\rho = 0.95$, $s_T = 5$. The green cell indicates the competitive equilibrium s^* , the red cell indicates the monopolistic equilibrium s^\dagger , the blue circles indicate the step-wise convergence to the competitive equilibrium on the discretised grid indicated by the filled blue circle.

Table 9: LP trade valuation with asymmetric spread grids

LP1 \ LP2	LP-1 expected trade valuation				LP2 \ LP1	LP-2 expected trade valuation		
	$s^* = 0.12$	$s = 1.51$	$s = 2.91$	$s^\dagger = 4.30$		$s^* = 0.12$	$s = 2.21$	$s^\dagger = 4.30$
$s^* = 0.12$	2.0	5.9 ³	5.9	5.9	$s^* = 0.12$	2.0 ⁴	5.9	5.9
$s = 2.21$	0.0	0.0	54.4 ¹	111	$s = 1.51$	0.0	75.7 ²	75.7
$s^\dagger = 4.30$	0.0	0.0	0.0	102	$s = 2.91$	0.0	0.0	145
					$s^\dagger = 4.30$	0.0	0.0	102

Note. This table reports the expected trade valuations for $N = 2$ LPs who select from $K = 3$ and $K = 4$ candidate spreads respectively. The baseline model parameters are used, except $\rho = 0.95$, $s_T = 5$. The green cell indicates the competitive equilibrium s^* , the red cell indicates the monopolistic equilibrium s^\dagger , the blue circles indicate the step-wise convergence to the competitive equilibrium on the discretised grid indicated by the filled blue circle.

Figure 20: UCB-V behaviour for different candidate spread grids



Note. For the UCB-V algorithm with baseline market model parameters and $s_T = 0.5$, this chart draws the expected trade valuation and frequency of arm draws in steady state when the set of candidate spreads is varied. The baseline grid is equidistantly spaced between s^* and s^\dagger with $K = 5$, and the two alternative grids are shifted two increments to the left and right respectively.

B Trading activity in FX spot market

Our aim here is to establish an “order-of-magnitude” estimate for the average number of daily trades that a representative trader executes in EURUSD FX spot, one of the most actively traded and liquid markets in the world.

First, we estimate the average trade size. In the detailed Table 1a of the Bank of England’s Semi-Annual FX Turnover Survey for October 2021²⁵ participating dealers report an average daily FX spot transaction volume of \$730bn across a total of 789,346 trades, implying an average trade size of about \$925k. Another estimate is provided by Siikanen, Nögel, and Kanninen (2019) who analyse a proprietary dataset of a large FX aggregator and their Table 2 reports an average trade size of EUR920k or about \$950k. So based on this, and for convenience of presentation, we will assume an average trade size of \$1m here.

Next, we note that EURUSD represents about a quarter of all FX spot trading volumes (e.g. Table 1a of the Bank of England’s Semi-Annual FX Turnover Surveys in October 2021 reports 22.1%, Graph 1 of BIS (2019a) reports 24%).

To get data on clients we turn to one of the main FX dealing platforms 360T who report on their website that they have over 2,300 buy-side customers.²⁶ The platform’s reported trading volumes in FX spot for April 2022 are \$25,759mn²⁷. And so this implies an average number of trades in EURUSD FX spot for a representative customer of about $\$25,759 / 2,300 / \$1\text{m} \times 0.25 = 2.8$. The 360T data, however, aggregates volumes executed across a request-for-quote or RFQ protocol and streaming liquidity API trading. The latter is expected to contribute a higher trade count but it is not possible to disentangle this from the data available. However, the FX aggregator data analysed in Siikanen, Nögel, and Kanninen (2019) is exclusively streaming liquidity for EURUSD FX spot and they report a total of 10,000 trades over a 10 day period by 100 customers. This implies an average number of trades in EURUSD FX spot for a representative customer of about $10,000 / 10 / 100 = 10$.

C Trader’s transaction costs

The trader’s effective transaction cost associated with their execution (not considering the opportunity cost when having liquidity demand and not acting on this demand) is given by

$$\pi_t^{(T)}(s) = \left(\min \left\{ \min_j \{a_t^{(j)}(s_j)\}, a_t^{(T)}(s_T) \right\} - p_t^* \right) \mathbb{1}_{D_t=1} + \left(p_t^* - \max \left\{ \max_j \{b_t^{(j)}(s_j)\}, b_t^{(T)}(s_T) \right\} \right) \mathbb{1}_{D_t=-1}. \quad (34)$$

The expected cost is:

$$\mathbb{V}_T(s) \equiv \mathbb{E}[\pi_t^{(T)}(s)] = \delta \left(\frac{s_T}{2} (1 - \theta_T^{(1)}(s)) - \omega_T (1 - \theta_T^{(2)}(s)) \right) + \sum_{i=1}^N \mathbb{V}_i(s), \quad (35)$$

²⁵Available at <https://www.bankofengland.co.uk/markets/london-foreign-exchange-joint-standing-committee/results-of-the-fxjsc-turnover-survey-for-october-2021>

²⁶See <https://www.360t.com> “With over 2,300 buy-side customers and more than 200 liquidity providers across 75 different countries, today 360T is uniquely positioned to help connect the global FX industry via our proprietary suite of web-based technology solutions.”

