Decentralised Finance and Automated Market Making

Marcello Monga
St Hugh's College
University of Oxford

Thesis submitted for the degree of
DPhil in Mathematics

Trinity 2024

Abstract

Automated market makers (AMMs) are a new type of trading venues which are revolutionising the way market participants interact. At present, the majority of AMMs are constant function market makers (CFMMs) where a deterministic trading function determines how markets are cleared. Within CFMMs, we focus on constant product market makers (CPMMs) which implements the concentrated liquidity (CL) feature. In this thesis we formalise and study the trading mechanism of CPMMs with CL, and we develop liquidity provision and liquidity taking strategies. Our models are motivated and tested with market data.

We derive optimal strategies for liquidity takers (LTs) who trade orders of large size and execute statistical arbitrages. First, we consider an LT who trades in a CPMM with CL and uses the dynamics of prices in competing venues as market signals. We use Uniswap v3 data to study price, liquidity, and trading cost dynamics, and to motivate the model. Next, we consider an LT who trades a basket of crypto-currencies whose constituents co-move. We use market data to study lead-lag effects, spillover effects, and causality between trading venues.

We derive optimal strategies for strategic liquidity providers (LPs) who provide liquidity in CPMM with CL. First, we use stochastic control tools to derive a self-financing and closed-form optimal liquidity provision strategy where the width of the LP's liquidity range is determined by the profitability of the pool, the dynamics of the LP's position, and concentration risk. Next, we use a model-free approach to solve the problem of an LP who provides liquidity in multiple CPMMs with CL. We do not specify a model for the stochastic processes observed by LPs, and use a long short-term memory (LSTM) neural network to approximate the optimal liquidity provision strategy.

Acknowledgements

First and foremost, I would like to thank my supervisor, Álvaro Cartea, who guided me through this journey. The endless hours spent discussing research had significant impact on my professional and personal development. He truly was a unique supervisor; he dedicated plenty of his precious time to discuss our projects, he truly cared about my research progress and he understood how the ups and downs of my personal life were impacting my studies. I will be forever grateful for his supervision.

I would like to thank Fayçal Drissi, who worked closely with me and whose dedication and talent were crucial to the results achieved by our research group. I would also like to thank Phillippe Bergault and Leandro Sánchez-Betancourt for the fruitful discussions during my PhD and my viva. I am grateful to Mihai Cucuringu, Olivier Guéant, Sebastian Jaimungal, and José Penalva, whose papers and books I have avidly read multiple times over the years. I would like to thank the director of the CDT in Random Systems, Rama Cont, and Anthony Ledford from Man AHL, who helped me step out of my comfort zone and provided immensely valuable advice.

I would like to thank Simon Davenport, Renaud Drappier, Gregoire Loeper, and Boris Oumow from BNP Paribas for their insights from the industry and for helping me bridge my research with real-world applications throughout my PhD. Moreover, during my internship at BNP Paribas, I had the privilege of working alongside these exceptional practitioners, who guided and mentored me.

I am grateful to all the people I have met during these years at the Mathematical Institute and the Oxford-Man Institute (OMI). In particular, I would like to thank Jen Desmond and Vanessa Wilkins for their wonderful work at the OMI; Álvaro Arroyo Nuñez and Patrick Chang, who were the first students I met at the Institute; Gabriel Garcia Arenas and Jonathan Plenk for the fruitful discussions we had over various cups of tea; Andrea Clini and László Mikolas for their advice; Harrison Waldon and Fernando Moreno Pino for being role models as postdocs at the OMI; Milena Vuletic for her energy; and all the participants of the Oxford Victoria seminar, including Sergio Calvo Ordoñez, Gerardo Durán-Martín, Qi Jin, Donggeun Kim, Yannick Limmer, Nicolas Petit, Daniel Poh, and Danni Shi.

I will never stop saying how grateful I am to my family for their endless support. Finally, I would like to thank all my friends in Italy and Switzerland who have always been there for me during these years. In particular, I would like to thank my friends from my hometown and my friends in Switzerland from the bQm analytics club.

Contents

1 Introduction **1**

1.1 Motivation . . . 1

1.2 Contribution . . . 1

1.2.1 Optimal execution . . . 2

1.2.2 Optimal liquidity provision . . . 2

2 Background **4**

2.1 Optimal execution . . . 4

2.2 Optimal liquidity provision . . . 6

2.3 Automated market makers . . . 7

3 Execution and Speculation **9**

3.1 Introduction . . . 9

3.2 Automated Market Making . . . 11

3.2.1 Description . . . 11

3.2.2 Data analysis . . . 13

3.2.2.1 Data description . . . 13

3.2.2.2 Rates and liquidity dynamics . . . 15

3.2.2.3 Convexity and execution costs . . . 18

3.3 Model I: optimal trading with an oracle rate . . . 21

3.3.1 Optimal trading strategy: execution and statistical arbitrage . . . 22

3.3.2 Constant impact parameter strategy . . . 23

3.3.3 The closed-form approximation strategy . . . 27

3.3.4 Comparison with the numerical approximation strategy . . . 28

3.4 Model II: optimal execution without oracle rate . . . 30

3.4.1 Performance criterion and value function . . . 30

3.4.2 The dynamic programming equation . . . 30

3.5 Performance of strategies . . . 32

3.5.1 Liquidation strategy . . . 33

3.5.2 Speculative strategy . . . 34

3.6 Conclusions . . . 36

4 Optimal Trading with Signals in Multiple Automated Market Makers **37**
4.1 Introduction . . . 37
4.2 Constant function market makers . . . 38
4.3 Data analysis . . . 40
4.3.1 Lead-lag relationships between trading venues . . . 40
4.3.1.1 Cointegration . . . 41
4.3.1.2 Causality . . . 41
4.3.1.3 Spillover effects . . . 42
4.3.2 Cointegration and oracle rates . . . 45
4.4 Optimal portfolio trading with oracles and signals . . . 46
4.4.1 Performance criterion and value function . . . 47
4.4.2 Optimal strategy with deterministic execution costs . . . 49
4.4.3 Closed-form approximation strategy . . . 50
4.5 Performance analysis . . . 52
4.6 Conclusions . . . 54

5 Optimal Liquidity Provision in constant product market makers with concentrated liquidity **55**
5.1 Introduction . . . 55
5.2 Constant function market makers and concentrated liquidity . . . 57
5.2.1 Constant function market makers . . . 57
5.2.2 Concentrated liquidity markets . . . 58
5.3 The wealth of liquidity providers in CL pools . . . 59
5.3.1 Position value . . . 59
5.3.2 Fee income . . . 61
5.3.2.1 Fee income: pool fee rate . . . 61
5.3.2.2 Fee income: spread and concentration risk . . . 61
5.3.2.3 Fee income: drift and asymmetry . . . 63
5.3.3 Rebalancing costs and gas fees . . . 66
5.4 Optimal liquidity provision in CL pools . . . 66
5.4.1 The problem . . . 66
5.4.2 The optimal strategy . . . 67
5.4.3 Discussion: profitability, PL, and concentration risk . . . 69
5.4.4 Discussion: drift and position skew . . . 71
5.5 Performance of strategy . . . 72
5.5.1 Methodology . . . 72
5.5.2 Benchmark . . . 73
5.5.3 Performance results . . . 74
5.6 Discussion: modelling assumptions . . . 75
5.6.1 Discussion: related work . . . 77

5.7 Conclusions . . . 77

6 Deep Liquidity Provision **78**

6.1 Introduction . . . 78

6.2 Automated Market Making . . . 80

6.2.1 Constant product market makers . . . 80

6.2.2 Concentrated liquidity . . . 81

6.3 Optimal liquidity provision . . . 82

6.3.1 Stochastic control problem . . . 83

6.3.2 Liquidity provision . . . 84

6.3.3 Performance criterion . . . 85

6.4 Model-free optimisation . . . 86

6.4.1 Implementation . . . 86

6.5 LSTM strategy with simulated data . . . 87

6.5.1 Model . . . 87

6.5.2 Single pool . . . 88

6.5.2.1 Trading frequency . . . 88

6.5.2.2 Fee tier . . . 90

6.5.2.3 Trend . . . 92

6.5.3 Two pools . . . 93

6.5.3.1 Trading frequency . . . 94

6.5.3.2 Fee tier . . . 95

6.5.3.3 Trend . . . 96

6.6 Numerical experiments with market data . . . 97

6.6.1 Data . . . 97

6.6.1.1 Uniswap v3 . . . 97

6.6.1.2 Binance . . . 98

6.6.2 Features . . . 99

6.6.3 Results . . . 100

A Appendix for Chapter 3 **103**

A.1 Proof of Theorem 1 . . . 103

A.2 Example for the liquidation strategy . . . 104

A.3 Model I with stochastic convexity cost . . . 105

B Appendix to Chapter 5 **111**

B.1 Proof of Proposition 5 . . . 111

B.2 Proof of Theorem 4 . . . 113

C Appendix for Chapter 6 **114**

C.1 Moving average convergence/divergence . . . 114

Bibliography **114**

Chapter 1

Introduction

1.1 Motivation

Decentralised Finance (DeFi) is a collective term for blockchain-based financial services that do not rely on intermediaries such as central authorities or banks. Over the past few years, DeFi has gained massive traction and the total value locked in DeFi platforms is as of July 2024 around 91.66 billion US dollars.[1] New powerful technologies are the engine behind the remarkable growth of DeFi, which is rapidly changing the financial landscape and is in direct competition with many traditional stakeholders. Within DeFi, automated market makers (AMMs) are a new paradigm in the design of trading venues and are revolutionising the way market participants provide and take liquidity. As of July 2024, the daily traded volume in AMMs is around 6.44 billions US dollars,[2] which accounts for more than 8.0% of the total spot cryptocurrency trading volume.[3] Currently, AMMs are mainly known as exchanges for cryptocurrencies; however, the core concepts of AMMs go beyond the cryptocurrency sector and they are poised to challenge traditional electronic exchanges in all asset classes.

Despite the remarkable growth of AMMs, there is little understanding of their microstructure properties. AMMs are a technology with interesting features and high potential. However, more research is needed to understand, regulate, and further develop AMMs. Indeed, little is known of what happens at a microstructural level in AMMs and it is important to provide academics and practitioners the right tools and to study in detail the advantages and disadvantages of this new type of trading venue.

1.2 Contribution

In this thesis we formalise and study the trading mechanism of AMMs, and we develop liquidity provision and liquidity taking strategies. Our models are motivated and tested with market data. For the sake of chapters being self-contained, the necessary terminology is reintroduced in each chapter.

[1] https://www.stelareum.io/en/defi-tvl.html
[2] https://www.coingecko.com/en/exchanges/decentralized
[3] https://www.theblock.co/data/decentralized-finance/dex-non-custodial

1.2.1 Optimal execution

A classical problem in the market microstructure literature is how an agent can execute a large order while minimising intrinsic and extrinsic execution costs. This problem is extensively studied in the context of limit order book (LOB) markets. In the extant literature, there is a large number of execution models that are specifically designed for traditional LOBs; however, there are no rigorous models to trade and execute assets in AMMs.

Chapter 3 presents the work in Cartea et al. (2022b), which solves the problem of a liquidity taker (LT) who wishes to trade in a particular class of AMMs. The LT wishes to execute a large position in an asset and to execute statistical arbitrages based on market signals. The LT trades in the AMM and uses the exchange rate from a more liquid venue to anticipate future price movements.

We formulate the trading problem as a stochastic control problem where the LT controls the speed at which she sends liquidity taking orders. Key to the performance of the LT's strategies, is to balance price risk and execution costs. To showcase the performance of our strategies, we use in-sample market data to estimate model parameters and out-of-sample market data to execute the strategies in 'real time' as an LT would have done. Furthermore, we use market data to derive important properties about execution costs and market impact in AMMs.

Chapter 4 presents the work in Cartea et al. (2023a), which solves the problem of an LT who executes large orders and executes statistical arbitrages in a basket of crypto-currencies whose constituents co-move. We use statistical tools to study lead-lag effects, spillover effects, and causality between trading venues and we study co-movements between different crypto-currencies.

We use stochastic control tools and derive a closed-form strategy that can be computed and implemented by the LT in real time. The LT uses market signals and exchange rate information from relevant AMMs and traditional venues to enhance the performance of her strategy. Finally, we use market data from two pools of Uniswap v3 and from the LOB-based exchange Binance to showcase the performance of the strategy.

1.2.2 Optimal liquidity provision

In traditional trading venues, liquidity providers quote buy and sell prices and profit from the spread between the two. This problem is extensively studied in the context of LOB markets and quote-driven markets; however, little research has been conducted on strategic liquidity provision in AMMs.

AMMs introduce a new liquidity provision mechanism where rules to clear demand and supply of liquidity are considerably different. The most popular type of AMMs are constant product market makers (CPMMs) with concentrated liquidity (CL). In CPMMs with CL, LPs specify the rate intervals (i.e., tick ranges) over which they deposit their assets, and this liquidity is counterparty to trades of LTs when the marginal exchange rate of the pool is within the liquidity range of the LPs. When LPs deposit liquidity, fees paid by LTs accrue and are paid to LPs when they withdraw their assets from the pool. The amount of fees accrued to LPs is proportional to the share of liquidity they hold in each liquidity range of the pool.

Chapter 5 presents the work in Cartea et al. (2022c), which solves the problem of an LP who provides liquidity in a single pool with CL. We use stochastic control to derive a self-financing and closed-form optimal liquidity provision strategy, which the LP uses to dynamically adjust the range where she provides liquidity.

The width of the LP's liquidity range is determined by the profitability of the pool (provision fees minus gas fees), the dynamics of the LP's position, and concentration risk. Concentration risk refers to the decrease in fee revenue if the marginal exchange rate (akin to the midprice in an LOB) in the pool exits the LP's range of liquidity. When the drift in the marginal rate is stochastic, we show how to optimally skew the range of liquidity to increase fee revenue and profit from the expected changes in the marginal rate. Finally, we use Uniswap v3 data to show that, on average, LPs have traded at a significant loss, and to show that the out-of-sample performance of our strategy is superior to the historical performance of LPs in the pool we consider.

Chapter 6 uses a model-free approach to solve the problem of an LP who provides liquidity in multiple liquidity pools with CL. We do not specify a model for the stochastic processes observed by the LP, and we characterise the discrete-time dynamics of the LP's wealth, which results in fee income and the value of her holdings in the pool.

At each time step, the LP specifies the proportion of wealth to deposit in each pool and the ranges where she want to provide liquidity. The LP evaluates the performance of her strategy with a performance criterion and uses LSTM neural networks to approximate the optimal strategy. We train the LSTM network on simulated data and on market data.

We test the model-free strategy on simulated data to showcase the performance of the strategy under different market conditions. First, we consider an LP who provides liquidity in a single liquidity pool with CL and we recover established results from Chapter 5. Then, we consider an LP who provides liquidity in two liquidity pools with CL.

Finally, we train the LSTM network on market data from Uniswap v3 and Binance. The input to the LSTM strategy is a set of features which we extract from the data in both venues. Overall, we find that gas fees, a flat fee paid by blockchain users, have a great impact on the performance of the strategy.

Chapter 2

Background

2.1 Optimal execution

Whenever an LT has to liquidate or to acquire a large position, she may incur adversarial price movements, as a consequence of her own trades. Indeed, if she exits or enters her position using a single order, she might consume all the available liquidity at the best price and receive worse prices to complete the order. Typically, to reduce execution costs, LTs split the parent order in smaller child orders which are executed over a longer time window. However, if the LT executes the child orders too slowly, then she is exposed to market price risk and she might face a loss.

The literature on optimal execution in traditional electronic markets is vast. It is built on the seminal works Bertsimas and Lo (1998) and Almgren and Chriss (2000). In their work, the authors study the problem of an LT who uses market orders (MOs) to liquidate a large position in a limit order book (LOB). More specifically, the Almgren–Chriss model is a discrete-time model where the LT maximises a mean-variance objective function. Later on, many variations to the model framework were proposed. Forsyth et al. (2012) use a mean-quadratic variation approach, which can be approximated with a mean-variance type model for short trading windows. Schied and Schoneborny (2009) consider an infinite time-horizon problem for a Von Neumann–Morgenstern LT and characterises the optimal strategy using stochastic control tools. Cartea et al. (2015) introduce various frameworks where the LT use an objective function equal to the LT's terminal wealth minus an inventory penalty to solve the optimal execution problem in the context of LOB markets. Cartea and Jaimungal (2014) extend this framework to allow the LT to both post LOs and send MOs.

Regarding model parameters, Almgren (2012) assumes that volatility and temporary price impact are random and compute the optimal strategy in this framework. Cheridito and Sepin (2014) also study the case of random volatility and random temporary price impact but compares three different optimal strategies obtained using three different performance criteria. Gatheral and Schied (2013) review dynamical market impact models and studies the regularity of their associated optimal strategies. Graewe et al. (2018) introduce a price-dependent market impact and considers a liquidation problem where the LT can unwind her position both in the lit market and in a dark pool. Fouque et al. (2022) introduce fast-mean reverting stochastic price impact to reproduce exactly how price impact moves during the day.

Numerous works in the optimal execution literature explore different models for price impact and in LOBs. Lorenz and Schied (2013) use linear transient price impact with exponential decay and employs singular stochastic control to deal with the non-Markovian framework. Bank and Voß (2019) introduces a price impact model which accounts for finite market depth, tightness, and resilience. Obizhaeva and Wang (2013) introduce a dynamic model for LOBs and shows the importance of supply/demand dynamics in determining optimal trading strategies. Later on, the Obizhaeva-Wang model was extended and in Alfonsi et al. (2010) the authors consider an optimal liquidation problem where the LOB has a general shape. Alfonsi and Schied (2010) derive explicit optimal execution strategies in a discrete-time LOB model with general shape functions and an exponentially decaying price impact. Gatheral et al. (2010) study the problem of an LT subject to transient price impact who wishes to liquidate a large position while minimising execution costs. Hey et al. (2023) quantify the cost that LTs face when they misspecify price impact. Muhle-Karbe et al. (2024) use a stochastic liquidity parameter to approximate nonlinear price impact.

The Almgren-Chriss model was further extended to allow for the use of market signals to improve performance of strategies. Cartea and Jaimungal (2016) use order flow as a predictive market signal and derives closed-form solutions. Bechler and Ludkovski (2015) study an optimal execution framework where the LT accounts for order flow imbalance and endogenise the trading horizon to react to it. Cartea et al. (2018a) also employs order flow imbalance and uses real NASDAQ data to calibrate a Markov chain-modulated pure jump model of price, spread, limit orders (LOs) and MOs arrivals, and volume imbalance. In Lehalle and Neuman (2019) the authors consider a Von Neumann-Morgenstern LT who uses a Markovian signal to trade and Neuman and Voß (2020) extend this framework to include transient price impact. Forde et al. (2022) consider an LT subject to power-law resilience and zero temporary price impact who employs a Gaussian signal. Cartea et al. (2018b) and Bergault et al. (2022) consider a multi-asset framework where the LT trades in a basket of assets which co-move. Cartea et al. (2022a) use the signature of the path of a stochastic process to derive an optimal double-execution strategy.

Another strain of the optimal execution literature focuses on trying to reduce latency. Specifically, Cartea and Sánchez-Betancourt (2021) derive the price that LTs would be willing to pay to reduce their latency in the marketplace. Cartea et al. (2021) show how LTs can leverage marketable limit orders (MLOs) to reduce latency and Cartea and Sánchez-Betancourt (2022b) derive an optimal trading strategy which uses MLOs to liquidate a position over a trading window when there is latency in the marketplace.

Recent works in the literature use reinforcement learning (RL) to solve the optimal execution problem; see Guéant (2023). One of the first example of the use of RL in the algorithmic trading literature was the work of Nevmyvaka et al. (2006), where the authors use tabular Q learning to optimise trade execution. Hendricks and Wilcox (2014) use Q learning to outperform the model by Almgren and Chriss, and they test their strategy on market data. Ning et al. (2021) solve the optimal execution problem using a model free double deep Q learning algorithm where the input of the LT strategy are features of the LOB. Cartea et al. (2023e) use double deep Q networks to

derive the optimal strategies for an LT who executes statistical arbitrage in a foreign exchange (FX) triplet. Cartea et al. (2023c) use contextual bandits and Gaussian processes to exploit trading signals and improve the Almgren–Chriss strategy. Waldon et al. (2024) introduce a framework to trade in continuous time under uncertainty.

2.2 Optimal liquidity provision

In traditional trading venues, LPs quote buy and sell prices and profit from the spread between the two. In LOB markets, LPs send LOs that specify price and quantity they would like to buy or sell of a given financial instrument. An LO sent to the market is stored in the LOB until an LT matches it or the LO is cancelled. LTs send market orders (MOs) when they wish to buy or to sell the underlying product immediately, which are matched with LOs resting in the LOB. Most LOB markets employ a price-time-priority mechanism where incoming buy (sell) MOs are matched by sell LOs with lowest (highest) price amongst outstanding sell (buy) LOs. Moreover, if two or more outstanding sell LOs offer the same price, then the one with earliest timestamp is executed first.

In trading venues organised as over-the-counter (OTC) markets, trading happens directly between two parties. Typically, an LT interested in trading sends a formal request to various LPs specifying the size of her order. Then, LPs reply to the LT and propose bid and ask prices, and the LT decides whether to trade on the proposed quotes or not. In OTC markets, LPs propose quotes based on their own inventory, the size of the LT's order, and market conditions.

The literature on optimal liquidity provision in traditional trading venues is vast and it is built on the seminal work of Ho and Stoll (1981). In their work, the authors use dynamic programming to derive optimal bid and ask prices in feedback form that maximise the expected utility of the LP's final wealth. Later, their work was revived by Avellaneda and Stoikov (2008) who use econophysics results to model arrival rates of MOs to the LOB, and derive an optimal strategy to post LOs. Gueant et al. (2012) provide a detailed analysis of the model proposed by Avellaneda and Stoikov, and solve the problem under inventory constraints.

Later on, variations of the model by Avellaneda and Stoikov were proposed. Cartea et al. (2014) use self-exciting processes to model order flow, and solve the optimisation problem using an objective function equal to the LP's terminal wealth minus an inventory penalty. Cartea et al. (2017) consider the ambiguity in the specification of fill probabilities and the dynamics of intensities and prices. In the work by Cartea and Wang (2020), the LP incorporates alpha signals into her optimal posting strategy. Jusselin (2021) uses generalised Hawkes processes to model order flow and solves the problem of and LP who provides liquidity in an LOB market. Bergault and Guéant (2021) consider an LP who provides liquidity in multiple assets and uses a factor model to reduce the dimension of the optimisation problem. Moreover, the authors assumes that the the LP receives liquidity taking orders of various sizes, and show how the size of the order impacts the optimal trading strategy. Bergault et al. (2021) derive a closed-form approximation for the bid and ask quotes offered by an LP who provides liquidity in multiple assets. Barzykin et al. (2021, 2022, 2023) study liquidity provision in the FX market. Specifically, they consider an LP who provides liquidity in various

currencies and hedges inventory risk by trading with other LPs. Barzykin et al. (2024) studies an LP who provides liquidity in spot precious metals, where liquidity is mainly provided by futures contracts.

Inspired by the seminal works of Glosten and Milgrom (1985) and Kyle (1985), various works in the optimal liquidity provision literature consider LPs who trades with informed LTs. Campi and Zabaljauregui (2020) considers an LP who provides liquidity in a financial markets where arrival rates and price dynamics depend on unobservable factors. Herdegen et al. (2023) consider an informed LT who trades with multiple LPs who are balancing inventory risk and adverse selection. In the work from Cartea and Sánchez-Betancourt (2022a), the LP trades at a loss with an informed LT to extract trading signals. Their work was later extended by Bergault and Sánchez-Betancourt (2024) who frame the problem as a mean field game.

Recent works in the literature use RL to solve the optimal liquidity provision problem; see Gašperov et al. (2021). One of the first example of the use of RL for liquidity provision was the work of Chan and Shelton (2001), where the authors use various RL algorithms to provide liquidity in a model similar to the one proposed by Glosten and Milgrom. Other notable examples in the literature include the work of Spooner et al. (2018), where the authors use temporal difference RL to solve the optimal liquidity provision problem, and Spooner and Savani (2021) and Gašperov and Kostanjčar (2021) who use adversarial RL. Guéant and Manziuk (2019) use a model-based approach to solve the optimal liquidity provision problem, and Jerome et al. (2023) introduce a Python module to solve model-based optimal liquidity provision problems in an LOB market.

2.3 Automated market makers

AMMs are a new paradigm in the design of trading venues and are revolutionising the way market participants provide and take liquidity. Instead of relying on a matching mechanism, they rely on hard-coded and immutable programs running on peer-to-peer networks such as Ethereum (Buterin (2014)), Tezos (Goodman (2014)) and Solana (Yakovenko (2014)).

At present, the majority of AMMs are CFMMs. In CFMMs, a trading function and a set of rules determine how LTs and LPs interact, and how markets are cleared. The trading function is deterministic and known to all market participants. CFMMs display pools of liquidity for different assets, where the relative prices assets is determined by their quantities in the pool as prescribed by the trading function. The trading function establishes the link between liquidity and prices, so liquidity takers can compute the execution costs of their trades as a function of the trade size. A key difference between CFMMs and a LOB is that execution costs in CFMMs are given by a closed-form formula, where the convexity of the trading function plays a key role. As in traditional markets that operate an LOB, the larger the size of an order, the higher are the execution costs.

The most widely used CFMMs are CPMMs, such as Uniswap v3 Adams et al. (2021), PancakeSwap,[1] SushiSwap,[2] Balancer Martinelli and Mushegian (2019), DODO,[3] and Bancor Hertzog

[1] https://docs.pancakeswap.finance/
[2] https://docs.sushi.com/
[3] https://dodoex.github.io/docs/docs/whitepaper/

et al. (2017). Another important class of CFMMs is that of constant sum market makers (CSMMs) such as mStable[4] and StableSwap Egorov (2019). We refer the interested reader to Xu et al. (2021) for an overview of the different types of AMMs.

AMMs are a relatively new concept and thus academic literature is still restricted. Early works on AMMs focus on their market microstructure properties and include the work of Angeris et al. (2021), where the authors formalise the mathematics of Uniswap v2 and show how to take advantage of the spread between the protocol and another reference market. This work is generalised to CFMMs in Angeris and Chitra (2020). Cartea et al. (2023b) study the losses faced by LPs in CFMMs and then introduce predictable loss, which measures the unhedgeable losses of LPs stemming from the depreciation of their holdings in the pool and from the opportunity costs from locking their assets in the pool. Predictable loss is similar to loss-versus-rebalancing in Milionis et al. (2022) which describes the unhedgeable losses of LPs in traditional CFMMs due to losses to arbitrageurs.[5] Capponi and Jia (2021) show that rates at which users provide liquidity are negatively correlated with volatility and positively correlated with traded volume. This feature is also observed in traditional LOB markets. Berg et al. (2022) empirically studies inefficiencies in AMMs with particular focus on Uniswap and SushiSwap. Milionis et al. (2023) study the adverse selection incurred by LPs in the presence of fees, and Fukasawa et al. (2023) study the hedging of the impermanent losses of LPs.

Another literature strain on strategic liquidity provision include Heimbach et al. (2022) which discusses the tradeoff between risks and returns that LPs face in Uniswap v3. Cartea et al. (2022c) introduces a continuous-time model for optimal liquidity provision, and Lı et al. (2023) study the economics of liquidity provision. The models in Fan et al. (2021) and Fan et al. (2022) focus on fee revenue and use approximation techniques to obtain dynamic strategies.

Finally, there is a growing literature on AMM design for fair competition between LPs and LTs. Goyal et al. (2023) study an AMM with dynamic trading functions that incorporate beliefs of LPs, Lommers et al. (2023) study AMMs where the LP's strategy adjusts dynamically to market information, Cartea et al. (2023d) generalise CFMMs and propose AMM designs where LPs express their beliefs and risk preferences, and Bergault et al. (2023) introduce an AMM design inspired by OTC markets.

Other works on AMMs focus on arbitrage opportunities arising from blockchain inefficiencies. Daian et al. (2020) formalises front-running in AMMs and introduces priority gas auctions, which consist in competitively bidding up transaction fees in order to obtain priority ordering. Park (2021) discusses front-running arbitrages in AMMs, and proposes a pricing rule to prevent them. Zhou et al. (2021) provides a detailed analysis of sandwich attacks and quantifies how much a LT performing sandwich attacks can earn on a daily basis.

[4] `https://docs.mstable.org/`

[5] Milionis et al. (2022) introduced loss-versus-rebalancing in August 2022 as a measure that quantifies the unhedgeable losses of LPs in CFMMs to arbitrageurs. Contemporaneously, Cartea et al. (2023b) introduced predictable loss in November 2022 as a component in the wealth of strategic LPs that quantifies predictable loss due to the convexity of the trading function and due to opportunity costs in CFMMs and in CL markets.

Chapter 3

Execution and Speculation

3.1 Introduction

Decentralised Finance (DeFi) is a collective term for blockchain-based financial services that do not rely on intermediaries such as brokers or banks. New powerful technologies are the engine behind the remarkable growth of DeFi, which is changing the financial landscape and is in direct competition with many traditional stakeholders. Within DeFi, Automated Market Makers (AMMs) are a new paradigm in the design of trading venues and are revolutionising the way market participants provide and take liquidity. Currently, AMMs are mainly known as exchanges for cryptocurrencies; however, the core concepts of AMMs go beyond the cryptocurrency sector and they are poised to challenge traditional electronic exchanges in all asset classes.

At present, the majority of AMMs are constant function market makers (CFMMs). In CFMMs, a trading function and a set of rules determine how liquidity takers (LTs) and liquidity providers (LPs) interact, and how markets are cleared. The trading function is deterministic and known to all market participants. CFMMs display pools of liquidity for pairs of assets, where the relative prices between the two assets is determined by their quantities in the pool as prescribed by the trading function. The trading function establishes the link between liquidity and prices, so LTs can compute the execution costs of their trades as a function of the trade size. A key difference between CFMMs and limit order books (LOBs) is that execution costs in CFMMs are given by a closed-form formula, where the convexity of the trading function plays a key role. As in traditional markets that operate an LOB, the larger the size of an order, the higher are the execution costs.

Within CFMMs, we focus on constant product market makers (CPMMs), which are the most popular type of CFMM and where the trading function uses the product of the quantities of each asset in the pool to determine clearing prices. In this chapter, we present the work of Cartea et al. (2022b), who solve the problem of an LT who wishes to trade in a CPMM to execute a large position in an asset and to execute statistical arbitrages based on market signals. We formulate the trading problem as a stochastic control problem where the LT controls the speed at which she sends liquidity taking orders. Key to the performance of the LT's strategies, is to balance price risk and execution costs. In CPMMs, the execution costs given by the trading function are inversely proportional to the depth of the pool and proportional to a non-linear transformation of the relative prices of the two assets in the pool.

Despite very high levels of activity for many of the pairs traded in CPMMs, price formation currently occurs in the LOBs of alternative electronic markets. In one version of our model, we assume that LTs in a CPMM inform their decisions with the prices in the CPMM and those from an alternative venue, and assume that the depth of the pool does not vary during the LT's trading horizon. In this setup, we derive a versatile trading strategy which can be used to focus on the execution of a large order or on statistical arbitrages.

When the focus is to exchange a large position in one asset for another asset, both of which are provided as a pair in the CPMM, the strategy relies on two components. One component is as in the traditional execution strategies (e.g., TWAP-like or Almgren-Chriss), and the second component is an arbitrage that takes advantage of short-lived discrepancies in the prices of the CPMM and those in the alternative venue. Instead, if the objective is speculation, the strategy relies more on the statistical arbitrage component to take advantage of the lead-follow relationship between the prices in the CPMM and the alternative venue.

In anticipation of the growth of AMMs, another version of our model assumes that prices in the CFMMs are efficient, so the discrepancies between the CFMM and LOB prices are not economically significant. The increase in the efficiency of prices in CFMMs will be a result of an increase in the activity of LPs and LTs, which will also result in more changes in the depth of the pool of the CPMM. Thus, we propose another model where the depth of the pool is stochastic and we solve the LT's execution problem for large orders.

We use Uniswap v3 data for CPMMs that trade pairs of cryptocurrencies to study the empirical properties of this particular AMM, and to illustrate the performance of the proposed liquidation and speculative strategies. The efficient prices are those from Binance where LPs and LTs interact through a traditional price-time priority LOB. To showcase the performance of our strategies, we use in-sample data to estimate model parameters and out-of-sample data to execute the strategies in 'real time' as an LT would have done. In our analysis, we use rolling time windows of a few hours starting 1 July 2021 and ending 5 May 2022 to obtain the distribution of the financial performance of the strategies. We look at two pairs of assets, one that is heavily traded and one that is not as frequently traded. We show the superior performance of our execution strategy over TWAP and over a strategy that would have executed the whole inventory in one trade at the beginning of the trading horizon. Finally, we show that there are profitable opportunities to execute statistical arbitrages in Uniswap v3 when the strategy is informed by Binance prices.

The remainder of this chapter is organised as follows. Section 3.2 discusses how CFMMs operate and uses Uniswap v3 data to study price, liquidity, and trading cost dynamics in CPMMs. Section 3.3 solves the optimal execution problem when the pool has constant depth during the execution window and price formation is in an alternative trading venue. Section 3.4 solves the optimal execution problem when the pool depth is stochastic and price formation takes place in the AMM. Finally, Section 3.5 showcases the performance of liquidation and statistical arbitrage strategies.

3.2 Automated Market Making

In this section, we discuss how CFMMs operate and how they differ from electronic markets where traders interact through an LOB. In particular, we describe the interactions of market participants with a CFMM that is in charge of a pair of assets. We use transaction data from Uniswap v3 to study the activity of market participants, the dynamics of liquidity, and implicit transaction costs.

3.2.1 Description

AMMs are hard-coded and immutable programs running on a peer-to-peer network. They provide a venue to trade pairs of assets X and Y, where the liquidity of the pool consists of x units of X and y units of Y. The exchange rate of the pool is the price of Y in terms of the price of X, and it is determined by the quantities x and y.[1] Two types of market participants interact in an AMM: LPs deposit their assets in the pool and LTs trade directly with the pool.

Here, we consider a CFMM that trades the pair of assets X and Y. CFMMs are characterised by a deterministic trading function $f(x, y)$ that determines the rules of engagement among participants in the pool. For instance, the trading function of the CPMM is $f(x, y) = x \times y$. Other types of CFMMs are the constant sum market maker with $f(x, y) = x + y$; the constant mean market maker with $f(x, y) = w_x\, x + w_y\, y$, where $w_x, w_y > 0$ and $w_x + w_y = 1$; and the hybrid function market maker, which uses combinations of trading functions.

In peer-to-peer networks, participants invoke the code of the AMM smart contract to instruct market operations. LPs send messages with instructions to deposit or withdraw liquidity, and LTs send messages to exchange one asset for the other. To provide liquidity, an LP instructs the AMM with the quantities in assets X and Y to be deposited in a specific pool. On the other hand, LTs indicate to the AMM the pool and the quantity of the asset to be exchanged. The available liquidity in the pool and the trading function of the AMM determine the exchange rate received by the LT. For each trade, LTs pay the AMM a transaction fee, which is distributed amongst LPs in the same proportion as their contributions to the pool.[2]

The trading function $f\,(x, y)$ is increasing in x and y, and it ties the state of the pool before and after an LT transaction is executed. Throughout, the signs of Δx and Δy are the same. If $\Delta y > 0$, the LT sells asset Y, and if $\Delta y < 0$, the LT buys asset Y; for simplicity, we assume zero fees.[3] The condition

$$f(x - \Delta x, y + \Delta y) = f(x, y) = \kappa^2$$

determines the quantity Δx that the agent receives (pays) when exchanging $\Delta y > 0$ ($\Delta y < 0$). The trading function keeps the quantity κ^2 constant before and after a trade is executed. We write

[1] Some AMMs also display pools with more than two assets.

[2] See Heimbach et al. (2021) and Cartea et al. (2022b) for an analysis on how LPs profit from their activity.

[3] To take into account the fee for an LT transaction, one applies a discount to the quantity Δy before calculations are carried out.

$f(x, y) = \kappa^2$ as $x = \varphi(y)$ for an appropriate function φ that depends on κ; we refer to φ as the *level function*, and assume it is convex.[4]

If an LT wishes to sell Δy units of asset Y, she receives $\Delta x = \Delta y \times \tilde{Z}(\Delta y)$ units of asset X in exchange. Here, $\tilde{Z}(\Delta y)$, with units X/Y, is the exchange rate received by the agent when trading a quantity Δy of asset Y. Therefore,

$$x - \Delta x = \varphi(y + \Delta y) \implies \varphi(y) - \Delta y\, \tilde{Z}(\Delta y) = \varphi(y + \Delta y)\,,$$

so

$$\tilde{Z}(\Delta y) = \frac{\varphi(y) - \varphi(y + \Delta y)}{\Delta y}\,, \tag{3.1}$$

and for an infinitesimal quantity Δy we write

$$Z = -\varphi'(y)\,. \tag{3.2}$$

We refer to Z as the *instantaneous rate* of the AMM, which is equivalent to the midprice in an LOB. The instantaneous rate Z is a reference exchange rate – the difference between its value and the execution rate is similar to the difference between the LOB midprice and the average price obtained by a liquidity taking order that crosses the spread and walks the book when it is filled.

The trading function $f(x, y)$ is increasing in the pool quantities x and y. Thus, when LP activity increases (decreases) the size of the pool, the value of κ increases (decreases). We refer to κ as the depth of the pool. A distinctive characteristic of AMMs is that liquidity provision changes the depth of the pool, but it does not change the instantaneous rate. For example, in a CPMM, the instantaneous rate is the ratio of the quantities supplied in the pool, i.e.,

$$Z = \frac{x}{y}\,,$$

and when an LP deposits quantities Δx and Δy in the pool, the pair $(\Delta x, \Delta y)$ must satisfy

$$\frac{x}{y} = \frac{x + \Delta x}{y + \Delta y} = Z\,, \tag{3.3}$$

and the value of κ changes from $\sqrt{x \times y}$ to $\sqrt{(x + \Delta x)\,(y + \Delta y)}$. For (3.3) to hold, there exists ρ such that $\Delta x = \rho\, x$ and $\Delta y = \rho\, y$, i.e., liquidity provision and removal by LPs in a CPMM is performed in fractions of the pool quantities x and y, and the depth changes from κ to $(1 + \rho)\, \kappa$.