²⁷See <https://www.360t.com/volumes-overview/360t-daily-trading-volumes>

where

$$\theta_T^{(k)}(s) = 1 - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} z^{k-1} f_T(x, z, s) \phi(x) \phi(z) dx dz, \quad (36)$$

$$f_T(x, z, s) = \prod_{i=1}^N \Phi\left(\frac{\omega_T z - \omega \sqrt{\rho} x + \frac{1}{2}(s_i - s_T)}{\omega \sqrt{1-\rho}}\right). \quad (37)$$

D Proofs

Proof of Lemma 1 and the trader's expected cost in Eq. 35 We note directly from the definition of $\pi_t^{(i)}(s)$ that

$$\begin{aligned} \mathbb{V}_i(s) &= \mathbb{E}\left[\left(a_t^{(i)}(s_i) - p_t^*\right) \mathbb{1}_{a_t^{(i)}(s_i) < \min\{\min_{j \neq i}\{a_t^{(j)}(s_j)\}, a_t^{(T)}(s_T)\}, D_t=1}\right] + \mathbb{E}\left[\left(p_t^* - b_t^{(i)}(s_i)\right) \mathbb{1}_{b_t^{(i)}(s_i) > \max\{\max_{j \neq i}\{b_t^{(j)}(s_j)\}, b_t^{(T)}(s_T)\}, D_t=-1}\right], \\ &= \delta \mathbb{E}\left[\left(p_t^* - b_t^{(i)}(s_i)\right) \mathbb{1}_{b_t^{(i)}(s_i) > \max\{\max_{j \neq i}\{b_t^{(j)}(s_j)\}, b_t^{(T)}(s_T)\}}\right], \end{aligned}$$

where the second equality is due to the symmetry between buying and selling in the model. At this point, it is convenient to introduce independent normal random variables $u_0, u_1, \dots, u_N, u_T$ and write

$$p_t^{(j)} - p_t^* = \omega \sqrt{\rho} u_0 + \omega \sqrt{1-\rho} u_j$$

for $j = 1, \dots, N$ as well as $p_t^{(T)} - p_t^* = \omega_T u_T$. With such a representation of the pricing errors, it is clear that

$$\begin{aligned} \mathbb{V}_i(s) &= \delta \mathbb{E}\left[\left(p_t^* - \left(p_t^{(i)} - \frac{s_i}{2}\right)\right) \mathbb{1}_{p_t^{(i)} - \frac{s_i}{2} > \max\{\max_{j \neq i}\{p_t^{(j)} - \frac{s_j}{2}\}, p_t^{(T)} - \frac{s_T}{2}\}}\right], \\ &= \delta \mathbb{E}\left[\left(\frac{s_i}{2} - \omega \sqrt{\rho} u_0 - \omega \sqrt{1-\rho} u_i\right) \mathbb{1}_{u_i + \frac{s_j - s_i}{2\omega \sqrt{1-\rho}} > u_j \ \forall \ j \neq i, \frac{\omega_T u_0 + \omega \sqrt{\rho} u_i}{\omega_T} + \frac{s_T - s_i}{2\omega_T} > u_T}\right], \\ &= \delta \frac{s_i}{2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_i(x, y, s) \phi(x) \phi(y) dx dy - \delta \omega \sqrt{1-\rho} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(y + \sqrt{\frac{\rho}{1-\rho}} x\right) f_i(x, y, s) \phi(x) \phi(y) dx dy, \\ &= \delta \left(\frac{s_i}{2} \theta_i^{(1)}(s) - \omega \sqrt{1-\rho} \mu_i^{(1)}(s)\right). \end{aligned}$$

Using similar arguments as above, we deduce Eq. (35) directly from the definition of $\pi_t^{(T)}(s)$

$$\begin{aligned} \mathbb{V}_T(s) - \sum_{i=1}^N \mathbb{V}_i(s) &= \mathbb{E}\left[\left(a_t^{(T)}(s_T) - p_t^*\right) \mathbb{1}_{a_t^{(T)}(s_T) < \min_j\{a_t^{(j)}(s_j)\}, D_t=1}\right] + \mathbb{E}\left[\left(p_t^* - b_t^{(T)}(s_T)\right) \mathbb{1}_{b_t^{(T)}(s_T) > \max_j\{b_t^{(j)}(s_j)\}, D_t=-1}\right], \\ &= \delta \mathbb{E}\left[\left(p_t^* - b_t^{(T)}(s_T)\right) \mathbb{1}_{b_t^{(T)}(s_T) > \max_j\{b_t^{(j)}(s_j)\}}\right], \\ &= \delta \mathbb{E}\left[\left(p_t^* - \left(p_t^{(T)} - \frac{s_T}{2}\right)\right) \mathbb{1}_{p_t^{(T)} - \frac{s_T}{2} > \max_j\{p_t^{(j)} - \frac{s_j}{2}\}}\right], \\ &= \delta \mathbb{E}\left[\left(\frac{s_T}{2} - \omega_T u_T\right) \mathbb{1}_{\frac{\omega_T u_T - \omega \sqrt{\rho} u_0 + \frac{s_j - s_T}{2}}{\omega \sqrt{1-\rho}} > u_j \ \forall \ j}\right], \\ &= \delta \frac{s_T}{2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_T(x, z, s) \phi(x) \phi(z) dx dz - \delta \omega_T \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} z f_T(x, z, s) \phi(x) \phi(z) dx dz, \\ &= \delta \left(\frac{s_T}{2} (1 - \theta_T^{(1)}(s)) - \omega_T (1 - \theta_T^{(2)}(s))\right). \end{aligned}$$