The level function φ is decreasing, thus the rate in (3.1) received by an LT deteriorates as the size of the trade increases.[5] The formulas (3.1) and (3.2) encode all the information needed by an LT to interact with an AMM. For a trade of size Δy, the distance between the instantaneous rate Z and the execution rate $\tilde{Z}\,(\Delta y)$ is the *unitary execution cost* of the AMM, specifically

$$\text{Unitary execution cost } = \left| Z - \tilde{Z}(\Delta y) \right|\,. \tag{3.4}$$

[4]The convexity of the level function is by design. One can show that a no-arbitrage condition leads to the necessary convexity of the level function; see Cartea et al. (2022b).

[5]The trading function f is increasing in x and y and $\partial_y f(x, y) = \partial_y f\,(\varphi(y), y) = 0$ so $\varphi'(y) = -\frac{\partial_y f(x,y)}{\partial_x f(x,y)} < 0$.

The unitary execution cost is defined as the average execution cost of one unit of asset Y, in units of asset X.

Another distinctive characteristic of AMMs is that liquidity taking activity may change the instantaneous rate Z, but does not change the depth of the pool. Furthermore, AMMs and LOBs differ in a number of other aspects including accessibility and the way LPs are compensated. By design, AMMs are permissionless, so anyone can participate in the market.[6] LPs in AMMs are compensated in two ways. One, they are rewarded with the proceeds from fees that LTs pay for every trade. Two, the execution costs in (3.4) incurred by LTs due to the convexity of the level function stay in the pool. In some AMMs, the fee proceeds are put back in the pool, so the number of assets owned by LPs increases. In other AMMs, in particular those with a CL feature, fees are accumulated in a separate account and are earned by LPs when they withdraw their liquidity from the pool.

3.2.2 Data analysis

3.2.2.1 Data description

Uniswap v3 is considered the cornerstone of DeFi and is currently the most liquid AMM. It is a CPMM with trading function

$$f(x, y) = x \times y = \kappa^2, \tag{3.5}$$

so the level function is

$$\varphi(y) = \frac{\kappa^2}{y}\,.$$

When an LT trades Δy, the execution rate (3.1) is

$$\tilde{Z}(\Delta y) = \frac{1}{\Delta y}\left(\frac{\kappa^2}{y} - \frac{\kappa^2}{y + \Delta y}\right)\,,$$

and the instantaneous rate (3.2) is

$$Z = -\varphi'(y) = \frac{\kappa^2}{y^2}\,.$$

To further study the characteristics of CPMMs and motivate our framework, we use transaction data from Uniswap v3 and the traditional LOB-based exchange Binance. Specifically, we look at the two pairs ETH/USDC and ETH/DAI. The ticker ETH represents the cryptocurrency *Ether*, which is the native cryptocurrency of the Ethereum blockchain. The ticker USDC represents *USD coin*, a cryptocurrency fully backed by U.S. Dollars (USD); and DAI represents the cryptocurrency *Dai*, which tracks parity with the U.S. Dollar. For the pool ETH/USDC, the unit of depth κ is $\sqrt{\text{ETH} \cdot \text{USDC}}$, of x is USDC, of y is ETH, and the instantaneous rate, the execution rate, and the unitary execution cost are all in USDC/ETH; similarly for the pool ETH/DAI. For ease of reading, in the remainder of this work we omit the units of κ.

[6] This is true for AMMs running on permissionless peer-to-peer networks. If the same AMMs were to be implemented by a central authority for fiat currencies or stocks, then one expects stricter participation rules.

We analyse transaction data of the most liquid pools for the pairs ETH/USDC and ETH/DAI. Transaction information from decentralised exchanges is public. Between 5 May 2021 and 5 May 2022, there are 1,757,181 LT transactions and 42,403 LP transactions for the pair ETH/USDC, and 101,538 LT transactions and 12,142 LP transactions for the pair ETH/DAI. Figures 3.1 and 3.2 show the historical daily volumes of transactions in terms of USD value and in terms of the number of transactions for ETH/USDC and ETH/DAI, respectively. These pools are considered an alternative to LOB-based trading venues such as Binance, which is the most liquid and active venue for both pairs.

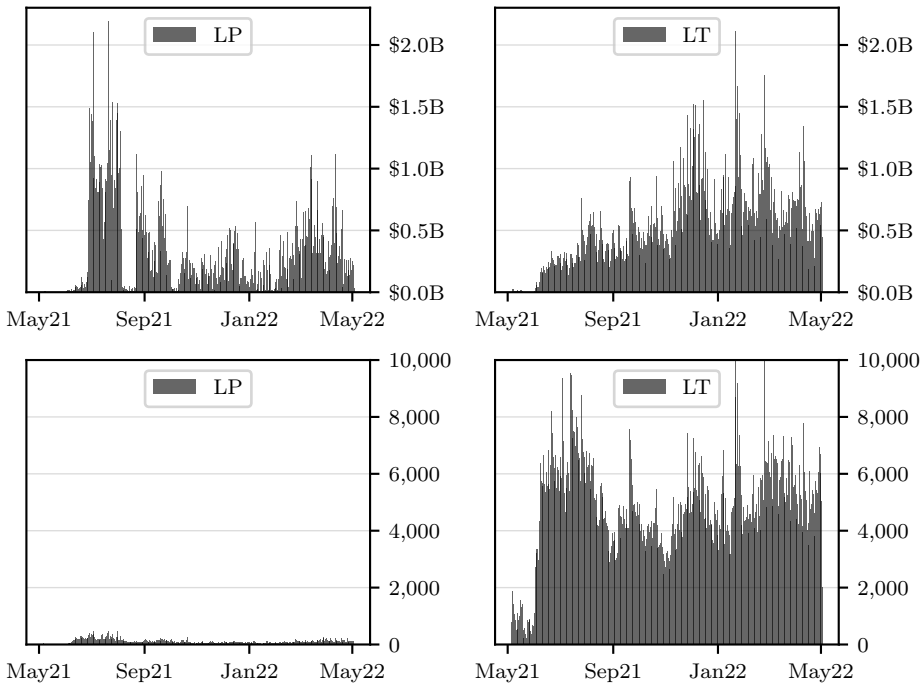


Figure 3.1: Daily transactions for the ETH/USDC pool (B is shorthand for billions), between 5 May 2021 to 5 May 2022. **Top left**: LP transaction volume in USD. **Top right**: LT transaction volume in USD. **Bottom left**: number of LP transactions. **Bottom right**: number of LT transactions.

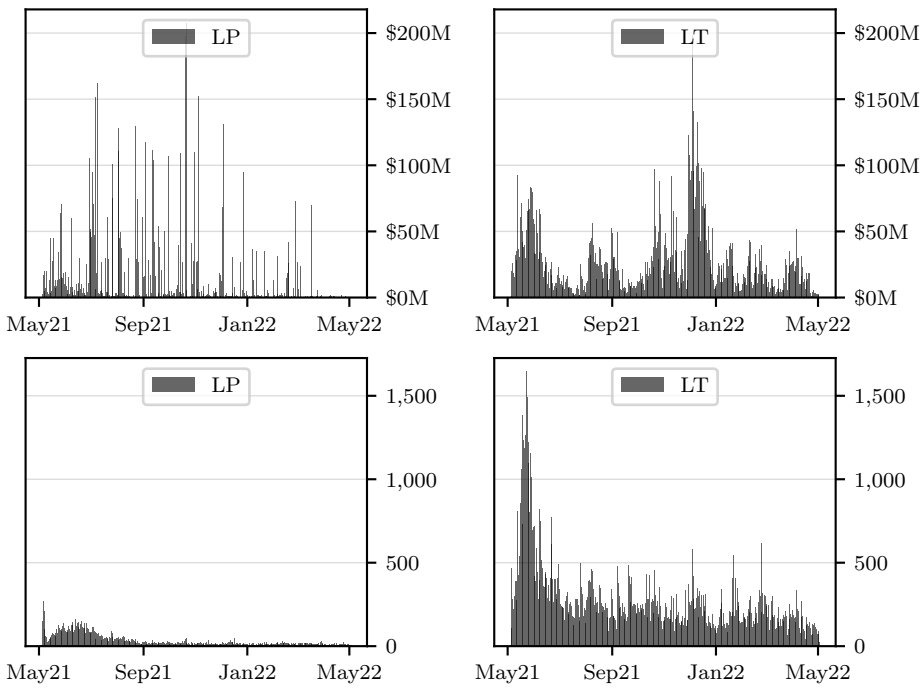


Figure 3.2: Daily transactions for the ETH/DAI pool (M is shorthand for millions), between 5 May 2021 to 5 May 2022. **Top left**: LP transaction volume in USD. **Top right**: LT transaction volume in USD. **Bottom left**: number of LP transactions. **Bottom right**: number of LT transactions.

Next, we study liquidity provision and taking activity within the pools and examine their key characteristics, all of which we use to frame the agent's execution problem.

3.2.2.2 Rates and liquidity dynamics

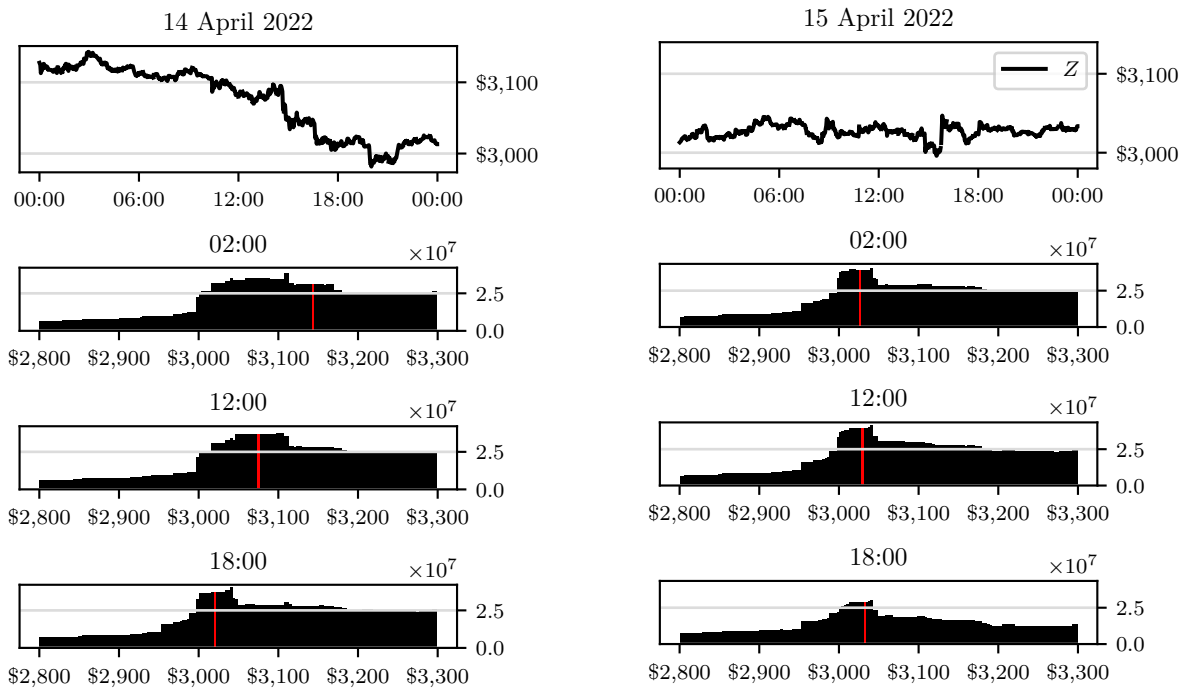
Compared with most CPMMs, Uniswap v3 operates with the concentrated liquidity (CL) feature. In CPMMs without this feature, each LP owns a percentage of the pool, and the fees paid by LTs are distributed to LPs in the same proportion; thus, LPs provide liquidity at all feasible rates. In contrast, LPs in Uniswap v3 specify the range of rates where they supply liquidity. For example, LPs can target a range around the instantaneous rate to earn more fees than those who provide liquidity at rates far from the instantaneous rate. In practice, the continuous space of possible rates is discretised and subdivided in rate intervals whose boundaries are called *ticks*.[7] Two consecutive ticks define a *tick range* and the rate can take values in this range with increments set by the AMM. LPs designate two ticks between which they wish to provide liquidity. Therefore, with CL, the pool is characterised by the available quantities in tick ranges. Finally, in Uniswap v3, fees paid by LTs are distributed among the LPs who had provided liquidity in a range that included the rate at which liquidity was taken.

In CPMMs without CL, the value of κ is the same for all the tick ranges and recall that the value of κ can only change when LPs deposit or withdraw liquidity from the pool. On the other hand, due to the CL feature of Uniswap v3, the depth of the pool may be different when the instantaneous

[7] In Uniswap v3, ticks are specific rates that are used as the boundaries of an LP transaction. In contrast, ticks in LOBs represent the smallest price increment.

rate crosses the boundary of a tick because liquidity is distributed unevenly across rates in the pool. Thus, to estimate the execution cost incurred when trading, one must track the distribution of liquidity across ranges of rates. Finally, in the extreme case where all liquidity is withdrawn from a range of rates around the current rate Z, this effectively constitutes a change in the instantaneous rate of the pool – the next LT transaction will start at a rate different from Z.

To analyse liquidity provision and consumption activity, we reconstruct the supply and demand of liquidity since the inception of Uniswap v3 in May 2021. For example, Figure 3.3 shows the amount of liquidity, given by the depth κ_i, available at each rate range i at 02:00, 12:00, and 18:00, on 14 April 2022 and 15 April 2022 for the ETH/USDC pool.



(a) **Top**: ETH/USDC rates on 14 April 2022 sampled every 60 seconds. **Other plots**: Pool depth κ at 02:00 (top), 10:00 (middle), and 18:00 (bottom). The red bar corresponds to the instantaneous rate.

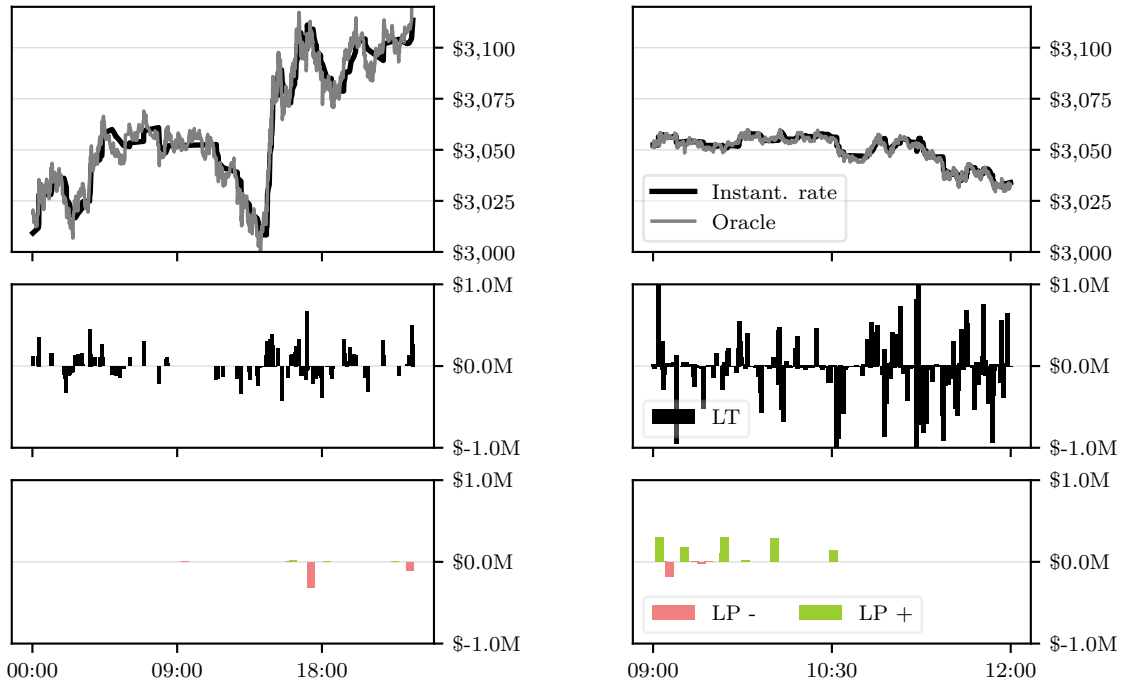
(b) **Top**: ETH/USDC rates on 15 April 2022 sampled every 60 seconds. **Other plots**: Pool depth κ at 02:00 (top), 10:00 (middle), and 18:00 (bottom). The red bar corresponds to the instantaneous rate.

Figure 3.3: Rate and LP dynamics in AMMs.

The bars in the panels of Figure 3.3 represent a tick range. Whenever the instantaneous rate is in a given range, the liquidity of that range defines the depth κ used by the trading function, and hence the execution rate (3.1). When the volume of an LT transaction is large enough to make the instantaneous rate cross a tick where the level of liquidity changes, the AMM treats it as multiple trades, each with a different κ_i; similar to a market order walking the levels of an LOB. Tracking the instantaneous rate, and the liquidity around it, is critical for an agent interacting with an AMM. In Figure 3.3, most of the liquidity is concentrated around the instantaneous rate and the depth is the same over a large range around the rate. The behaviour of LPs is different when volatility is high or low. When Z is volatile, such as on 14 April on the left panel of Figure 3.3, LPs provide

liquidity over a wider range around the rate to earn fees that anticipate large swings in Z. On the other hand, when markets are less volatile, such as on 15 April in the right panel of Figure 3.3, LPs concentrate their liquidity provision more tightly around the rate Z.

Next, we use transaction data from the LOB-based exchange Binance for the same pairs to compare the dynamics in the two trading venues. Figure 3.4 shows the Binance quoted rate and the AMM rate, and liquidity provision activity for both pools between 09:00 and 12:00 on 13 April 2022.



(a) **Top**: ETH/DAI instantaneous rate Z and oracle S between April 13 00:00 and April 13, 2022 12:00. **Middle**: LT Transaction size. **Bottom**: LP transaction size.

(b) **Top**: ETH/USDC Instantaneous rate Z and oracle S between April 13 09:00 and April 13, 2022 12:00. **Middle**: LT Transaction size. **Bottom**: LP transaction size.

Figure 3.4: Rate and LP dynamics in AMMs.

Figure 3.4 shows that Binance rates are more volatile and lead the rates in the AMM. This is not by design, it is a consequence of the higher liquidity in Binance. Currently, it is crucial to consider the rate from a more liquid venue when trading in an AMM. In what follows, we call the leading exchange rate from another trading venue the *oracle*, which in this case is the Binance quoted rate. Similar to Figures 3.1 and 3.2, Figure 3.4 shows that there is more LT than LP activity measured by the frequency of instructions and the size of the orders. During the periods with little trading activity in the ETH/DAI pool, the oracle rate plays a central role to attract LT activity in the AMM whenever the difference between the instantaneous rate and the oracle rate is significant (recall that only liquidity taking trades can change the rate of the pool). The widening of the difference between the two rates triggers LT activity which drives the two exchange rates to converge; i.e., arbitrageurs keep markets in check.