■

Proof of Theorems 1 and 2 Let us consider the i^{th} LP's objective under the assumption that the remaining $N-1$ LPs are quoting the spread $s_{\neq i}$

$$\begin{aligned}\mathbb{V}_i(s) &= \delta \left(\frac{s_i}{2} \theta_i^{(1)}(s) - \omega \sqrt{1-\rho} \mu_i^{(1)}(s) \right) \\ &= \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s_i}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s_i}{2\omega_T} \right) \Phi \left(y + \frac{s_{\neq i} - s_i}{2\omega \sqrt{1-\rho}} \right)^{N-1} \phi(x) \phi(y) dx dy.\end{aligned}$$

Competitive equilibrium. In this case, we calculate

$$\mathbb{V}_i(s) = \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s_{\neq i}}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s_{\neq i}}{2\omega_T} \right) \Phi(y)^{N-1} \phi(x) \phi \left(y + \frac{s_i - s_{\neq i}}{2\omega \sqrt{1-\rho}} \right) dx dy,$$

where the equality follows from the substitution

$$y \mapsto y + \frac{s_i - s_{\neq i}}{2\omega \sqrt{1-\rho}}.$$

Differentiating with respect to s_i , we obtain

$$\begin{aligned}\frac{\partial \mathbb{V}_i}{\partial s_i}(s) &= -\frac{1}{2\omega \sqrt{1-\rho}} \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s_{\neq i}}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \left(y + \frac{s_i - s_{\neq i}}{2\omega \sqrt{1-\rho}} \right) \dots \\ &\quad \dots \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s_{\neq i}}{2\omega_T} \right) \Phi(y)^{N-1} \phi(x) \phi \left(y + \frac{s_i - s_{\neq i}}{2\omega \sqrt{1-\rho}} \right) dx dy.\end{aligned}$$

Setting $\frac{\partial \mathbb{V}_i}{\partial s_i}(s) = 0$ and substituting $s^* := s_i = s_{\neq i}$, we obtain a necessary condition for a competitive equilibrium spread s^*

$$0 = -\frac{1}{2\omega \sqrt{1-\rho}} \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s^*}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) y \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s^*}{2\omega_T} \right) \Phi(y)^{N-1} \phi(x) \phi(y) dx dy.$$

Solving for s^* gives the claimed result.

Monopolistic equilibrium In this case, we first set $s_i = s_{\neq i}$ and obtain

$$\mathbb{V}_i(s) = \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s_i}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s_i}{2\omega_T} \right) \Phi(y)^{N-1} \phi(x) \phi(y) dx dy.$$

Using the substitution

$$x \mapsto x + \frac{s_i - \tilde{s}_i}{2\omega \sqrt{1-\rho}},$$

where $\tilde{s}_i > 0$ is an arbitrary constant, we have

$$\mathbb{V}_i(s) = \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{\tilde{s}_i}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - \tilde{s}_i}{2\omega_T} \right) \Phi(y)^{N-1} \phi \left(x + \frac{s_i - \tilde{s}_i}{2\omega \sqrt{1-\rho}} \right) \phi(y) dx dy.$$

Differentiating with respect to s_i , we obtain

$$\begin{aligned}\frac{\partial \mathbb{V}_i}{\partial s_i}(s) &= -\frac{1}{2\omega \sqrt{1-\rho}} \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{\tilde{s}_i}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) \left(x + \frac{s_i - \tilde{s}_i}{2\omega \sqrt{1-\rho}} \right) \dots \\ &\quad \dots \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - \tilde{s}_i}{2\omega_T} \right) \Phi(y)^{N-1} \phi \left(x + \frac{s_i - \tilde{s}_i}{2\omega \sqrt{1-\rho}} \right) \phi(y) dx dy.\end{aligned}$$

Setting $\frac{\partial \mathbb{V}_i}{\partial s_i}(s) = 0$ and substituting $s^\dagger := \tilde{s}_i = s_i$, we arrive at a necessary condition for a monopolistic equilibrium spread s^\dagger

$$0 = -\frac{1}{2\omega \sqrt{1-\rho}} \delta \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(\frac{s^\dagger}{2} - \omega \sqrt{1-\rho} y - \omega \sqrt{\rho} x \right) x \Phi \left(\frac{\omega \sqrt{\rho} x + \omega \sqrt{1-\rho} y}{\omega_T} + \frac{s_T - s^\dagger}{2\omega_T} \right) \Phi(y)^{N-1} \phi(x) \phi(y) dx dy.$$

Solving for s^\dagger gives the claimed result. ■

Proof of the likelihood function in Eq. 25. Since the observations recorded at different times $(m_1^{(1)}, D_1^{(1)}), \dots, (m_T^{(1)}, D_T^{(1)})$ are independent, it is clear that the joint likelihood is a product of individual contributions that depend only on $(m_t^{(1)}, D_t^{(1)})$, where $t \in \{1, \dots, T\}$. Let us now consider time period $t \in \{1, \dots, T\}$ and a candidate parameter set $\Theta = (\delta, \delta_B, \omega, \rho, s_2, \omega_T, s_T)$. Let us also introduce standard normal random variables u_0, u_1, u_2, u_T and write

$$m_t^{(j)} = \omega \sqrt{\rho} u_0 + \omega \sqrt{1 - \rho} u_j$$

for $j = 1, 2$ as well as $m_t^{(T)} = \omega_T u_T$. We now make a crucial observation (which holds in distribution):

$$m_t^{(2)} \left| \left(m_t^{(1)} = m \right) = \rho m + \omega \sqrt{1 - \rho^2} u_c,$$

where u_c is a standard normal random variable. Using this observation, we can calculate the likelihood contribution