3.2.2.3 Convexity and execution costs

Here, we analyse execution costs implied by the pool reserves and the trading function (3.5). Figure 3.5 shows the historical unitary execution costs as a function of the transaction size Δy and the liquidity for all transactions between 1 March and 31 March 2022 in the ETH/USDC pool. The figure shows that for the same transaction size, it is cheaper to trade in a more liquid pool; clearly, as liquidity increases (higher value of κ), execution costs decrease. The figure shows two unitary execution cost curves, i.e., $\Delta y \mapsto |Z - (\varphi(y + \Delta y) - \varphi(y))|$, corresponding to the same depth $\kappa = 3 \times 10^7$. One curve assumes $Z = 2{,}975$ or $y = 500{,}000$, and the other assumes $Z = 3{,}600$ or $y = 550{,}000$.

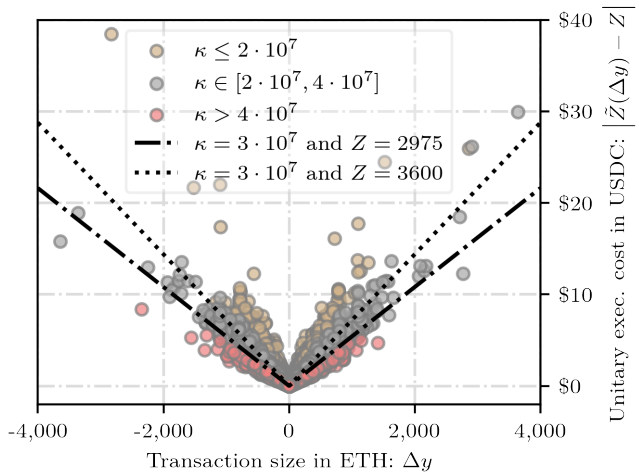


Figure 3.5: Unitary execution costs for all transactions between 1 March and 31 March, 2022 in the ETH/USDC pool. Negative transaction sizes corresponds to buying ETH, and positive sizes to selling ETH. The unitary execution cost is defined in (3.4) and is given for the following ranges of the depth $\kappa : [0, 2 \times 10^7]$, $[2 \times 10^7, 4 \times 10^7]$, and $[4 \times 10^7, +\infty]$.

Next, we analyse the geometry of the constant product trading function to understand how unitary execution costs relate to the depth of the pool and the instantaneous rate Z. Figure 3.6 shows the level function φ for $\kappa = 2{,}500{,}000$. Every point on the curve gives possible values for x and y that result in the same pool depth. The point O corresponds to the current pool quantities x and y. The slope of the tangent at that point gives the current instantaneous rate $Z = -\varphi'(y)$.

A change from O to A in Figure 3.6 is the result of an LT selling 2,500 ETH. A change from O to B is the result of an LT buying 2,500 ETH. The new rates, after these transactions, are given by the slopes at the new points A and B, respectively. When an LT sells $\Delta y = 2{,}500$ ETH, the unitary execution rate $\Delta x / \Delta y$ is given by the slope of the line (OA). Similarly, the slope of the line (OB) gives the unitary rate for buying Δy. On the other hand, the unitary execution cost is given by the difference between the slope of the lines and the slope of the tangent at point O; the magnitude of this difference depends on the curvature of φ in the neighbourhood of O. This curvature is proportional to the convexity of the level function and can be approximated by the second-order Taylor polynomial $\frac{1}{2}\varphi''(y_O)\,\Delta y^2$. A higher degree of convexity, i.e., more curvature

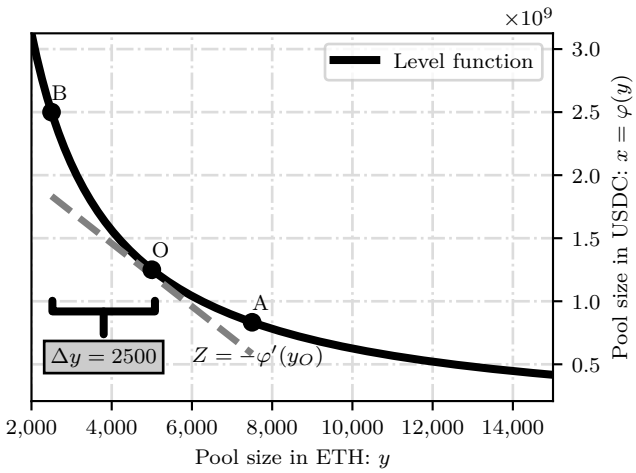


Figure 3.6: Geometry of the constant product trading function. The figure shows the function $\varphi(y) = x$ where x is the quantity of USDC, and y is the quantity of ETH in the pool.

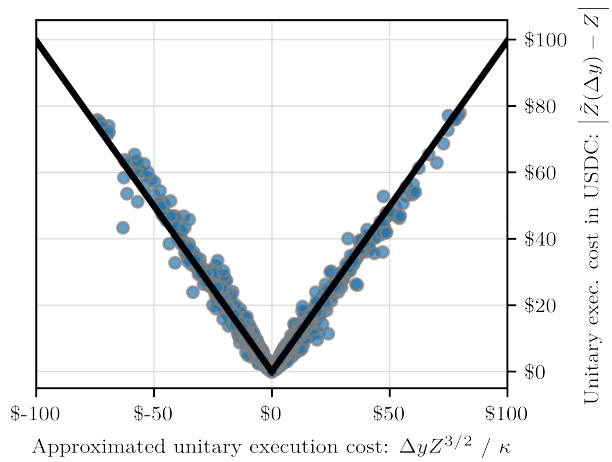


Figure 3.7: Scatter plot of transaction costs and the approximation (3.6) for all transactions in the ETH/USDC pool between 5 May 2021 and 5 May 2022.

around point O, does not change the slope of the tangent at point O, but changes the slopes of the lines (OA) and (OB). The convexity of the level function is given by

$$\varphi''(y) = \frac{2\,\kappa^2}{y^3} = \frac{2\,Z^{3/2}}{\kappa}\,.$$

Therefore, the execution rate obtained for buying or selling Δy is always less advantageous than the instantaneous rate Z because the level function φ is convex. Clearly, as the depth κ increases, the convexity of the level function is less pronounced. For orders of "small size" one can approximate the unitary execution cost in (3.4) with

$$|Z - \tilde{Z}(\Delta y)| = \frac{1}{\kappa}\,Z^{3/2}\,|\Delta y|\,, \tag{3.6}$$

or equivalently approximate the execution rate with

$$\tilde{Z}(\Delta y) = Z - \frac{1}{\kappa}\,Z^{3/2}\,\Delta y\,.$$

To show the extent to which the convexity of the level function captures the execution cost, we use transaction data from the ETH/USDC pool from inception on 5 May 2021 until 5 May 2022, approximately 1,757,000 LT transactions, to compare unitary execution costs incurred by LTs with our approximation in (3.6). Figure 3.7 shows a scatter plot of the observed unitary execution costs from transaction data and the approximation $\left(Z^{3/2}/\kappa\right)\Delta y$. Recall that negative values of Δy are buy orders and positive values are sell orders. The figure shows that the expression in (3.6) is a good approximation, and that the convexity of the level function φ can be used to model the unitary execution cost incurred when interacting in CPMMs.

In an optimal trading framework, the LT controls the speed ν_t at which she sends orders to the AMM. Now, assume that the LT trades the quantity $\Delta y_t = \nu_t\,\Delta t$, where Δt is a fixed time-step that

determines the LT's frequency of trading and ν_t is fixed during the time interval. The execution rate for Δy is $\tilde{Z}(\Delta y) = Z - Z^{3/2}\,\nu_t\,\Delta t\,/\,\kappa$. Thus, to reflect the LT's pace of trading we write the LT's execution rate as

$$\tilde{Z}_t = Z_t - \frac{\eta}{\kappa}\,Z_t^{3/2}\,\nu_t\ , \tag{3.7}$$

where the parameter η scales the execution costs according to the LT's trading frequency Δt. Note that the continuous-time formulation of the problem in (3.7) is used in the algorithmic trading literature to solve optimal execution problems, which are by nature discrete.

Due to the convexity of the level function, it is sub-optimal to execute large orders in one trade.[8] An optimal trading framework, similar to those developed for traditional LOB-based markets, should balance the trade-off between execution costs and rate risk. The execution cost in CFMMs is similar to the cost of "walking the book" when trading in LOBs, sometimes referred to as the temporary price impact. The difference between the temporary price impact and the CFMM execution cost is that in the CFMM we have a deterministic closed-form expression for the execution cost as a function of the depth and the rate, both of which can be employed by LTs to estimate execution costs. On the other hand, in LOBs, traders usually rely on historical data analysis and assumptions to obtain an estimate of the execution costs. In LOBs, it is generally assumed that temporary impact is a linear function of the speed of trading where the slope of the function is assumed to be fixed; see Cartea et al. (2015) and Guéant (2016).

As in LOBs, there is uncertainty in execution costs received by LTs. On blockchains, rate and liquidity updates occur at the block validation frequency. For instance, in the *Ethereum* network hosting the most popular AMMs, a block is validated every 13 seconds, on average. Hence, the most up-to-date information available is the previous block's rate and liquidity data. Transactions are grouped and executed in a block which is externally validated. Within the block, the transactions form a queue that determines the priority with which they are executed. Market participants can pay *tips* to gain priority in the block. Finally, every transaction sent on the network pays *gas fees*. Gas is a unit that measures computational effort of network transactions, and all market participants pay gas fees to the network – fees are proportional to the algorithmic complexity of the functions a trader invokes in the AMM. Consequently, there is randomness regarding the exact execution costs.

Below, in Section 3.3, we propose a trading model (Model I) in AMMs where execution costs are given by the convexity of the level function, where the depth κ does not change over the agent's trading window, and where it is assumed that there is an oracle rate. Model I introduces a method that uses piecewise constant execution costs to obtain a closed-form approximation strategy that adjusts to the convexity of the level function. We use Model I in our performance analysis of Section 3.5. In the future, an oracle rate may become less relevant because activity in AMMs will increase and significant rate discrepancies will very seldom appear. Thus, in Section 3.4 we propose Model II for liquid AMMs where the depth κ of the pool is stochastic and where an oracle rate is redundant.

[8] This is a consequence of the trading activity of other market participants.

3.3 Model I: optimal trading with an oracle rate

We consider the problem of an LT who trades in a CPMM and wishes to exchange a large position in asset Y into asset X or to execute a statistical arbitrage in the pair. In both cases, the LT uses rate information from the pool in the CPMM and from another more liquid exchange in which the oracle rate S is the price of Y in terms of that of X. The LT acknowledges that rate formation does not take place in the AMMs.

The depth κ of the pool is assumed to be constant during the execution window $[0, T]$, where $T > 0$. Despite intraday changes in the depth of the pool, this assumption does not have a material effect on the LT's execution problem. Currently, the activity and liquidity in AMMs is such that the depth κ is constant for periods of time which are longer than the trading horizon T considered by the LT.[9]

In the future, activity in AMMs is expected to increase, and so will the informational content of the rates implied by the pool of AMMs. An increase in activity of the pool would affect our modelling choices in two ways. One, the innovations in depth κ will occur more often, so the value of κ cannot be regarded as constant throughout the execution window. Two, the oracle rate becomes redundant because the rates in the AMMs become efficient, so they incorporate all the information available to market participants – i.e., the discrepancies with rates and prices in other trading venues are negligible and economically insignificant. Therefore, below in Section 3.4 we propose Model II to solve the LT's execution problem with stochastic dynamics for the depth κ and without an external oracle rate.

We proceed with the setup of Model I, for which we fix a filtered probability space $\big(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{F} = (\mathcal{F}_t)_{t\in[0,T]}\big)$ satisfying the usual conditions, where $\mathbb{F}$ is the natural filtration generated by the collection of observable stochastic processes that we define below.

The LT must liquidate a position $\tilde{y}_0$ in asset Y over the period of time $[0, T]$, and her wealth is valued in terms of asset X. We introduce the progressively measurable oracle rate process $(S_t)_{t\in[0,T]}$ with dynamics

$$dS_t = \sigma\, S_t\, dW_t\,, \tag{3.8}$$

where the volatility parameter σ is a nonnegative constant, and $(W_t)_{t\in[0,T]}$ is a standard Brownian motion.

As discussed above, when an LT sends a trade to the AMM, the rate impact received by the order is encoded in the trading function $f(x, y)$. As in traditional models for optimal execution (see e.g., Cartea et al. (2015) and Guéant (2016)), the rate impact received by the LT's trade is a function of the trading speed. In AMMs, rate impact is a function of the trading speed and of the rate Z and the depth κ. Specifically, we write the difference between the execution rate $\tilde{Z}$ and the instantaneous rate Z as in (3.7), i.e.,

$$\tilde{Z} - Z = -\frac{\eta}{\kappa}\, Z^{3/2}\, \nu\,. \tag{3.9}$$

[9] Clearly, the LT cannot know when a change in liquidity of the pool occurs. Thus, there is the possibility that within the trading window of the execution programme the value of κ changes.

The term on the right-hand side of (3.9) is the rate impact function of the AMM, which is determined by the convexity of the level function φ, see (3.6), and depends on the instantaneous rate Z_t of the pool at the time the liquidity taking order is executed. The key difference between the functional form of the execution costs in (3.9) and those in the equity LOB literature is that in general, the price impact functions proposed for LOB models do not depend on the price of the asset, see Cartea et al. (2015).

The LT trades at the speed $(\nu_t)_{t\in[0,T]}$, so her inventory $(\tilde{y}_t)_{t\in[0,T]}$ evolves as

$$d\tilde{y}_t = -\nu_t \, dt \,, \tag{3.10}$$

where, for simplicity, trading fees are zero. We do not restrict the speed in (3.10) to be positive; if $\nu > 0$ the LT sells the asset and if $\nu < 0$ the LT buys the asset. When the initial inventory is $\tilde{y}_0 > 0$ (resp. $\tilde{y}_0 = 0$) the LT executes a liquidation (resp. speculation) programme.

When an LT trades at speed ν, the quantity of Y swapped at every instant in time is given by $\nu \, dt$, so the dynamics of her holdings in asset X are given by

$$d\tilde{x}_t = \tilde{Z}_t \, \nu_t \, dt = \left(Z_t - \frac{\eta}{\kappa} \, Z_t^{3/2} \, \nu_t \right) \, \nu_t \, dt \,. \tag{3.11}$$

Here, the execution cost is stochastic and its dynamics are known; see Barger and Lorig (2019) and Fouque et al. (2022) for similar frameworks where the price impact of trading is stochastic.

In the absence of market frictions, continuous arbitrage between the oracle rate and the rate quoted in the AMM would make rates converge so that $S_t = Z_t$ at any time t. However, exchanges and AMMs are not frictionless, and the oracle rate is the most efficient rate (i.e., reflects all the information available) so we consider the following dynamics for $(Z_t)_{t\in[0,T]}$:

$$dZ_t = \beta \, (S_t - Z_t) \, dt + \gamma \, Z_t \, dB_t \;, \tag{3.12}$$

where $\beta > 0$ is the mean-reverting parameter, $\gamma > 0$ quantifies the dispersion induced by the trading flow that drives the AMM instantaneous rate Z_t away from the oracle rate S_t, and $(B_t)_{t\in[0,T]}$ is a standard Brownian motion independent of $(W_t)_{t\in[0,T]}$.

3.3.1 Optimal trading strategy: execution and statistical arbitrage

The LT maximises her expected terminal wealth in units of X while penalising inventory in Y. The set of admissible strategies is

$$\mathcal{A}_t = \left\{ (\nu_s)_{s\in[t,T]}, \ \mathbb{R}\text{-valued}, \ \mathbb{F}\text{-adapted, and } \int_t^T |\nu_s|^2 \, ds < +\infty, \quad \mathbb{P}\text{-a.s.} \right\} . \tag{3.13}$$

We write $\mathcal{A} := \mathcal{A}_0$ and let $\nu \in \mathcal{A}$. The performance criterion of the LT, who trades at speed ν, is the function $u^\nu \colon [0,T] \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \to \mathbb{R}$ given by

$$u^\nu(t, \tilde{x}, \tilde{y}, Z, S) = \mathbb{E}_{t,\tilde{x},\tilde{y},Z,S} \left[\tilde{x}_T^\nu + \tilde{y}_T^\nu \, Z_T - \alpha \, (\tilde{y}_T^\nu)^2 - \phi \int_t^T (\tilde{y}_s^\nu)^2 \, ds \right] , \tag{3.14}$$

and the LT's value function $u : [0,T] \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \to \mathbb{R}$ is

$$u(t, \tilde{x}, \tilde{y}, Z, S) = \sup_{\nu \in \mathcal{A}} u^{\nu}\left(t, \tilde{x}, \tilde{y}, Z, S\right) .$$

The first term on the right-hand side of (3.14) represents the LT's holdings in asset X at the end of the trading window. The second term represents the LT's earnings from liquidating her remaining inventory at the final time T at rate Z_T. The third term represents an inventory penalty where the parameter $\alpha > 0$ quantifies the aversion of the LT to holding non-zero inventory at time T; the units of ϕ are such that the penalty is in units of X. Finally, the last term on the right-hand side of (3.14) represents a running inventory penalty where the parameter $\phi \geq 0$ quantifies the urgency of the LT to liquidate inventory; the units of α are such that the penalty is in units of X. Note that the third term on the right-hand side of (3.14) is sometimes interpreted in the algorithmic trading literature as the 'cost' of liquidating the final inventory $\tilde{y}_T$ at time T. Using this interpretation, and in light of the results of the previous section, this term should be proportional to $Z_T^{3/2}$. Here we interpret this term as penalisation term and we observe that for α sufficiently large, $\alpha\ (\tilde{y}_T^{\nu})^2$ is greater than expected execution costs of terminal inventory.

In Appendix A.3, we use the tools of stochastic control to study the optimisation problem in (3.14). The functional form of the execution costs leads to a system of PDEs, one of which is a semilinear PDE which we cannot solve in closed-form; see (A.9). The optimal trading speed in feedback form is a function of the solution to the semilinear PDE, see (A.6), and one can compute the optimal trading speed with a numerical scheme. However, in our case, the numerical scheme is computationally expensive because the semilinear PDE requires a thin grid and a linearisation iterative method at each time step to transform the nonlinear problem into a sequence of linear problems.

We refer to the optimal strategy obtained with a numerical scheme as the numerical approximation strategy. Clearly, the numerical approximation strategy takes too long both to compute and to implement by the LT in real time. In practice, the profitability of execution and statistical arbitrage strategies relies on computing the strategy and instructing the AMM within very short periods of time (e.g., milliseconds). Thus, because speed is paramount for LTs, below we derive a strategy in closed-form that can be deployed by the LT in real time. We refer to this strategy as the closed-form approximation strategy. Finally, to assess the precision of the closed-form approximation strategy, Subsection 3.3.4 compares it with the numerical approximation strategy (A.6) derived in A.3.

3.3.2 Constant impact parameter strategy

To obtain a trading scheme that can be implemented by the LT in real time, we first derive a strategy where the impact parameter of the execution cost is deterministic. We use this strategy as the building block for the LT's closed-form approximation strategy we derive below. Accordingly, we write the execution cost in (3.9) as

$$\tilde{Z} - Z = -\eta\, \zeta\, \nu\,,$$

where $\zeta > 0$ is the impact parameter and recall that the value of η depends on the LT's trading frequency. With fixed executions costs, the LT can derive a closed-form optimal trading strategy $\left(\nu_t^{\star,\zeta}\right)_{t\in[0,T]}$ for a given value of the parameter ζ.

For each ζ the set of admissible strategies is

$$\mathcal{A}_t^\zeta = \left\{ (\nu_s)_{s\in[t,T]},\ \mathbb{R}\text{-valued},\ \mathbb{F}\text{-adapted, and } \int_t^T |\nu_s|^2\, ds < +\infty,\quad \mathbb{P}\text{-a.s.} \right\},$$

and we write $\mathcal{A}^\zeta := \mathcal{A}_0^\zeta$. We consider an optimal trading problem in which the LT trades at speed $\left(\nu_t^\zeta\right)_{t\in[0,T]}$, so the inventory $\left(\tilde{y}_t^\zeta\right)_{t\in[0,T]}$ evolves as

$$d\tilde{y}_t^\zeta = -\nu_t^\zeta\, dt\,,$$

where we do not restrict the speed to be positive, and recall that trading fees are zero.

When an LT trades at speed ν^ζ, the quantity of Y swapped at every instant in time is given by $\nu^\zeta\, dt$, so the dynamics of the LT's holdings in asset X are given by

$$d\tilde{x}_t^\zeta = \tilde{Z}_t\, \nu_t^\zeta\, dt = \left(Z_t - \eta\,\zeta\,\nu_t^\zeta\right)\, \nu_t^\zeta\, dt\,.$$

Let $\nu^\zeta \in \mathcal{A}^\zeta$. The performance criterion of the LT, who trades at speed ν^ζ, is a function $u^{\nu^\zeta}\colon [0,T]\times\mathbb{R}^4 \to \mathbb{R}$ given by

$$u^{\nu^\zeta}(t,\tilde{x},\tilde{y},Z,S) = \mathbb{E}_{t,\tilde{x},\tilde{y},Z,S}\left[\tilde{x}_T^{\nu^\zeta} + \tilde{y}_T^{\nu^\zeta}\, Z_T - \alpha\left(\tilde{y}_T^{\nu^\zeta}\right)^2 - \phi\int_t^T \left(\tilde{y}_s^{\nu^\zeta}\right)^2 ds\right]\,,$$

and the value function $u_\zeta\colon [0,T]\times\mathbb{R}^4 \to \mathbb{R}$ of the LT is given by

$$u_\zeta(t,\tilde{x},\tilde{y},Z,S) = \sup_{\nu^\zeta\in\mathcal{A}^\zeta} u^{\nu^\zeta}(t,\tilde{x},\tilde{y},Z,S)\,. \tag{3.15}$$

Next, we solve the above dynamic optimisation problem for each value of $\zeta > 0$ and derive the optimal liquidation strategy. The value function (3.15) is the unique classical solution to the Hamilton–Jacobi–Bellman (HJB) equation

$$\begin{aligned} 0 = {} & \partial_t w_\zeta - \phi\,\tilde{y}^2 + \beta\,(S-Z)\,\partial_Z w_\zeta + \frac{1}{2}\gamma^2 Z^2\,\partial_{ZZ} w_\zeta + \frac{1}{2}\sigma^2 S^2\,\partial_{SS} w_\zeta \\ & + \sup_{\nu\in\mathbb{R}}\left(\left(\nu\,Z - \eta\,\zeta\,\nu^2\right)\partial_{\tilde{x}} w_\zeta - \nu\,\partial_{\tilde{y}} w_\zeta\right)\,, \end{aligned} \tag{3.16}$$

with terminal condition

$$w_\zeta(T,\tilde{x},\tilde{y},Z,S) = \tilde{x} + \tilde{y}\,Z - \alpha\,\tilde{y}^2\,. \tag{3.17}$$

The form of the terminal condition (3.17) suggests the ansatz

$$w_\zeta(t,\tilde{x},\tilde{y},Z,S) = \tilde{x} + \tilde{y}\,Z + \theta_\zeta(t,\tilde{y},Z,S)\,,$$

where the first two terms on the right-hand side represent the mark-to-market value of the LT's holdings at the instantaneous rate of the AMM. The last term represents the additional value obtained by the LT when following the optimal strategy. The ansatz is justified by the following proposition, for which a proof is straightforward.

Proposition 1. *Assume there exists a function* $\theta_\zeta \in C^{1,1,2,2}\left([0,T]\times\mathbb{R}^3\right)$ *which solves*

$$\begin{aligned} 0 = &\partial_t\theta_\zeta - \phi\,\tilde{y}^2 + \beta\,(S-Z)\,(\tilde{y}+\partial_Z\theta_\zeta) + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta_\zeta + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta_\zeta \\ &+ \sup_{\nu\in\mathbb{R}}\left(-\eta\,\zeta\,\nu^2 - \nu\,\partial_{\tilde{y}}\theta_\zeta\right)\ , \end{aligned} \tag{3.18}$$

on $[0,T)\times\mathbb{R}^3$*, with terminal condition*

$$\theta_\zeta(T,\tilde{y},Z,S) = -\alpha\,\tilde{y}^2\ . \tag{3.19}$$

Then, the function $w_\zeta\colon [0,T]\times\mathbb{R}^4\to\mathbb{R}$ *defined by*

$$w_\zeta(t,\tilde{x},\tilde{y},Z,S) = \tilde{x} + \tilde{y}\,Z + \theta_\zeta(t,\tilde{y},Z,S)\ ,$$

is a solution to (3.16) *on* $[0,T)\times\mathbb{R}^4$*, with terminal condition* (3.17) *.*

Next, solve the first order condition of the supremum term in (3.18) to obtain the LT's optimal trading speed in feedback form

$$\nu^{\zeta,\star} = -\frac{1}{2\,\eta\,\zeta}\,\partial_{\tilde{y}}\theta_\zeta\ . \tag{3.20}$$

Substitute (3.20) into (3.18) to write

$$\begin{aligned} \partial_t\theta_\zeta = &-\phi\,\tilde{y}^2 + \beta\,(S-Z)\,(\tilde{y}+\partial_Z\theta_\zeta) + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta_\zeta + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta_\zeta \\ &+ \frac{1}{4\,\eta\,\zeta}\,(\partial_{\tilde{y}}\theta_\zeta)^2\ . \end{aligned} \tag{3.21}$$

Finally, simplify (3.21) with the ansatz

$$\begin{aligned} \theta_\zeta(t,\tilde{y},Z,S) = &A_\zeta(t)\,\tilde{y}^2 + B_\zeta(t)\,Z\,\tilde{y} + C_\zeta(t)\,\tilde{y}\,S + D_\zeta(t)\,\tilde{y} + E_\zeta(t)\,Z^2 \\ &+ F_\zeta(t)\,S^2 + G_\zeta(t)\,Z\,S + H_\zeta(t)\,Z + I_\zeta(t)\,S + J_\zeta(t)\ , \end{aligned}$$

which is justified by the following proposition, for which a proof is straightforward.

Proposition 2. *Assume there exist functions* $A_\zeta\in C^1([0,T])$, $B_\zeta\in C^1([0,T])$, $C_\zeta\in C^1([0,T])$, $D_\zeta\in C^1([0,T])$, $E_\zeta\in C^1([0,T])$, $F_\zeta\in C^1([0,T])$, $G_\zeta\in C^1([0,T])$, $H_\zeta\in C^1([0,T])$, $I_\zeta\in$

$C^1([0,T])$, *and* $J_\zeta \in C^1([0,T])$ *which solve the system of ODEs*

$$\begin{cases} A'_\zeta(t) = \phi - \dfrac{A_\zeta(t)^2}{\eta\,\zeta} \ , \\ B'_\zeta(t) = \beta + \beta\, B_\zeta(t) - \dfrac{A_\zeta(t)\, B_\zeta(t)}{\eta\,\zeta} \ , \\ C'_\zeta(t) = -\beta - \beta\, B_\zeta(t) - \dfrac{A_\zeta(t)\, C_\zeta(t)}{\eta\,\zeta} \ , \\ D'_\zeta(t) = -\dfrac{A_\zeta(t)\, D_\zeta(t)}{\eta\,\zeta} \ , \\ E'_\zeta(t) = -(\gamma^2 - 2\beta)\, E_\zeta(t) - \dfrac{B_\zeta(t)^2}{4\,\eta\,\zeta} \ , \\ F'_\zeta(t) = -\beta\, G_\zeta(t) - \sigma^2\, F_\zeta(t) - \dfrac{C_\zeta(t)^2}{4\,\eta\,\zeta} \ , \\ G'_\zeta(t) = -2\,\beta\, E_\zeta(t) + \beta\, G_\zeta(t) - \dfrac{B_\zeta(t)\, C_\zeta(t)}{2\,\eta\,\zeta} \ , \\ H'_\zeta(t) = \beta\, H_\zeta(t) - \dfrac{B_\zeta(t)\, D_\zeta(t)}{2\,\eta\,\zeta} \ , \\ I'_\zeta(t) = -\beta\, H_\zeta(t) - \dfrac{C_\zeta(t)\, D_\zeta(t)}{2\,\eta\,\zeta} \ , \\ J'_\zeta(t) = -\dfrac{D_\zeta(t)^2}{4\,\eta\,\zeta} \ , \end{cases} \tag{3.22}$$

on $[0,T)$, *with terminal conditions*

$$A_\zeta(T) = -\alpha \ ,$$
$$B_\zeta(T) = C_\zeta(T) = D_\zeta(T) = E_\zeta(T) = F_\zeta(T) = G_\zeta(T) = H_\zeta(T) = I_\zeta(T) = J_\zeta(T) = 0 \ .$$

Then, the function $\theta_\zeta : [0,T] \times \mathbb{R}^3 \to \mathbb{R}$ *defined by*

$$\begin{aligned} \theta_\zeta(t,\tilde{y},Z,S) = {} & A_\zeta(t)\,\tilde{y}^2 + B_\zeta(t)\, Z\,\tilde{y} + C_\zeta(t)\,\tilde{y}\, S + D_\zeta(t)\,\tilde{y} + E_\zeta(t)\, Z^2 \\ & + F_\zeta(t)\, S^2 + G_\zeta(t)\, Z\, S + H_\zeta(t)\, Z + I_\zeta(t)\, S + J_\zeta(t) \ , \end{aligned}$$

solves (3.18) *over* $[0,T) \times \mathbb{R}^3$ *with terminal condition* (3.19) *.*

The system of ODEs (3.22) can be solved sequentially. First, solve the ODE in A_ζ, and then solve for B_ζ , C_ζ , D_ζ , E_ζ , G_ζ , F_ζ , H_ζ , I_ζ , J_ζ , respectively. The system in (3.22) admits a

unique solution, which is given by

$$\begin{cases} A_\zeta(t) = \sqrt{\phi\,\eta\,\zeta}\,\tanh\left(\frac{\sqrt{\phi}}{\sqrt{\eta\,\zeta}}\,(T-t) + \operatorname{arctanh}\left(-\frac{\alpha}{\sqrt{\phi\,\eta\,\zeta}}\right)\right)\,, \\ B_\zeta(t) = \int_t^T \beta \exp\left(\int_t^s \left(\beta - \frac{1}{\eta\,\zeta} A_\zeta(u)\right) du\right) ds\,, \\ C_\zeta(t) = -B_\zeta(t)\,, \\ D_\zeta(t) = 0 \\ E_\zeta(t) = -\int_t^T \exp\left(-(\gamma^2 - 2\,\beta)(t-s)\right) \frac{1}{4\,\eta\,\zeta}\, B_\zeta(s)^2 ds\,, \\ F_\zeta(t) = -\int_t^T \exp\left(-\sigma^2(t-s)\right) \left(\beta\, G_\zeta(s) + \frac{1}{4\,\eta\,\zeta}\, C_\zeta(s)^2\right) ds\,, \\ G_\zeta(t) = -\int_t^T \exp\left(\beta\,(t-s)\right) \left(2\,\beta\, E_\zeta(s) - \frac{1}{2\,\eta\,\zeta}\, B_\zeta(s)^2\right) ds\,, \\ H_\zeta(t) = 0\,, \\ I_\zeta(t) = 0\,, \\ J_\zeta(t) = 0\,. \end{cases}$$

The optimal trading strategy in feedback form (3.20) over $[0,T] \times \mathbb{R}^3$ is now given by

$$\nu^{\zeta,\star}(t,\tilde{y},Z,S) = -\frac{1}{\eta\,\zeta}\, A_\zeta(t)\,\tilde{y} + \frac{1}{2\,\eta\,\zeta}\, B_\zeta(t)\,(S-Z)\,. \tag{3.23}$$

The first term on the right-hand side of (3.23) is the optimal liquidation rate in the continuous Almgren-Chriss model. The second term is an arbitrage component; it accounts for the spread between the instantaneous rate Z and the oracle rate S.

3.3.3 The closed-form approximation strategy

Here, we use a family of closed-form strategies of the type in (3.23) to derive a piecewise-defined trading strategy which approximates the optimal trading speed in feedback form (A.11). Specifically, we partition the space of the rate Z into strips and define a piecewise strategy which uses a different impact parameter ζ in each different strip. Finally, we show that as the width of the strip becomes arbitrarily small, the piecewise strategy converges to the closed-form approximation strategy.

Let $\left\{Z_0^N, \ldots, Z_N^N\right\}$ be a partition of $\left[\underline{Z}, \overline{Z}\right]$, where $0 < \underline{Z} < \overline{Z}$. Here $\underline{Z}$ and $\overline{Z}$ are are such that the LT has high confidence that the rate Z will be within the range $\left[\underline{Z}, \overline{Z}\right]$ during the execution window. Next, for each $N \in \mathbb{N}$ and $j \in \{0, \ldots, N\}$ we define

$$Z_j^N := \underline{Z} + \frac{j}{N}\left(\overline{Z} - \underline{Z}\right) \quad \text{and} \quad \zeta_j^N = \frac{1}{\kappa}\left(Z_j^N\right)^{3/2}\,. \tag{3.24}$$

In the remainder of this section, we simplify the notation and use $\nu^{\star,j,N}$ instead of $\nu^{\zeta_j^N,\star}$ to denote the optimal trading strategy with impact parameter ζ_j^N.

Whenever Z is arbitrarily close to Z_j^N the impact parameter in (3.9) can be approximated by ζ_j^N. Thus, to construct the approximate trading strategy, we first define a strategy $\nu^{\star,N}$ that uses

the closed-form optimal trading speed $\nu^{\star,j,N}$ to approximate the optimal trading speed whenever the rate is close to ζ_j^N. We define the piecewise-defined trading speed $\nu^{\star,N}\colon [0,T]\times\mathbb{R}^3\to\mathbb{R}$

$$\nu^{\star,N}(t,\tilde{y},Z,S) = \nu^{\star,0,N}(t,\tilde{y},Z,S)\,\mathbb{1}_{Z<Z_1^N} + \sum_{j=1}^{N-1}\nu^{\star,j,N}(t,\tilde{y},Z,S)\,\mathbb{1}_{Z\in[Z_j^N,Z_{j+1}^N)} + \nu^{\star,N,N}(t,\tilde{y},Z,S)\,\mathbb{1}_{Z\geq Z_N^N}\,.$$

The strategy $\nu^{\star,N}(t,\tilde{y},Z,S)$ has first-type discontinuity points; specifically, it is discontinuous over $[0,T]\times\mathbb{R}\times\{Z_j^N\}\times\mathbb{R}$ for each $j\in\{1,\dots,N\}$ because for each $\left(t,\tilde{y},Z_{j+1}^N,S\right)\in[0,T]\times\mathbb{R}^2$ we have $\nu^{\star,j,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)\neq\nu^{\star,j+1,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)$.

The theorem below shows how to partition $\left[\underline{Z},\overline{Z}\right]$ to make the discontinuities in $\nu^{\star,N}(t,\tilde{y},Z,S)$ arbitrarily small. Furthermore, when the distance between points in the partition becomes sufficiently small, the sequence of piecewise-defined optimal strategies $\left\{\nu^{\star,N}\right\}_{N\in\mathbb{N}}$ converges uniformly to a continuous closed-form approximation strategy which we use in our performance study of Section 3.5.

Theorem 1. *For each $\varepsilon>0$ there exists $N\in\mathbb{N}$ such that*

$$\max_{j=1,\dots,N}\left|\nu^{\star,j,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)-\nu^{\star,j+1,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)\right|<\varepsilon\,. \tag{3.25}$$

Furthermore, for each $N\in\mathbb{N}$, let $\tilde{\nu}^{\star,N}:=\nu^{\star,N}\Big|_{[0,T]\times\mathbb{R}\times[\underline{Z},\overline{Z}]\times\mathbb{R}}$. Then, the sequence $\{\tilde{\nu}^{\star,N}\}$ converges to $\tilde{\nu}^{\star}$ uniformly in $[0,T]\times\mathbb{R}\times\left[\underline{Z},\overline{Z}\right]\times\mathbb{R}$, where

$$\tilde{\nu}^{\star}(t,\tilde{y},Z,S) = -\frac{\kappa}{\eta}\,Z^{-3/2}\,A(t,Z)\,\tilde{y}+\frac{\kappa}{2\,\eta}\,Z^{-3/2}\,B(t,Z)\,(S-Z)\,, \tag{3.26}$$

and

$$\begin{aligned} A(t,Z) &= \sqrt{\frac{\phi\,\eta\,Z^{3/2}}{\kappa}}\tanh\left(\frac{\sqrt{\phi\,\kappa}}{\sqrt{\eta\,Z^{3/2}}}t+\operatorname{arctanh}\left(-\frac{\alpha\sqrt{\kappa}}{\sqrt{\phi\,\eta\,Z^{3/2}}}\right)\right)\,,\\ B(t,Z) &= \int_t^T\beta\exp\left(\int_s^t\left(\beta-\frac{\kappa}{\eta\,Z^{3/2}}A(u,Z)\right)du\right)ds\,. \end{aligned} \tag{3.27}$$

For a proof see A.1 .

3.3.4 Comparison with the numerical approximation strategy

In this subsection, we use a Euler scheme to compute the numerical approximation strategy in (A.9), where execution costs are not piecewise constant. We compare this numerical solution with the closed-form approximation strategy in (3.26).

The numerical approximation strategy uses a three-dimensional grid; one dimension is time, one is the rate Z, and one is the oracle rate S. We consider a wide interval for the values of Z and S, and use a Neumann boundary condition for both space variables (the derivatives are zero at the boundaries) because the value function should not vary significantly for values of the rates Z and S that are considerably far from the observed rates. For the semilinear PDE in (A.9), we use the Picard iterative method to linearise the problem at every time level of the grid. More precisely,

at each time level, we use Picard iterations to linearise the term that is quadratic in the unknown solution, which is replaced by the product of the unknown solution and the most recent computed solution obtained at the previous time level. The iterative approximation process is carried on until the difference between two consecutive numerical solutions is smaller than a given threshold.