$$\begin{aligned} L(m, 1|\Theta, s) &\equiv \mathbb{P}(m_t^{(1)} = m, D_t^{(1)} = 1), \\ &= \mathbb{P}(D_t = 1) \mathbb{P}(m_t^{(1)} = m) \mathbb{P}\left(m_t^{(1)} + \frac{s}{2} < m_t^{(2)} + \frac{s_2}{2} \left| m_t^{(1)} = m\right.\right) \mathbb{P}\left(m_t^{(1)} + \frac{s}{2} < m_t^{(T)} + \frac{s_T}{2} \left| m_t^{(1)} = m\right.\right), \\ &= \delta \delta_B \mathbb{P}(\omega \sqrt{\rho} u_0 + \omega \sqrt{1 - \rho} u_1 = m) \mathbb{P}\left(m + \frac{s}{2} < \rho m + \omega \sqrt{1 - \rho^2} u_c + \frac{s_2}{2}\right) \mathbb{P}\left(m + \frac{s}{2} < \omega_T u_T + \frac{s_T}{2}\right), \\ &= \frac{\delta \delta_B}{\omega} \phi\left(\frac{m}{\omega}\right) \left(1 - \Phi\left(\frac{(1 - \rho)m + \frac{s - s_2}{2}}{\omega \sqrt{1 - \rho^2}}\right)\right) \left(1 - \Phi\left(\frac{m + \frac{s - s_T}{2}}{\omega_T}\right)\right). \end{aligned}$$

The likelihood contribution $L(m, -1|\Theta, s)$ is calculated very similarly. Finally, we note that

$$\begin{aligned} L(m, 0|\Theta, s) &\equiv \mathbb{P}(m_t^{(1)} = m, D_t^{(1)} = 0), \\ &= \mathbb{P}(m_t^{(1)} = m) - \mathbb{P}(m_t^{(1)} = m, D_t^{(1)} = 1) - \mathbb{P}(m_t^{(1)} = m, D_t^{(1)} = -1), \\ &= \frac{1}{\omega} \phi\left(\frac{m}{\omega}\right) - L(m, 1|\Theta, s) - L(m, -1|\Theta, s). \end{aligned}$$

■

Proof of Lemma 2 First, consider the case where $k = k_{i,t}^*$. Write the differenced equation as

$$\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) = \frac{(1 - \gamma) w_{i,t}(k) A_{k,t}^\gamma}{\sum_\ell w_{i,t}(\ell) + w_{i,t}(k) (A_{k,t}^\gamma - 1)} - \frac{(1 - \gamma) w_{i,t}(k)}{\sum_\ell w_{i,t}(\ell)}, \quad (38)$$

where $A_{k,t}^\gamma = \exp\left(\frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right)$. Substitute $(1 - \gamma) w_{i,t}(k) = \sum_\ell w_{i,t}(\ell) [\mathcal{P}_{i,t}(k) - \gamma/K]$ in Eq. (38) and write

$$\begin{aligned} \mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) &= \frac{\sum_\ell w_{i,t}(\ell) [\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}] A_{k,t}^\gamma}{\sum_\ell w_{i,t}(\ell) + \sum_\ell w_{i,t}(\ell) \frac{\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}}{1 - \gamma} (A_{k,t}^\gamma - 1)} - \frac{\sum_\ell w_{i,t}(\ell) [\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}]}{\sum_\ell w_{i,t}(\ell)}, \\ &= \frac{(1 - \gamma) (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}) A_{k,t}^\gamma}{(1 - \gamma) + (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}) (A_{k,t}^\gamma - 1)} - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right), \\ &= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right) \left[\frac{(1 - \gamma) A_{k,t}^\gamma - [(1 - \gamma) + (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}) (A_{k,t}^\gamma - 1)]}{1 - \gamma + (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}) (A_{k,t}^\gamma - 1)} \right], \\ &= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right) (A_{k,t}^\gamma - 1) \left[\frac{(1 - \gamma) - (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K})}{1 - \gamma + (\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}) (A_{k,t}^\gamma - 1)} \right]. \end{aligned}$$

Use geometric sums to obtain

$$\begin{aligned}
\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) &= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) (A_{k,t}^\gamma - 1) \left[(1 - \gamma) - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) \right] \left[1 + \gamma - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) (A_{k,t}^\gamma - 1) + O(\gamma^2) \right], \\
&= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) (A_{k,t}^\gamma - 1) \left[1 - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) \right] + O(\gamma^2), \\
&= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) (A_{k,t}^\gamma - 1) - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right)^2 (A_{k,t}^\gamma - 1) + O(\gamma^2), \\
&= \mathcal{P}_{i,t}(k) (A_{k,t}^\gamma - 1) - \mathcal{P}_{i,t}^2(k) (A_{k,t}^\gamma - 1) + O(\gamma^2).
\end{aligned}$$

Finally, expand the exponential function with the power series $A_{k,t}^\gamma = 1 + \frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k)K} + O(\gamma^2)$ to obtain

$$\begin{aligned}
\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) &= \frac{\gamma \pi_{i,t}(s(k))}{K} - \frac{\gamma \mathcal{P}_{i,t}(k) \pi_{i,t}(s(k))}{K} + O(\gamma^2), \\
&= \frac{\gamma \pi_{i,t}(s(k))}{K} (1 - \mathcal{P}_{i,t}(k)) + O(\gamma^2).
\end{aligned}$$