Figure 3.8 compares the numerical approximation strategy (A.11) and the closed-form approximation strategy (3.26) for different values of the inventory, the rate Z, and the oracle rate S. Figure 3.8 indicates that the closed-form and the numerical approximation strategies are significantly close and capture the same financial effects. In particular, the strategies clearly depend on the spread $S - Z$ and the inventory $\tilde{y}$. When the LT has zero inventory, the strategy is mostly speculative because the strategy buys asset Y when the oracle rate S is above the rate Z, and sells otherwise; recall that the LT buys asset Y when the trading speed ν is positive, and sells otherwise. When the LT has a positive (negative) inventory, the optimal strategy buys (sells) asset Y only when S is significantly higher (lower) than Z.

Figure 3.8 shows that the absolute difference between both strategies increases as the difference between the rate Z and S increases, and the difference is minimal when the rates are equal. In practice, by arbitrage, the rates Z and S are aligned by LTs in the pool, so differences between rates are small. For instance, with the market data we use in Section 3.5, the average absolute difference between rates is 2 USD (0.07%) in the liquid pool and 10 USD (0.37%) in the illiquid pool. Finally, it is unclear whether the difference between the optimal speed obtained with a numerical scheme and the closed-form approximation strategy shown in the lower panels of Figure 3.8 stem from the numerical approximation or from the closed-form approximation strategy, but in either case, the difference has no bearing on the results we discuss in Section 3.5.

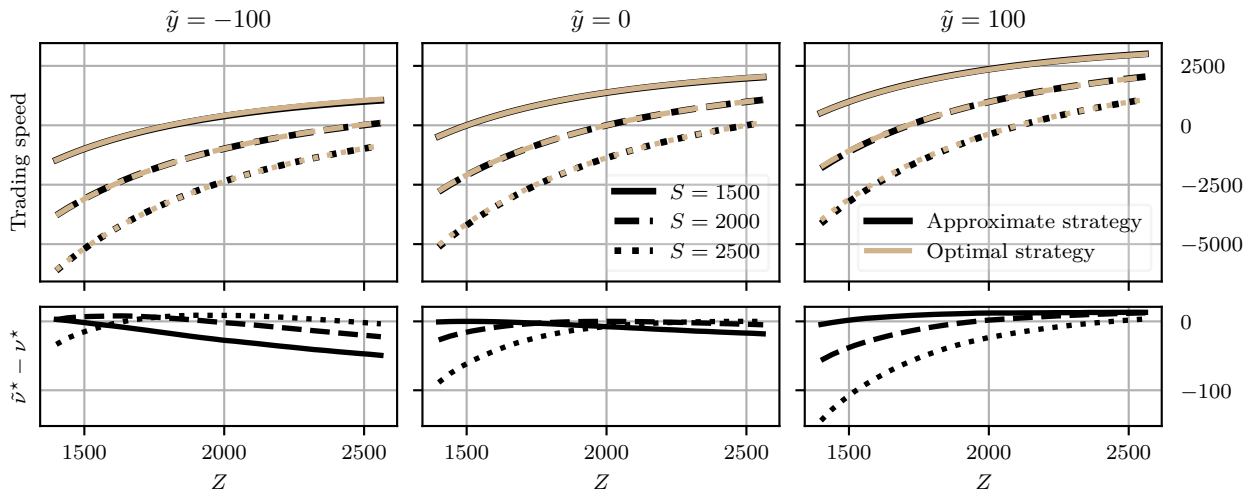


Figure 3.8: Comparison of the optimal speed (A.11) obtained with a numerical scheme and the closed-form approximation strategy in (3.27). Model parameters: $T = 0.1$, $\sigma = 0.03$, $\gamma = 0.02$, $\beta = 1$, $\alpha = 5$, $\kappa = 10^7$, $\eta = 1$, $\phi = 10^{-5}$, and $Z_0 = S_0 = 2000$. Inventory is $\tilde{y} = -100$ (left panel), $\tilde{y} = 0$ (middle panel), and $\tilde{y} = 100$ (right panel).

3.4 Model II: optimal execution without oracle rate

In this section, we consider the problem of an LT who wants to exchange a large position in asset Y into asset X in a CPMM. The key differences between Model II and Model I of Section 3.3 are that in Model II the AMM rates are efficient, so the LT does not use an oracle rate from another venue, and the depth of the pool is stochastic.

The LT must liquidate a large position in asset Y in a CPMM over a period of time $[0, T]$, where $T > 0$. Asset Y and the wealth of the LT are valued in terms of asset X. The CPMM is very active and with concentrated liquidity, so the depth κ exhibits very frequent and random updates. The instantaneous rate $(Z_t)_{t\in[0,T]}$ evolves as

$$dZ_t = \gamma\, Z_t\, dB_t \ ,$$

and the dynamics of the depth process $(\kappa_t)_{t\in[0,T]}$ are given by

$$d\kappa_t = \varsigma\, \kappa_t\, dL_t \ .$$

Here, $(B_t)_{t\in[0,T]}$ and $(L_t)_{t\in[0,T]}$ are independent standard Brownian motions and ς is the volatility of the depth κ. The wealth process $(\tilde{x}_t)_{t\in[0,T]}$ satisfies

$$d\tilde{x}_t = \left(Z_t - \eta\, \frac{Z_t^{3/2}}{\kappa_t}\, \nu_t \right) \nu_t\, dt\, .$$

3.4.1 Performance criterion and value function

For each $(t, \tilde{x}, \tilde{y}, Z, \kappa) \in [0, T] \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++}$, and for each admissible control $\nu \in \mathcal{A}$ the performance criterion of the LT is given by

$$u^\nu(t, \tilde{x}, \tilde{y}, Z, \kappa) = \mathbb{E}_{t,\tilde{x},\tilde{y},Z,\kappa} \left[\tilde{x}_T^\nu + \tilde{y}_T^\nu\, Z_T - \alpha\, (\tilde{y}_T^\nu)^2 - \phi \int_t^T (\tilde{y}_s^\nu)^2\, ds \right] \ , \tag{3.28}$$

and the value function is

$$u(t, \tilde{x}, \tilde{y}, Z, \kappa) = \sup_{\nu\in\mathcal{A}} u^\nu(t, \tilde{x}, \tilde{y}, Z, \kappa) \ . \tag{3.29}$$

3.4.2 The dynamic programming equation

The value function (3.29) is the unique classical solution to the HJB equation

$$\begin{aligned} 0 = {} & \partial_t w - \phi\, \tilde{y}^2 + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ} w + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa} w \\ & + \sup_{\nu\in\mathbb{R}} \left(\left(\nu\, Z - \eta\, \frac{Z^{3/2}}{\kappa}\, \nu^2 \right) \partial_{\tilde{x}} w - \nu\, \partial_{\tilde{y}} w \right) \ , \end{aligned} \tag{3.30}$$

with terminal condition

$$w(T, \tilde{x}, \tilde{y}, Z, \kappa) = \tilde{x} + \tilde{y}\, Z - \alpha\, \tilde{y}^2 \ . \tag{3.31}$$

The terminal condition (3.31) suggests the ansatz

$$w(t, \tilde{x}, \tilde{y}, Z, \kappa) = \tilde{x} + \tilde{y}\, Z + \theta(t, \tilde{y}, Z, \kappa)\, ,$$

which we justify by the following proposition, for which a proof is straightforward.

Proposition 3. *Assume there exists a function* $\theta \in C^{1,1,2,2}([0,T] \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++})$ *that solves*

$$0 = \partial_t \theta - \phi\, \tilde{y}^2 + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ}\theta + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa}\theta + \sup_{\nu \in \mathbb{R}} \left(-\eta\, \frac{Z^{3/2}}{\kappa}\, \nu^2 - \nu\, \partial_{\tilde{y}}\theta \right) , \tag{3.32}$$

on $[0,T) \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++}$ *, with terminal condition*

$$\theta(T, \tilde{y}, Z, \kappa) = -\alpha\, \tilde{y}^2 \ . \tag{3.33}$$

Then, the function $w\colon [0,T] \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \to \mathbb{R}$ *defined by*

$$w(t, \tilde{x}, \tilde{y}, Z, \kappa) = \tilde{x} + \tilde{y}\, Z + \theta(t, \tilde{y}, Z, \kappa) \tag{3.34}$$

is a solution to (3.30) *on* $[0,T) \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++}$ *with terminal condition* (3.31) *.*

Next, solve the first order condition in (3.32) to obtain the LT's trading speed in feedback form

$$\nu^\star = -\frac{\kappa}{2\,\eta}\, \partial_{\tilde{y}}\theta\, Z^{-3/2} \, . \tag{3.35}$$

Substitute (3.35) into (3.32) to write

$$0 = \partial_t \theta - \phi\, \tilde{y}^2 + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ}\theta + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa}\theta + \frac{\kappa}{4\,\eta}\, (\partial_{\tilde{y}}\theta)^2\ Z^{-3/2} \, . \tag{3.36}$$

Finally, simplify (3.36) with the ansatz

$$\theta(t, \tilde{y}, Z, \kappa) = \theta_0(t, Z, \kappa) + \theta_1(t, Z, \kappa)\, \tilde{y} + \theta_2(t, Z, \kappa)\, \tilde{y}^2 \, , \tag{3.37}$$

which is justified by the following proposition, for which a proof is straightforward.

Proposition 4. *Assume there exist functions* $\theta_0 \in C^{1,2,2}([0,T] \times \mathbb{R}_{++} \times \mathbb{R}_{++})$*,* $\theta_1 \in C^{1,2,2}([0,T] \times \mathbb{R}_{++} \times \mathbb{R}_{++})$*, and* $\theta_2 \in C^{1,2,2}([0,T] \times \mathbb{R}_{++} \times \mathbb{R}_{++})$ *which solve the system of PDEs*

$$\begin{cases} 0 = \partial_t \theta_2 - \phi + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ}\theta_2 + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa}\theta_2 + \frac{\kappa}{\eta}\, \theta_2^2\, Z^{-3/2} \, , \\ 0 = \partial_t \theta_1 + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ}\theta_1 + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa}\theta_1 + \frac{\kappa}{\eta}\, \theta_1\, \theta_2\, Z^{-3/2} \, , \\ 0 = \partial_t \theta_0 + \frac{1}{2}\, \gamma^2\, Z^2\, \partial_{ZZ}\theta_0 + \frac{1}{2}\, \varsigma^2\, \kappa^2\, \partial_{\kappa\kappa}\theta_0 + \frac{\kappa}{4\,\eta}\, \theta_1^2\, Z^{-3/2} \, , \end{cases} \tag{3.38}$$

on $[0,T) \times \mathbb{R}_{++} \times \mathbb{R}_{++}$ *with terminal conditions*

$$\theta_2(T, Z, \kappa) = -\alpha \, , \quad \theta_1(T, Z, \kappa) = 0 \, , \quad \text{and} \quad \theta_0(T, Z, \kappa) = 0 \, . \tag{3.39}$$

Then, the function $\theta\colon [0,T] \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \to \mathbb{R}$ *defined by*

$$\theta(t, \tilde{y}, Z, \kappa) = \theta_0(t, Z, \kappa) + \theta_1(t, Z, \kappa)\, \tilde{y} + \theta_2(t, Z, \kappa)\, \tilde{y}^2 \, ,$$

solves (3.36) *over* $[0,T) \times \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++}$ *with terminal condition* (3.33) *.*

The optimal strategy in feedback form (3.35) is now given by

$$\nu^\star = -\frac{\kappa}{2\,\eta}\,\left(2\,\theta_2\,\tilde{y} + \theta_1\right)\,Z^{-3/2}\,. \tag{3.40}$$

The system of PDEs in (3.38) can be solved sequentially as follows. Solve the first PDE in the system to obtain θ_2. Substitute θ_2 is the second and third equations of the system so the PDEs in θ_1 and θ_0 become linear. We cannot solve the semilinear PDE in θ_2 in closed-form, and providing an existence result is out of the scope of this work. However, Theorem 2 provides *a priori* estimates for θ_2 and suggests the system of PDEs is well-behaved. The proof is omitted as it uses the same arguments as those in the proof of Theorem 5.

Theorem 2. *Let $\theta_0 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$, $\theta_1 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$, and $\theta_2 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$ be a solution to the system of PDEs (3.38) with terminal condition (3.39). Define θ in (3.37) and define w in (3.34). Assume that for all $(t,\tilde{x},\tilde{y},Z,\kappa)\in[0,T]\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_{++}\times\mathbb{R}_{++}$ we have*

$$u^\nu(t,\tilde{x},\tilde{y},Z,\kappa) \le w\,(t,\tilde{x},\tilde{y},Z,\kappa)\,,$$

where u^ν is defined in (3.28), and that equality is obtained for the optimal control $(\nu^\star)_{t\in[0,T]}$ in feedback form in (3.40). Then θ_2 has the following bounds

$$-\alpha - \phi\,(T-t) \le \theta_2\,(t,\tilde{x},\tilde{y},Z,\kappa) \le 0, \quad \forall\,(t,\tilde{x},\tilde{y},Z,\kappa)\in[0,T]\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_{++}\times\mathbb{R}_{++}\,.$$

3.5 Performance of strategies

We study the performance of two versions of the closed-form approximation strategy in (3.26) and (3.27) of Model I; see Section 3.3. One version focuses on liquidating a large position in one asset and the other uses the lead-follow relationship between the oracle and AMM rates to execute a statistical arbitrage. We use the Uniswap v3 data of Section 3.2 for the liquid pool ETH/USDC and the illiquid pool ETH/DAI. We account for AMM and gas fees and assume that the orders sent by the LT do not impact the dynamics of the pools.

We use in-sample data to estimate model parameters and use out-of-sample data to execute the strategies. For in-sample data, we use a window of 24 hours prior to the start of the trading programme for both pools. For out-of-sample data, we use windows of 2 and 12 hours when the LT trades in the liquid and illiquid pools, respectively. To measure performance, we use rolling time windows for estimation and execution, between 1 July 2021 and 5 May 2022, to carry out this procedure. Specifically, after every execution programme, we shift both windows by 2 and 12 hours for the liquid and illiquid pools, respectively, and repeat the same procedure, i.e., estimate parameters with in-sample data and trade with out-of-sample data. We remark that we do not simulate rates, we use those of the AMM and Binance, and execution costs are those the trades would have received. In total, we run 3,635 and 607 execution programmes for ETH/USDC and ETH/DAI, respectively.

We proceed as follows. Subsection 3.5.1 describes how parameter estimates are obtained and showcases the performance of the liquidation strategy. Subsection 3.5.2 discusses the use of model I for statistical arbitrage, and showcases the performance.

3.5.1 Liquidation strategy

We describe how to estimate the in-sample model parameters for every run of the liquidation strategy. For rate dynamics, the LT performs OLS regressions on the discretised versions of (3.8) and (3.12):

$$\begin{aligned} \Delta \log S_t &= -\frac{\sigma^2}{2}\,\Delta t + \sigma\,\sqrt{\Delta t}\,\upsilon_t\,, \\ \Delta \log Z_t &= -\frac{\gamma^2}{2}\,\Delta t + \beta\left(\frac{S_t - Z_t}{Z_t}\right)\Delta t + \gamma\,\sqrt{\Delta t}\,\varepsilon_t\,, \end{aligned} \tag{3.41}$$

where $\{\varepsilon_t\,, \upsilon_t\}$ are error terms. Here, the size of the time-step Δt is the frequency of the liquidity taking orders (from all LTs) that arrive in the pool during the estimation period.

For the liquidation strategy, we target a participation rate of 50% of the observed hourly volume to set the initial inventory, which is liquidated by the LT over the trading window at the same frequency as the observed average trading frequency over the in-sample estimation period. The LT's trading frequency determines the value of the parameter η in (3.7), which scales the execution costs.

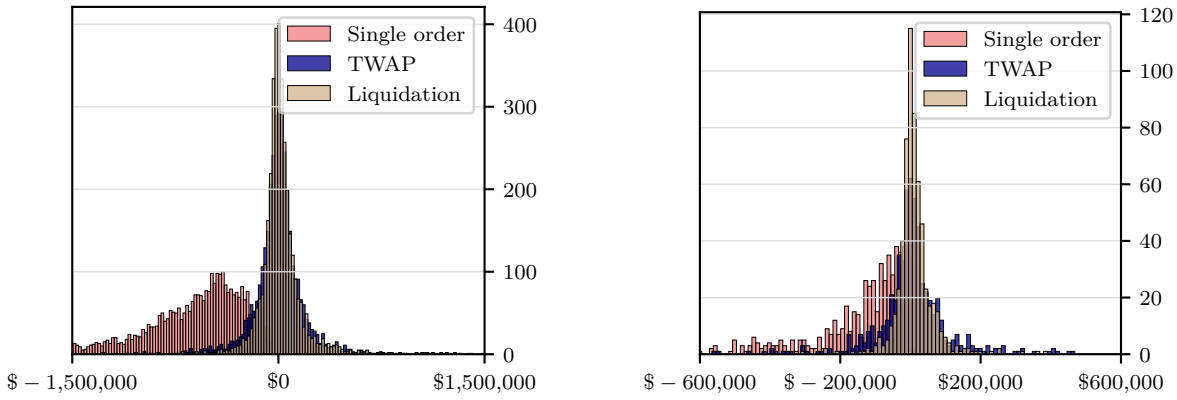
The value of the other model parameters are as follows. The value of the parameter κ in the trading speed (3.26) is the last observed depth of the pool before the start of the execution. The value of the running inventory parameter ϕ is kept constant for all runs. The value of the terminal penalty parameter α is arbitrarily large to enforce full liquidation of outstanding inventory by the end of the trading horizon. Table 3.1 shows the parameter values we use for all strategy runs. Finally, as a more detailed example, A.2 describes parameter estimation and performance for a specific run of the liquidation strategy.

	ETH/USDC	ETH/DAI
T	0.083 days	0.5 days
ϕ	0.005 USDC $\cdot$ ETH^{-2}	0.05 DAI $\cdot$ ETH^{-2}
α	10 USDC $\cdot$ ETH^{-2}	10 DAI $\cdot$ ETH^{-2}

Table 3.1: Values of the liquidation model parameters.

We benchmark the performance of the liquidation strategy with two strategies: TWAP, which consists in trading at a constant rate; and a single order execution strategy, which consists in executing the entire order at the beginning of the execution window. The market rates at the time of trading are used to compute the execution costs for all strategies. Gas fees are 5 USD per transaction, regardless of transaction size. On the other hand, AMM fees depend on transaction size, and here we impute a 0.01 % fee to the value of every transaction. Figure 3.9 depicts the distribution of the gross PnL for each execution programme, which is given by $\tilde{x}_T + \tilde{y}_T\,Z_T - \tilde{y}_0\,Z_0$. Tables 3.2 and 3.3 show the average and standard deviation of the gross PnLs, the number of transactions, the gas fees, and the AMM fees.[10]

[10] Gross PnL, as opposed to net PnL, is computed without the AMM fees and gas fees paid by the LT.