Similarly, consider the case where $k \neq k_{i,t}^* = \ell$, we get

$$\begin{aligned}
\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) &= \frac{\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}}{1 + \frac{\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K}}{1 - \gamma} (A_{\ell,t}^\gamma - 1)} - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right), \\
&= \frac{(1 - \gamma) \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right)}{(1 - \gamma) + \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1)} - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right), \\
&= \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) \left[\frac{(1 - \gamma) - [(1 - \gamma) + \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1)]}{(1 - \gamma) + \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1)} \right], \\
&= - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1) \left[1 + \gamma - \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1) + O(\gamma^2) \right], \\
&= - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K} \right) \left(\mathcal{P}_{i,t}(\ell) - \frac{\gamma}{K} \right) (A_{\ell,t}^\gamma - 1) + O(\gamma^2), \\
&= - \mathcal{P}_{i,t}(k) \mathcal{P}_{i,t}(\ell) (A_{\ell,t}^\gamma - 1) + O(\gamma^2), \\
&= - \frac{\gamma \pi_{i,t}(s(\ell))}{K} \mathcal{P}_{i,t}(k) + O(\gamma^2).
\end{aligned}$$

■

Proof of Theorem 3 To compute the expected increment of \mathcal{P} , we apply Lemma 2 to write the innovations as

$$\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) = \begin{cases} \frac{\gamma \pi_{i,t}(s(k))}{K} (1 - \mathcal{P}_{i,t}(k)) + O(\gamma^2), & k = k_{i,t}^* \\ - \frac{\gamma \pi_{i,t}(s(\ell))}{K} \mathcal{P}_{i,t}(k) + O(\gamma^2), & k \neq k_{i,t}^* = \ell. \end{cases}$$

The reward received depends on the spread quoted by LP- i , the remaining LPs, and the stochasticity from the model. We neglect terms of order $O(\gamma^2)$ and compute the expectation of the innovation respect to the rewards that can be received in three steps. First, take the expectation conditional on the actions of all the LPs. Second, take the expectation over the actions of the competing LPs. Finally, take the expectation with respect to LP- i 's own actions.

Step 1. The expected trade valuation of the trader's flow won by LP- i conditional on LP- i quoting $s(k_{i,t}^*)$ and the remaining LPs quoting s_{-i} is given by $\mathbb{V}_{i,t}(s(k_{i,t}^*), s_{-i})$. Next, write the expected increment of the policies conditional on actions $s(k_{i,t}^*)$ and s_{-i}

as

$$\mathbb{E}[\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) | s(k_{i,t}^*), s_{-i}] = \begin{cases} \frac{\gamma}{K} \mathbb{V}_{i,t}(s(k), s_{-i})(1 - \mathcal{P}_{i,t}(k)) & k = k_{i,t}^*, \\ -\frac{\gamma}{K} \mathbb{V}_{i,t}(s(\ell), s_{-i}) \mathcal{P}_{i,t}(k) & k \neq k_{i,t}^* = \ell. \end{cases}$$

Step 2. The expected trade valuation of the trader's flow won by LP- i for LP- i quoting $s(k_{i,t}^*)$ depends on the actions of the remaining LPs, and the actions of the remaining LPs depend on their probability of quoting a particular spread which varies with t . Therefore, take the expectation over the actions of the opponents to obtain the conditional expected payoff $\bar{\mathbb{V}}_{i,t}(s(k_{i,t}^*)) = \mathbb{E}_{\mathcal{P}_{-i}}[\mathbb{V}_{i,t}(s(k_{i,t}^*), s_{-i})]$ for LP- i quoting $s(k_{i,t}^*)$. Thus, the expected increment of the policies conditional on the action of agent i is

$$\mathbb{E}[\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k) | s(k_{i,t}^*)] = \begin{cases} \frac{\gamma}{K} \bar{\mathbb{V}}_{i,t}(s(k))(1 - \mathcal{P}_{i,t}(k)) & k = k_{i,t}^*, \\ -\frac{\gamma}{K} \bar{\mathbb{V}}_{i,t}(s(\ell)) \mathcal{P}_{i,t}(k) & k \neq k_{i,t}^* = \ell. \end{cases} \quad (39)$$

Step 3. Take the expectation over the action of agent i and use the two cases in Eq. (39) to write the expected increment of the policies:

$$\begin{aligned} \mathbb{E}[\mathcal{P}_{i,t+1}(k) - \mathcal{P}_{i,t}(k)] &= \frac{\gamma}{K} \mathcal{P}_{i,t}(k) \bar{\mathbb{V}}_{i,t}(s(k))(1 - \mathcal{P}_{i,t}(k)) - \frac{\gamma}{K} \sum_{\ell \neq k} \mathcal{P}_{i,t}(\ell) \bar{\mathbb{V}}_{i,t}(s(\ell)) \mathcal{P}_{i,t}(k) \\ &= \frac{\gamma}{K} \mathcal{P}_{i,t}(k) \left[\bar{\mathbb{V}}_{i,t}(s(k))(1 - \mathcal{P}_{i,t}(k)) - \sum_{\ell \neq k} \mathcal{P}_{i,t}(\ell) \bar{\mathbb{V}}_{i,t}(s(\ell)) \right] \\ &= \frac{\gamma}{K} \mathcal{P}_{i,t}(k) \left[\bar{\mathbb{V}}_{i,t}(s(k)) - \sum_{\ell} \mathcal{P}_{i,t}(\ell) \bar{\mathbb{V}}_{i,t}(s(\ell)) \right]. \end{aligned} \quad (40)$$

Finally, if we divide Eq. (40) by γ and as $\gamma \rightarrow 0$, then Eq. (40) is a continuous globally integrable vector field that describes the expected increment of \mathcal{P} .