(a) Distribution of gross PnL for all strategies for ETH/USDC (3,635 executions).

(b) Distribution of gross PnL of strategies for ETH/DAI (607 executions).

Figure 3.9: PnL distribution.

	Gross avg. PnL	Std. dev.	Avg. num. trades	Avg. fees
Single order	−956,298	1,963,014	1	2,538
TWAP	3998	217,001	439	10,529
Liquidation	27,185	288,518	439	11,885

Table 3.2: Performance and fees for ETH/USDC (3,635 executions). The Average PnL does not include fees

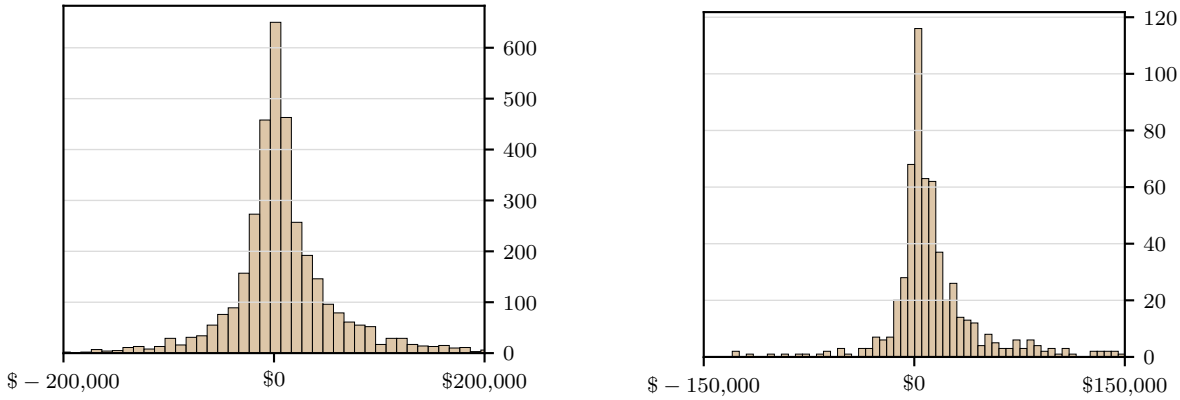
	Gross avg. PnL	Std. dev.	Avg. num. trades	Avg. fees
Single order	−233,390	428,688	1	634
TWAP	1,875	170,008	108	1,217
Liquidation	12,240	63,605	108	1,782

Table 3.3: Performance and fees for ETH/DAI (607 executions). The Average PnL does not include fees

Figure 3.9 and Tables 3.2 and 3.3 show that liquidating all the inventory in one trade is sub-optimal compared with the other strategies due to the high execution costs of the large order. In both cases, our model outperforms TWAP in terms of the ratio between performance, net of fees, and risk measured by the standard deviation. Key to the outperformance is that the liquidation strategy uses the rates in Binance as a trading signal.

3.5.2 Speculative strategy

We consider the same setup as before, i.e., the in-sample estimation and out-of-sample execution. Here, the LT arbitrages the AMM. To this end, the LT starts with zero inventory in Y and sets the values of ϕ to 0.001 USDC·ETH^{-2} and 0.01 DAI·ETH^{-2} for the liquid and illiquid pools, respectively. The strategy profits from the oracle rate as a predictive signal. Figure 3.10 depicts the distribution of the gross PnL for each run and Tables 3.4 and 3.5 show the average and standard deviation of the gross PnLs, the number of transactions, and the estimated AMM and gas fees. Note that the speculative strategy is more profitable in the illiquid pool due to a larger discrepancies between the oracle and the instantaneous pool rate, leading to more arbitrage opportunities.



(a) Distribution of gross PnL for ETH/USDC.

(b) Distribution of gross PnL for ETH/DAI.

Figure 3.10: Statistical arbitrage PnL distribution.

	Gross avg. PnL	Std. dev.	Avg. num. trades	Avg. fees
Speculative strategy	22,693	190,789	439	7,111

Table 3.4: Performance and fees for the pair ETH/USDC (3,635 executions). The Average PnL does not include fees

	Gross avg. PnL	Std. dev.	Avg. num. trades	Avg. fees
Speculative strategy	20,886	54,043	108	2,082

Table 3.5: Performance and fees for the pair ETH/DAI (607 executions). The Average PnL does not include fees

3.6 Conclusions

In this chapter, we used Uniswap v3 data to analyse rate, liquidity, and execution costs of CPMMs. We proposed two models for optimal trading in CPMMs. In the first model, we assumed that prices are formed in an alternative venue and the liquidity provided in the CPMM remains constant for relevant periods of time. In the second model, price formation takes place in the pool and the liquidity provided in the pool is stochastic. Finally, we used in-sample estimation of model parameters and out-of-sample market data to test the performance of the strategies, so our results do not rely on simulations. We show our strategy considerably outperforms TWAP and a strategy that consists in sending a single large order. We also showed that there are significant arbitrage opportunities between Binance and AMM rates.

Chapter 4

Optimal Trading with Signals in Multiple Automated Market Makers

4.1 Introduction

Automated market makers (AMMs) are open-source immutable programs that run on peer-to-peer networks. These programs encode the trading rules that govern how liquidity takers (LTs) and liquidity providers (LPs) interact. The emergence of these trading venues poses great challenges to traditional financial services because they do not rely on traditional intermediaries and stakeholders such as banks and brokers. At present, the most popular type of AMM is the constant product market maker (CPMM) such as Uniswap v3, and AMMs serve mainly as trading venues for crypto-currencies. CPMMs are a special type of constant function market maker (CFMM), where a convex trading function and a set of rules determine how the exchange operates.

The literature on the optimal design of these venues and the mathematical tools to optimally take and provide liquidity are new. In Cartea et al. (2022b), the authors propose models for LTs that trade in a CPMM when prices form in the pool and when the prices form in an alternative venue. In Cartea et al. (2023b), the authors characterise the wealth of LPs in CFMMs and in CPMMs with concentrated liquidity, and provide dynamic LP strategies. The work of Bergault et al. (2023) uses traditional market making models to propose a new design of AMMs. In Fan et al. (2021) and Fan et al. (2022) the authors show that LPs can strategically exploit their beliefs on future exchange rates, and study the implications on smart contract design. Finally, the work in Angeris et al. (2022a,b) uses convex optimisation for routing and multi-asset trading in CFMMs.

In practice, LTs usually trade in multiple assets simultaneously. At present, it is key for LTs that trade in CPMMs to consider exchange rates from more liquid alternative venues where prices are formed, and to take into account the significant co-movements that crypto-currencies exhibit. In this work, we solve the problem of an LT who wishes to execute large orders in multiple assets. She has access to different CPMM pools and uses information from exchange rates from alternative trading venues (oracle). Moreover, we assume that the LT uses a set of predictive signals to improve the performance of his strategy; see Forde et al. (2022) and Cartea et al. (2018a) for the use of signals in optimal trading.

We formulate the optimal trading problem as a stochastic control problem where the LT controls the speed at which she trades in the different pools. Key to the performance of the trading strategy, is to balance price risk and execution costs. In CPMMs, the execution costs are approximated with the convexity of the trading function, which is a function of the pool's depth and the pool's exchange rate; see Cartea et al. (2022b). In our model, the joint dynamics of the CPMM rates, the oracle rates, an a set of predictive signals follow a multi-dimensional Ornstein-Uhlenbeck process (multi-OU). Multi-OU dynamics are well suited to capture (i) the common stochastic trends that crypto-currencies usually share, (ii) the lead-lag effects between oracle and pool rates, and (iii) to incorporate predictive signals.

The remainder of this chapter is organised as follows. Section 4.2 recalls how LTs and LPs interact with a CPMM. Section 4.3 uses Uniswap v3 data to study the cointegrated dynamics of oracle and pool rates for ETH and BTC. Section 4.4 introduces the optimal trading model and derives a closed-form approximation strategy. Section 4.5 implements the model of Section 4.4 using oracle rates from Binance and pool rates from Uniswap v3 for an agent in charge of a portfolio composed of the crypto-currencies ETH and BTC. In particular, we show that our strategy uses the cointegration factors in the joint dynamics of the exchange rates, and that it outperforms a classical Almgren-Chriss strategy without cointegration and a TWAP strategy.

4.2 Constant function market makers

In this section, we discuss how CFMMs operate and characterise their execution costs and the market impact of liquidity taking activity. We recall the mathematical properties of the trading function and how market participants interact with CFMM pools in charge of a pair of assets.

In a CFMM, LPs and LTs interact through liquidity pooling. LPs deposit assets in a common pool, and LTs trade directly with the pool. In the remainder of this chapter, we consider n assets $\boldsymbol{Y} = \left(Y^1, \ldots, Y^n\right)^\intercal$ that are valued in terms of a reference asset X. Moreover, we consider a CFMM that offers a pool for every pair $\left(X, Y^i\right)$, and we refer to the instantaneous exchange rate of Y^i in terms of X in the pool by Z^i.

The liquidity pool for a pair $\left(X, Y^i\right)$ consists of a quantity $x > 0$ of asset X and a quantity $y > 0$ of asset Y^i. The CFMM pool is characterized by a trading function $f(x, y)$, which is differentiable, has convex level sets, and is increasing in both x and y. A CFMM pool imposes an LT trading condition and an LP trading condition both of which define the state of the pool after an LT transaction and after an LP operation is completed.

LT trading condition Let (x, y) be the state of the pool before the arrival of an LT buy order for a quantity $\Delta y > 0$ of asset Y^i. The quantity $\Delta x > 0$ of asset X that the LT pays to the pool is determined by the LT trading condition $f(x, y) = f(x + \Delta x, y - \Delta y) = \kappa^2$, where $\kappa > 0$ is the *depth* of the pool. For a sell order, the LT trading condition is $f(x, y) = f(x - \Delta x, y + \Delta y) = \kappa^2$. As in Cartea et al. (2022b,c), we define the *level function* φ as $x = \varphi(y)$, which is differentiable,

convex, and decreasing.[1] Moreover, the instantaneous exchange rate is given by

$$Z = -\varphi'(y) \ . \tag{4.1}$$

In Section 3.2, we show that one can use a Taylor expansion to approximate the execution costs for trading a quantity Δy with the convexity of φ, so that

$$\left| Z - \tilde{Z}(\Delta y) \right| \approx \frac{1}{2} \varphi''(y) \, |\Delta y| = \frac{1}{\kappa} Z^{3/2} |\Delta y| \, ,$$

where $\tilde{Z}(\Delta y)$ is the execution rate. The level function φ is decreasing and convex, thus the execution rate received by an LT deteriorates (i.e., execution costs increase) as the size of the trade increases, and as the depth of the pool κ decreases.

Finally, following an LT trade of size Δy, the *price impact* is the difference between the two rates, i.e., $-\varphi'(y + \Delta y) + \varphi'(y)$ or $-\varphi'(y - \Delta y) + \varphi'(y)$. Using the definition in (4.1) and a Taylor's expansion we write

$$\text{Price impact} \approx \varphi''(y) \, |\Delta y| = \frac{2 \, Z^{3/2}}{\kappa} |\Delta y| \, .$$

LP trading condition LPs interact with the pool by depositing assets in the pool or withdrawing assets from the pool. The LP trading condition requires that LP operations do not impact the rate Z. However, LP operations do change the depth κ of the pool. More precisely, let (x, y) be the state of the pool before an LP deposits liquidity $(\Delta x, \Delta y)$,[2] so $(x + \Delta x, y + \Delta y)$ is the state of the pool after the deposit, which increases the depth of the pool, because

$$f(x + \Delta x, y + \Delta y) = \overline{\kappa}^2 > f(x, y) = \kappa^2 \, ;$$

recall that the trading function is increasing.

Next, let φ_κ be the level function corresponding to the depth κ and let $\varphi_{\overline{\kappa}}$ be the level function corresponding to the depth $\overline{\kappa}$. The LP trading condition requires that when an LP deposits the quantities (x, y) such that $f(x + \Delta x, y + \Delta y) = \overline{\kappa}^2$, then the following equality must hold:

$$Z = -\varphi'_\kappa(y) = -\varphi'_{\overline{\kappa}}(y + \Delta y) \ .$$

To enforce this property, we require the trading function $f(x, y)$ to be homothetic. In practice, LPs usually invest an initial wealth V in terms of the reference asset X, so the amounts Δx and Δy are obtained from the LP trading condition and the equality $V = \Delta x + Z \, \Delta y$. In contrast, LTs usually specify a quantity Δx to trade, and receive or pay a quantity Δy which is given by the LT trading condition.

[1] To ensure that the pool cannot be depleted by LT trades, the level function should also satisfy $\lim_{y \to 0} \varphi(y) = +\infty$ and $\lim_{y \to +\infty} \varphi(y) = 0$.

[2] Here, $\Delta x \geq 0$ and $\Delta y \geq 0$.

Pair	Volume Uniswap	Volume Binance	Depth Uniswap
ETH/USDC	$\$\,129.8 \times 10^9$ (1 txn per 13 sec)	$\$\,267.44 \times 10^9$ (1 txn per 0.13 sec)	2.04×10^7
BTC/USDC	$\$\,5.1 \times 10^9$ (1 txn per 381 sec)	$\$\,358.37 \times 10^9$ (1 txn per 1.49 sec)	1.32×10^6

Table 4.1: Volume, trading frequency, and depth in Uniswap v3 and Binance for the pairs ETH/USDC and BTC/USDC between 1 January 2022 and 30 June 2022.

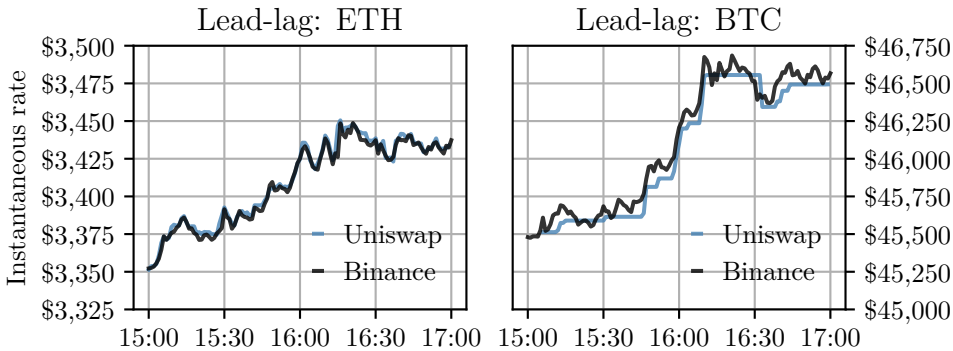


Figure 4.1: Exchange rates BTC/USDC and ETH/USDC in Uniswap v3 and Binance between 15:00 and 17:00 on 1 April 2022.

4.3 Data analysis

In this section, we use data from the Uniswap v3 pools ETH/USDC 0.05% and BTC/USDC 0.3% , and from the LOB-based market Binance for the same pairs, to study the joint dynamics of crypto-currencies in both markets between 1 January 2022 and 30 June 2022. Uniswap v3 is a CPMM with concentrated liquidity running on the Ethereum blockchain, which is a type of distributed ledger technology that validates and stores transactions. At present, it is the most liquid CPMM. In CPMMs, the trading function is $f(x, y) = x \times y$, so the level function is $\varphi(y) = \kappa^2/y$ and the instantaneous exchange rate is $Z = -\varphi'(y) = \kappa^2/y^2 = x/y$. Binance is the most liquid LOB-based exchange for crypto-currencies, and the venue where prices are formed. Table 4.1 shows descriptive statistics of the dataset.

4.3.1 Lead-lag relationships between trading venues

Figure 4.1 shows the dynamics of the pool and oracle exchange rates for both pairs throughout two hours of trading on 1 April 2023. The figure showcases the lead-lag relationship between both trading venues for a specific day, and suggests that the information from the oracle rates, i.e., those in Binance in our case, can be used to enhance the trading performance of an LT in the CPMM Uniswap v3.

The next subsections study the lead-lag relationships between trading venues by looking at cointegration, causality, and spillover effects.

	ETH/USDC Binance	ETH/USDC Uniswap v3	BTC/USDC Binance	BTC/USDC Uniswap v3
p-value	0.886	0.886	0.924	0.940
test statistic	–0.532	–0.532	–0.312	–0.185

Table 4.2: Augmented Dickey–Fuller test on each individual time series. The test concludes that each time series is not stationary and is integrated of order 1.

		ETH/USDC		BTC/USDC	
Null Hypothesis	Critical Value	Trace statistic	Result	Trace statistic	Result
$r \leq 0$	15.49	5,369.79	Rejected	4,923.37	Rejected
$r \leq 1$	3.84	0.29	Not Rejected	0.09	Not Rejected

Table 4.3: Results for two Johansen's cointegration Trace tests. The first test compares the ETH/USDC exchange rates in Uniswap v3 and Binance. The second test compares exchange rates for BTC/USDC in Uniswap v3 and Binance. Critical values are given for a significance level of 95%.

4.3.1.1 Cointegration

To further study the relationship between the two trading venues, we use the Johansen's cointegration Trace test to compare the ETH/USDC and BTC/USDC exchange rates in Uniswap v3 and Binance; see Johansen (1991).

To run the Johansen's Trace tests, we first verify the assumption that the time series are integrated of order one. Table 4.2 shows the result of the augmented Dickey–Fuller stationarity test for each time series; see Dickey and Fuller (1979). The test result show that all time series are not stationary and are integrated of order 1.

Table 4.3 shows the results of two Johansen's tests. The first test compares the ETH/USDC exchange rates in Uniswap v3 and Binance, and the second test compares those of BTC/USDC. Both tests reject the hypothesis of cointegration of rank $r = 0$, but fail to reject the hypothesis of cointegration of rank $r = 1$. Thus, the tests show that there exists common stochastic trends between Uniswap v3 and Binance for the pairs we consider.

4.3.1.2 Causality

Section 4.3.1.1 uses Johansen's test to study the presence of common stochastic trends driving the exchange rates in Binance and Uniswap v3. Here, we investigate the causality between the exchange rates in both venues.