To verify that the continuous-time affine interpolated process $\hat{\mathcal{P}}$ is a pseudo-trajectory of the flow induced by $\dot{\mathcal{P}}$, following Proposition 4.1 in Benaïm (1999), we verify that the following are true:

$$\limsup_{\gamma \rightarrow 0} \|\gamma U_t\| = 0, \quad (C1)$$

$$\sup_t \|\mathcal{P}_t\| < \infty, \quad (C2)$$

where $\gamma U_t = O(\gamma^2) + \gamma \tilde{U}_t$ and the entries of $\gamma \tilde{U}_t$ are given by

$$\gamma \tilde{U}_{i,t}(k) = \begin{cases} \frac{\gamma}{K} \left(\pi_{i,t}(s(k))(1 - \mathcal{P}_{i,t}(k)) - \mathcal{P}_{i,t}(k) \left[\bar{\mathbb{V}}_{i,t}(s(k)) - \sum_{\ell} \mathcal{P}_{i,t}(\ell) \bar{\mathbb{V}}_{i,t}(s(\ell)) \right] \right) & k = k_{i,t}^*, \\ \frac{\gamma}{K} \left(-\pi_{i,t}(s(\ell)) \mathcal{P}_{i,t}(k) - \mathcal{P}_{i,t}(k) \left[\bar{\mathbb{V}}_{i,t}(s(k)) - \sum_{\ell} \mathcal{P}_{i,t}(\ell) \bar{\mathbb{V}}_{i,t}(s(\ell)) \right] \right) & k \neq k_{i,t}^* = \ell. \end{cases} \quad (41)$$

First, condition (C2) is trivially satisfied because $\mathcal{P}_t \in [0, 1]^{N \times K}$ for all $t \in \mathbb{N}$. Second, to verify condition (C1), it suffices to verify that both $O(\gamma^2)$ and $\gamma \tilde{U}_t$ converge to zero as $\gamma \rightarrow 0$. The entries of $\gamma \tilde{U}_{ik}(n)$ converge to zero as $\gamma \rightarrow 0$ for all realisations of $\pi_{i,t}(s)$

Next, we verify that the higher-order terms converge to zero. Consider $k = k_{i,t}^*$, then the higher-order terms from the geometric expansion are

$$O(\gamma^2) = -\frac{\gamma}{K} (A_{k,t}^\gamma - 1) + 2 \frac{\gamma}{K} \mathcal{P}_{i,t}(k) (A_{k,t}^\gamma - 1) - \left(\frac{\gamma}{K}\right)^2 (A_{k,t}^\gamma - 1) - \gamma \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right) (A_{k,t}^\gamma - 1) \\ + \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right) (A_{k,t}^\gamma - 1) \left(1 - \gamma - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right)\right) \sum_{m=1}^{\infty} \left(\gamma - \left(\mathcal{P}_{i,t}(k) - \frac{\gamma}{K}\right) (A_{k,t}^\gamma - 1)\right)^m,$$

where $A_{k,t}^\gamma = \exp\left(\frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right)$ and the higher-order terms from the power series expansion are

$$O(\gamma^2) = \mathcal{P}_{i,t}(k) \left(\sum_{m=2}^{\infty} \frac{1}{m!} \left(\frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right)^m\right) - \mathcal{P}_{i,t}^2(k) \left(\sum_{m=2}^{\infty} \frac{1}{m!} \left(\frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right)^m\right).$$

For the case when $\gamma \ll \mathcal{P}_i(k)$, the infinite sum from the geometric expansion is finite and therefore zero when multiplied by $A_{k,t}^\gamma - 1$ because $A_{k,t}^\gamma - 1$ converges to zero as $\gamma \rightarrow 0$. The higher-order terms from the power series expansion can be rewritten as

$$O(\gamma^2) = (\mathcal{P}_{i,t}(k) - \mathcal{P}_{i,t}^2(k)) \left(\sum_{m=2}^{\infty} \frac{1}{m!} \left(\frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right)^m\right) \\ = (\mathcal{P}_{i,t}(k) - \mathcal{P}_{i,t}^2(k)) \left(A_{k,t}^\gamma - 1 - \frac{\gamma \pi_{i,t}(s(k))}{\mathcal{P}_{i,t}(k) K}\right),$$

which converges to zero as $\gamma \rightarrow 0$. Therefore, the higher-order terms from both the geometric and power series expansion converge to zero as $\gamma \rightarrow 0$.

When $\mathcal{P}_i(k)$ is near zero, the minimum value that can be attained by $\mathcal{P}_i(k)$ is γ/K . If $\mathcal{P}_i(k) = \gamma/K$, we have $A_{k,t}^\gamma = e^{\pi_{i,t}(s(k))}$ and $\mathcal{P}_i(k) - \gamma/K = 0$. Therefore, the infinite sum in the geometric expansion is zero. Finally, when $\mathcal{P}_i(k) = \gamma/K$, the infinite sums from the power series expansion are finite and are therefore equal to zero as $\gamma \rightarrow 0$. We follow the same logic for $k \neq k_{i,t}^* = \ell$.

Therefore, every entry in γU_t converges to zero as $\gamma \rightarrow 0$ and this verifies condition (C1). ■

Sketch of proof of Proposition 1 The result holds by Proposition 1 in [Cartea, Chang, and Penalva \(2022\)](#), because for two LPs with the same finite set of candidate spreads, the game is a symmetric bimatrix game. A symmetric bimatrix game is a partnership game. This gives us two useful results for the replicator dynamics. First, by Theorem 6.1 in [Hofbauer \(2011\)](#), we have that for every partnership game, every interior trajectory of the replicator dynamics converges to a Nash equilibrium. Second, mixed equilibria of partnership games are unstable under the replicator dynamics, which means the replicator dynamics only converge to a pure strategy Nash equilibrium; see Section 6 in [Hofbauer and Hopkins \(2005\)](#). Finally, because partnership games have a strict Lyapunov function, we conclude from Corollary 6.6 in [Benaïm \(1999\)](#) that the continuous-time affine interpolated processes of policies from the EXP3 algorithm converge to the set of equilibria from the replicator dynamics. ■

SALES AND TRADING DEPARTMENT – DISCLAIMER

This document is intended for discussion purposes only and does not create any legally binding obligations on the part of Deutsche Bank AG and/or its affiliates ("DB"). Without limitation, this document does not constitute an offer, an invitation to offer or a recommendation to enter into any transaction. DB is not acting as your legal, financial, tax or accounting adviser or in any other fiduciary capacity with respect to any proposed transaction mentioned herein. This document does not constitute the provision of investment advice and is not intended to do so, but is intended to be general information. Any product(s) or proposed transaction(s) mentioned herein may not be appropriate for all investors and before entering into any transaction you should take steps to ensure that you fully understand the transaction and have made an independent assessment of the appropriateness of the transaction in the light of your own objectives, needs and circumstances, including the possible risks and benefits of entering into such transaction. You should also consider seeking advice from your own advisers in making any assessment on the basis of this document. If you decide to enter into a transaction with DB, you do so in reliance on your own judgment. The information contained in this document is based on material we believe to be reliable; however, we do not represent that it is accurate, current, complete, or error free. Assumptions, estimates and opinions contained in this document constitute our judgment as of the date of the document and are subject to change without notice.

This material was prepared by a Sales or Trading function within DB, and was not produced, reviewed or edited by the Research Department (which is independent from the Sales or Trading function). Any opinions expressed herein may differ from the opinions expressed by other DB departments including the Research Department. Sales and Trading functions are subject to additional potential conflicts of interest which the Research Department does not face. DB may engage in transactions in a manner inconsistent with the views discussed herein. In general, Sales and Trading personnel are compensated in part based on the volume of transactions effected by them. DB seeks to transact business on an arm's length basis with sophisticated investors capable of independently evaluating the merits and risks of each transaction, with investors who make their own decision regarding those transactions.

The distribution of this document and availability of these products and services in certain jurisdictions may be restricted by law. You may not distribute this document, in whole or in part, without our express written permission. DB SPECIFICALLY DISCLAIMS ALL LIABILITY FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR OTHER LOSSES OR DAMAGES INCLUDING, WITHOUT LIMITATION, LOSS OF PROFITS INCURRED BY YOU OR ANY THIRD PARTY THAT MAY ARISE FROM, OR IN CONNECTION WITH, ANY RELIANCE ON THIS DOCUMENT OR FOR THE RELIABILITY, ACCURACY, COMPLETENESS OR TIMELINESS THEREOF. DB is authorized under German Banking Law (competent authority: the European Central Bank and the BaFin - Federal Financial Supervising Authority) and DB in the UK is authorised by the Prudential Regulation Authority and is subject to limited regulation by the Prudential Regulation Authority and Financial Conduct Authority. Details about the extent of our authorisation and regulation are available on request or from <https://www.db.com/disclosures>.

References

- Abada, I., 2022, "Artificial Intelligence: Can Seemingly Collusive Outcomes Be Avoided?," available at <https://ssrn.com/abstract=3559308>.
- Aouad, A., and A. V. den Boer, 2021, "Algorithmic Collusion in Assortment Games," available at <https://ssrn.com/abstract=3930364>.
- Asker, J., C. Fershtman, and A. Pakes, 2021, "Artificial intelligence and pricing: The impact of algorithm design," NBER working paper, available at <https://www.nber.org/papers/w28535>.
- Audibert, J.-Y., R. Munos, and C. Szepesvári, 2007, "Tuning bandit algorithms in stochastic environments," in *International conference on algorithmic learning theory*, pp. 150–165. Springer.
- Auer, P., N. Cesa-Bianchi, and P. Fischer, 2002, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, 47(2), 235–256.
- Auer, P., N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, 2002, "The nonstochastic multiarmed bandit problem," *SIAM journal on computing*, 32(1), 48–77.
- Barzykin, A., P. Bergault, and O. Guéant, 2021, "Market making by an FX dealer: tiers, pricing ladders and hedging rates for optimal risk control," available at <https://arxiv.org/abs/2112.02269>.
- Benaïm, M., 1999, "Dynamics of stochastic approximation algorithms," in *Séminaire de Probabilités XXXIII*, ed. by J. Azéma, M. Émery, M. Ledoux, and M. Yor, pp. 1–68, Berlin, Heidelberg. Springer Berlin Heidelberg.
- BIS, 2019a, "Triennial Central Bank Survey – Foreign exchange turnover in April 2019," Monetary and Economic Department, Bank for International Settlements, available at https://www.bis.org/statistics/rpfx19_fx.pdf.
- , 2019b, "Triennial Central Bank Survey – OTC interest rate derivatives turnover in April 2019," Monetary and Economic Department, Bank for International Settlements, available at https://www.bis.org/statistics/rpfx19_ir.pdf.
- Börger, T., and R. Sarin, 1997, "Learning Through Reinforcement and Replicator Dynamics," *Journal of Economic Theory*, 77(1), 1–14.
- Brown, Z. Y., and A. MacKay, 2021, "Competition in pricing algorithms," Harvard Business School, working paper 20-067.
- Butz, M., and R. Oomen, 2019, "Internalisation by electronic FX spot dealers," *Quantitative Finance*, 19(1), 35–56.
- Calvano, E., G. Calzolari, V. Denicolò, and S. Pastorello, 2020, "Artificial intelligence, algorithmic pricing, and collusion," *American Economic Review*, 110(10), 3267–97.
- Cartea, A., P. Chang, and J. Penalva, 2022, "Algorithmic Collusion in Electronic Markets: The Impact of Tick Size," working paper, available at <https://ssrn.com/abstract=4105954>.
- Cartea, A., S. Jaimungal, and J. Walton, 2018, "Foreign exchange markets with last look," *Mathematics and Financial Economics*, 13(1), 1–30.
- Competition & Markets Authority, 2018, "Pricing Algorithms," available at <https://www.gov.uk/government/publications/pricing-algorithms-research-collusion-and-personalised-pricing>.
- , 2021, "Algorithms: How they can reduce competition and harm consumers," available at <https://www.gov.uk/government/publications/algorithms-how-they-can-reduce-competition-and-harm-consumers>.

- Dafoe, A., E. Hughes, Y. Bachrach, T. Collins, K. R. McKee, J. Z. Leibo, K. Larson, and T. Graepel, 2020, “Open problems in cooperative AI,” DeepMind working paper, available at <https://www.deepmind.com/publications/open-problems-in-cooperative-ai>.
- Dorner, F. E., 2021, “Algorithmic collusion: A critical review,” available at <https://arxiv.org/abs/2110.04740>.
- European Commission, 2017, “Algorithms and Competition,” Speech by Commissioner Margrethe Vestager at Bundeskartellamt 18th Conference on Competition, Berlin.
- Ezrachi, A., and M. E. Stucke, 2017, “Artificial intelligence & Collusion: When Computers Inhibit Competition,” *University of Illinois Law Review*, pp. 1775 – 1809.
- Fan, J., and I. Gijbels, 1992, “Variable Bandwidth and local linear regression smoothers,” *Annals of statistics*, 20(4), 2008 – 2036.
- Hansen, K. T., K. Misra, and M. M. Pai, 2021, “Frontiers: Algorithmic collusion: Supra-competitive prices via independent algorithms,” *Marketing Science*, 40(1), 1–12.
- Harrington, J. E., 2018, “Developing Competition Law for Collusion by Autonomous Artificial Agents,” *Journal of Competition Law & Economics*, 14(3), 331 – 363.
- Hofbauer, J., 2011, “Deterministic evolutionary game dynamics,” *Proceedings of Symposia in Applied Mathematics Volume*, 69.
- Hofbauer, J., and E. Hopkins, 2005, “Learning in perturbed asymmetric games,” *Games Econ. Behav.*, 52, 133–152.
- Ivaldi, M., B. Jullien, P. Rey, P. Seabright, and J. Tirole, 2003, “The Economics of Tacit Collusion,” Final Report for DG Competition, European Commission, available at https://ec.europa.eu/competition-policy/system/files/2021-04/the_economics_of_tacit_collusion_2003.pdf.
- Klein, T., 2021, “Autonomous algorithmic collusion: Q-learning under sequential pricing,” *RAND Journal of Economics*, 52(3), 538–599.
- Lattimore, T., and C. Szepesvári, 2020, *Bandit Algorithms*. Cambridge University Press, available at <https://tor-lattimore.com/downloads/book/book.pdf>.
- Löfström, T., H. Ralsmark, and U. Johansson, 2021, “Collusion in Algorithmic Pricing,” Swedish Competition Authority, Uppdragsforskningsrapport 2021:3, available at https://www.konkurrensverket.se/globalassets/dokument/informationmaterial/rapporter-och-broschyrer/uppdagsforskning/forsk-rapport_2021-3.pdf.
- OECD, 2017, “Algorithms and Collusion: Competition Policy in the Digital Age,” available at <https://www.oecd.org/competition/algorithms-collusion-competition-policy-in-the-digital-age.htm>.
- Oomen, R., 2017a, “Execution in an aggregator,” *Quantitative Finance*, 17(3), 383–404.
- , 2017b, “Last look,” *Quantitative Finance*, 17(7), 1057–1070.
- Schwalbe, U., 2018, “Algorithms, Machine Learning, and Collusion,” *Journal of Competition Law & Economics*, 14(4), 568 – 607.
- SIFMA, 2021, “2021 Capital Markets Fact Book,” available at <https://www.sifma.org/wp-content/uploads/2021/07/CM-Fact-Book-2021-SIFMA.pdf>.
- Siikanen, M., U. Nögel, and J. Kanninen, 2019, “Trading too expensively in the FX market?,” *Quantitative Finance*, 19 (12), 1933–1944.

Sutton, R. S., and A. G. Barto, 2018, *Reinforcement Learning: An Introduction*. The MIT Press, 2nd edn.

The White House, 2015, “Big data and differential pricing,” available at https://obamawhitehouse.archives.gov/sites/default/files/whitehouse_files/docs/Big_Data_Report_Nonembargo_v2.pdf.

Xiong, W., and R. Cont, 2021, “Interactions of Market Making Algorithms: a Study on Perceived Collusion,” conference proceedings ICAIF’21, USA, available at <https://dl.acm.org/doi/pdf/10.1145/3490354.3494397>.

Zhang, K., Z. Yang, and T. Başar, 2021, “Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms,” in Handbook on RL and Control (Springer Studies in Systems, Decision and Control), available at <https://arxiv.org/abs/1911.10635>.