Specifically, we run four Granger's causality tests; see Granger (1969). For each pair, we test whether historical rates in Binance can be used to forecast future rates in Uniswap v3, or vice versa. We perform Granger's test for $n = 1$ to $n = 5$ lags, where one lag corresponds to 5 minutes. Table 4.4 shows the test result. The test shows that for ETH/USDC, historical rates in Binance forecast

	ETH/USDC			
	Uniswap v3 $\not\rightarrow$ Binance		Binance $\not\rightarrow$ Uniswap v3	
Number of lags	p-value	Test statistic χ^2	p-value	Test statistic χ^2
$n = 1$	< 0.001	105.09	< 0.001	1,317.03
$n = 2$	< 0.001	182.78	< 0.001	1,870.48
$n = 3$	< 0.001	231.26	< 0.001	2,003.54
$n = 4$	< 0.001	232.55	< 0.001	2,123.15
$n = 5$	< 0.001	243.18	< 0.001	2,156.65

	BTC/USDC			
	Uniswap v3 $\not\rightarrow$ Binance		Binance $\not\rightarrow$ Uniswap v3	
Number of lags	p-value	Test statistic χ^2	p-value	Test statistic χ^2
$n = 1$	0.1663	1.92	< 0.001	5,381.31
$n = 2$	0.4766	1.48	< 0.001	6,479.78
$n = 3$	< 0.001	18.59	< 0.001	6,656.85
$n = 4$	< 0.001	23.15	< 0.001	6,883.71
$n = 5$	< 0.001	26.97	< 0.001	6,963.54

Table 4.4: Granger's causality test results. **Top panel:** The test shows that for ETH/USDC, historical rates in Binance can be used to forecast rates in Uniswap v3 and vice versa. **Bottom panel:** The test shows that for BTC/USDC, historical rates in Binance can be used to forecast rates in Uniswap v3, but rates in Uniswap v3 cannot be used to forecast rates in Binance after 5 and 10 minutes.

rates in Uniswap v3 and vice versa. For BTC/USDC, the test shows that historical rates in Binance forecast rates in Uniswap v3, but rates in Uniswap v3 do not forecast rates in Binance after 5 and 10 minutes.

The results of the Granger's causality tests suggest that for ETH/USDC prices in Uniswap v3 and Binance are aligned. On the other hand, the exchange rates in Uniswap v3 for BTC/USDC are delayed with respect to the exchange rates in Binance.

4.3.1.3 Spillover effects

Next, we study spillover effects between Uniswap v3 and Binance. Spillover effects are the consequences that shocks in a random variable have on other variables. Here, we look at the spillover indices introduced in Diebold and Yilmaz (2012) and we refer to Giudici and Pagnottoni (2020) for a detailed analysis of spillover effects in BTC/USDC between various centralised trading venues.

First, we recall the definition of generalised impulse response function introduced in Koop et al. (1996). For each pair, let S be the oracle from Binance and let Z be the exchange rate in Uniswap v3. We represent the time series of the exchange rates returns using a VAR(k) model

$$\begin{pmatrix} \Delta Z_t \\ \Delta S_t \end{pmatrix} = \sum_{i=1}^{k} \mathbf{\Phi}_i \begin{pmatrix} \Delta Z_{t-i} \\ \Delta S_{t-i} \end{pmatrix} + \varepsilon_t \,, \tag{4.2}$$

where $\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_k$ are 2×2 regression matrices and $\boldsymbol{\varepsilon}_t$ is the error term, which is distributed as $\mathcal{N}(0, \boldsymbol{\Sigma})$, where

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{ZZ} & \sigma_{ZS} \\ \sigma_{ZS} & \sigma_{SS} \end{pmatrix}$$

is the covariance matrix. Next, we write the moving average representation of (4.2) as

$$\begin{pmatrix} \Delta Z_t \\ \Delta S_t \end{pmatrix} = \sum_{k=1}^{\infty} \boldsymbol{A}_k \, \boldsymbol{\varepsilon}_{t-k} \,,$$

where the coefficient matrices $\boldsymbol{A}_k$ satisfy

$$\boldsymbol{A}_k = \begin{cases} 0 & \text{for } k < 0 \,, \\ I_2 & \text{for } k = 0 \,, \\ \displaystyle\sum_{l=1}^{k} \boldsymbol{\Phi}_l \, \boldsymbol{A}_{k-l} & \text{for } k > 0 \,. \end{cases}$$

The generalised impulse response function of $(\Delta Z_t, \Delta S_t)$ at horizon n is defined as

$$GI_n(\boldsymbol{\delta}) = \mathbb{E}_{t-1} \left[\begin{matrix} \Delta Z_{t+n} \\ \Delta S_{t+n} \end{matrix} \middle| \boldsymbol{\varepsilon}_t = \boldsymbol{\delta} \right] - \mathbb{E}_{t-1} \begin{bmatrix} \Delta Z_{t+n} \\ \Delta S_{t+n} \end{bmatrix} = \boldsymbol{A}_n \, \boldsymbol{\delta} \,, \tag{4.3}$$

and it represents the effect that a shock $\boldsymbol{\delta}$ at time t has on the expected future return at time $t+n$. Next, use $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ to write

$$\mathbb{E}\left[\boldsymbol{\varepsilon}_t \middle| \varepsilon_t^i = \delta_i \right] = \frac{1}{\sigma_{ii}^2} \, \boldsymbol{\Sigma} \, e_i \, \delta_i \,, \quad \text{for } i \in \{Z, S\} \,. \tag{4.4}$$

Next, set $\delta_i = \sigma_{ii}$ and use (4.3) and (4.4) to obtain, for each $n \in \mathbb{N}$, the scaled impulse response function

$$\psi_i^g(n) = \frac{1}{\sigma_{ii}} \boldsymbol{A}_n \, \boldsymbol{\Sigma} \, \boldsymbol{e}_i \,, \quad \text{for } i \in \{Z, S\} \,. \tag{4.5}$$

Here, for $i \in \{Z, S\}$, $\psi_i^g(n)$ in (4.5) measures the effect that one standard error shock in the returns of the trading venue i has on $(\Delta Z, \Delta S)$ after n time steps.

Next, we use (4.5) to derive, for each $n \in \mathbb{N}$, the generalised forecast error variance decomposition

$$\theta_{ij}^g(n) = \frac{1}{\sigma_{ii}^2} \times \frac{\sum_{l=0}^{n} (\boldsymbol{e}_i^\intercal \, \boldsymbol{A}_l \, \boldsymbol{\Sigma} \, \boldsymbol{e}_j)^2}{\sum_{l=0}^{n} (\boldsymbol{e}_i^\intercal \, \boldsymbol{A}_l \, \boldsymbol{\Sigma} \, \boldsymbol{A}_l^\intercal \, \boldsymbol{e}_i)^2} \,, \quad \text{for } i, j \in \{Z, S\} \,,$$

and we refer the reader to Lütkepohl (1991) and Pesaran and Shin (1998) for more details. Here, $\theta_{ij}^g(n)$ represents the proportion of the n-step-ahead forecast error variance of the returns in venue i which is accounted for by shocks in venue j. Finally, normalise the generalised forecast error variance decomposition to obtain

$$\tilde{\theta}_{ij}^g(n) = \frac{\theta_{ij}^g(n)}{\displaystyle\sum_{j \in \{Z,S\}} \theta_{ij}^g(n)} \,.$$

Next, define the total spillover index as

$$TSI(n) = \frac{\displaystyle\sum_{\substack{i,j \in \{Z,S\} \\ i \neq j}} \tilde{\theta}_{ij}^g(n)}{\displaystyle\sum_{i,j \in \{Z,S\}} \tilde{\theta}_{ij}^g(n)} \times 100 \,,$$

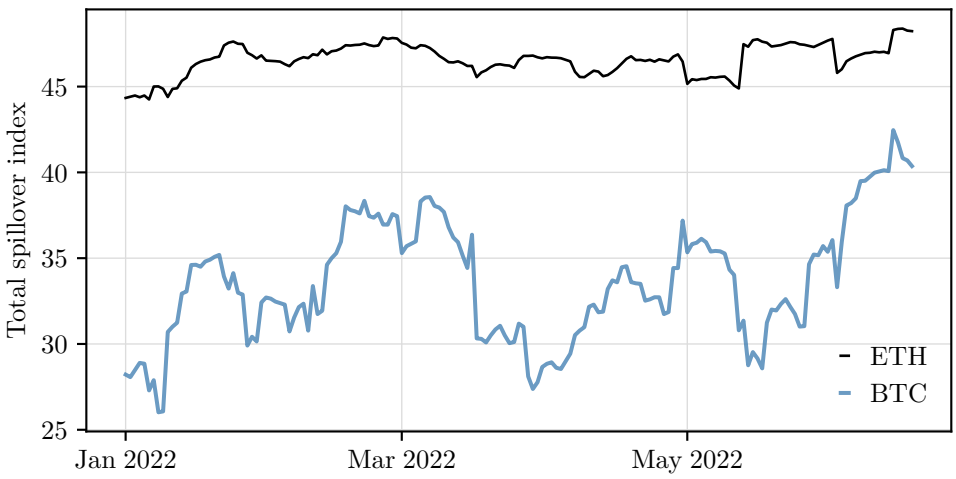


Figure 4.2: 10-step-ahead total spillover index between Uniswap v3 and Binance for ETH/USDC and BTC/USDC.

which represent the percentage of total n-step ahead forecast error variance which is caused by spillovers between trading venues. Then, for each $i \in \{Z, S\}$, define the directional spillover indices as

$$DSI_{i\to J}(n) = \frac{\sum_{\substack{j\in\{Z,S\}\\ i\neq j}} \tilde{\theta}^g_{ji}(n)}{\sum_{i,j\in\{Z,S\}} \tilde{\theta}^g_{ij}(n)} \times 100\,, \quad DSI_{J\to i}(n) = \frac{\sum_{\substack{j\in\{Z,S\}\\ i\neq j}} \tilde{\theta}^g_{ji}(n)}{\sum_{i,j\in\{Z,S\}} \tilde{\theta}^g_{ij}(n)} \times 100\,.$$

Finally, for each $i \in \{Z, S\}$, define the net spillover index as

$$NSI_i(n) = DSI_{i\to J}(n) - DSI_{J\to i}(n)\,.$$

The directional spillover index $DSI_{i\to J}(n)$ represents the percentage of total n-step ahead forecast error variance which is caused by spillovers from i to the other trading venues. On the other hand, the directional spillover index $DSI_{J\to i}(n)$ represents the percentage of total n-step ahead forecast error variance which is caused by spillovers from other trading venues to i.

For ETH/USDC and BTC/USDC, we compute the 10-step-ahead spillover indices between Uniswap v3 and Binance over a one-day period. Then, using rolling windows of two hours, we look at how spillover indices evolve. Figure 4.2 shows how the total spillover index between Uniswap v3 and Binance evolve over time for ETH/USDC and BTC/USDC. The figure shows that for the ETH/USDC pair, the total spillover index is higher than 45% for almost the entire time window considered. This suggests that both trading venues are equally important in determining the exchange rate. On the other hand, for the BTC/USDC pair, the total spillover index is for almost the whole time window below 40% and occasionally drops below 30%. This, indicates that prices are formed in Binance, with returns in Uniswap v3 adding only little information. This claim about the asymmetry in the role of the two trading venues is confirmed by Granger's causality test in Table 4.4.

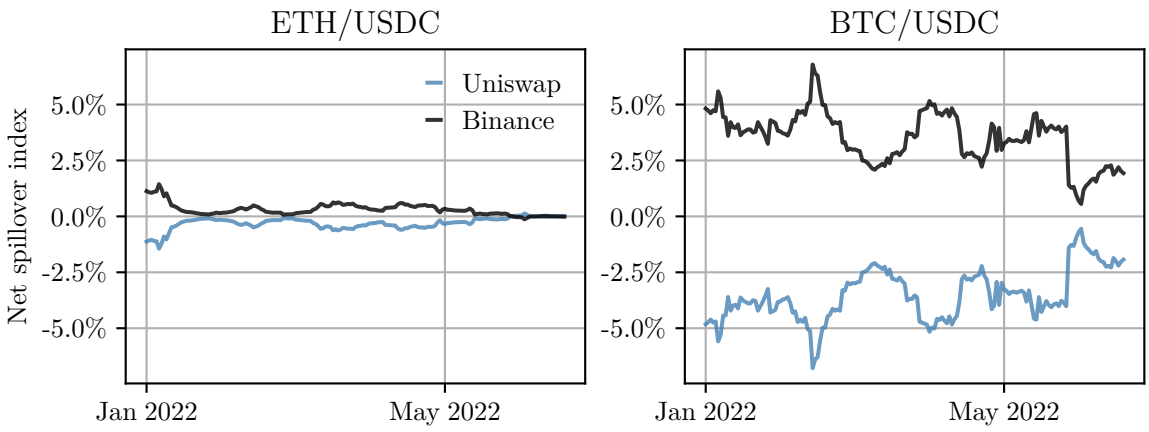


Figure 4.3: 10-step ahead net spillover index between Uniswap v3 and Binance for ETH/USDC and BTC/USDC.

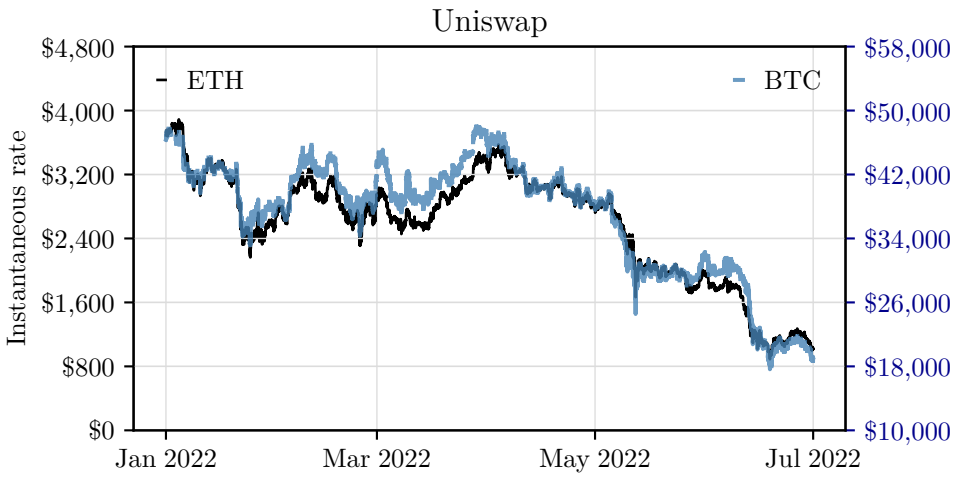


Figure 4.4: Exchange rates ETH/USDC and BTC/USDC in Uniswap v3 between January 2022 and June 2022.

Figure 4.3 shows the evolution of the net spillover indices between Uniswap v3 and Binance over time for ETH/USDC and BTC/USDC. The figure shows that, for both pairs, shocks in rates in Binance have larger impact than shocks in Uniswap v3. This shows that exchange rates in Binance have higher forecasting power than exchange rates in Uniswap v3. Thus, an LT who trades in Uniswap v3 can enhance the performance of her strategy by using the exchange rate in Binance as a predictive signal. This further motivates the model we introduced in Section 3.3, where we assume that price formation is in the alternative trading venue and not in the CPMM.

4.3.2 Cointegration and oracle rates

Figure 4.4 shows the rates for both BTC and ETH in terms of USDC in the pools we consider. The figure suggests that the exchange rates share common stochastic trends.

In our model, we capture the joint dynamics of the pool rates and the oracle rates using a multi-OU process. This process is suited to capture equilibrium relationships as those exhibited between BTC and ETH, and to capture lead-lag effects between different trading venues as those exhibited

Null Hypothesis	Trace statistics	Critical Value	Result
$r \leq 0$	$11,196.97$	54.68	Rejected
$r \leq 1$	$4,821.52$	35.46	Rejected
$r \leq 2$	11.45	19.93	Not rejected

Table 4.5: Johansen's cointegration Trace test results (critical values are given for a significance level of 99%). The test confirms a cointegration rank $r = 2$.

between Uniswap and Binance. The lead-lad effects are encoded in the cointegration factors in the mean-reversion matrix, which drives the deterministic and long-term component of the process.

To further study the suitability of the multi-OU process, we use the data described above to perform a Johansen's cointegration Trace test for the vector $\boldsymbol{R} = \left(Z^1, Z^2, S^1, S^2\right)$, where Z^1 and S^1 are the pool and oracle rates of ETH, and Z^2 and S^2 are the pool and oracle rates of BTC. Johansen's test rejects the hypothesis of no cointegration and the hypothesis of a cointegration rank $r = 1$, but does not reject a cointegration rank $r = 2$; see Johansen (1991).

The cointegration vectors are $(1, -0.022, -1.002, 0.022)$ and $(1, 0.09, -1.004, -0.088)$. The cointegration vectors, and hence the mean-reversion matrix (i.e., the matrix $\boldsymbol{\beta}$ in (4.6)), capture the joint dynamics of BTC and ETH, but also the lead-lag relation between the pool rates and oracle rates. Table 4.5 shows the results for the Johansen's cointegration Trace test.

Thus, it is key for an LT who trades multiple crypto-currencies to account for external information from similar assets and from other trading venues.

4.4 Optimal portfolio trading with oracles and signals

In this section, we solve the problem of an LT in charge of liquidating a large position in a basket of n risky and cointegrated assets $\boldsymbol{Y} = \left(Y^1, \ldots, Y^n\right)$ in a CPMM. The CPMM offers n pools for each asset against a reference asset X and we denote by $(\boldsymbol{Z}_t)_{t\in[0,T]} = \left(Z^1, \ldots, Z^n\right)_{t\in[0,T]}$ the exchange rates in these pools. The LT uses n oracle rates $(\boldsymbol{S}_t)_{t\in[0,T]} = \left(S^1, \ldots, S^n\right)_{t\in[0,T]}$ from another more liquid exchange and a set of m predictive signals $(\boldsymbol{I}_t)_{t\in[0,T]} = \left(I^1, \ldots, I^m\right)_{t\in[0,T]}$.

We fix a filtered probability space $\left(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{F} = (\mathcal{F}_t)_{t\in[0,T]}\right)$ satisfying the usual conditions, where $\mathbb{F}$ is the natural filtration generated by the collection of observable stochastic processes that we define below.

The LT must liquidate her initial inventory $\tilde{\boldsymbol{y}}_0 \in \mathbb{R}^n$ over a period of time $[0, T]$ and her wealth is valued in terms of asset X. The joint dynamics of the pool rates $\boldsymbol{Z}$, the oracle rates $\boldsymbol{S}$, and the signals $\boldsymbol{I}$ are modelled by a $2n + m$-dimensional vector $(\boldsymbol{R}_t)_{t\in[0,T]}$ that follows multi-OU dynamics

$$d\boldsymbol{R}_t = \boldsymbol{\beta}\left(\boldsymbol{\mu} - \boldsymbol{R}_t\right) dt + \boldsymbol{\sigma}^\intercal \, d\boldsymbol{W}_t \,, \tag{4.6}$$

where we force the first n components of $\boldsymbol{R}$ to coincide with $\boldsymbol{Z}$, $\boldsymbol{\beta}$ is a $2n + m \times 2n + m$ mean-reversion matrix, $\boldsymbol{\mu}$ is a $2n + m$-dimensional vector that models the long-term unconditional mean, $\boldsymbol{\sigma}^\intercal$ is the Cholesky decomposition of the asset prices correlation matrix $\boldsymbol{\Sigma}$, the symbol $^\intercal$ denotes

the transpose operator, and $(\boldsymbol{W}_t)_{t\in[0,T]}$ is a $2n+m$-dimensional Brownian motion with independent coordinates. For previous work with multi-OU price dynamics see Cartea et al. (2018b), Bergault et al. (2022), and Drissi (2022).

The LT trades at a continuous n-dimensional speed $(\boldsymbol{\nu}_t)_{t\in[0,T]}$ and the dynamics of her holdings are given by

$$d\tilde{\boldsymbol{y}}_t = -\boldsymbol{\nu}_t\, dt\,.$$

We do not restrict the speed to be positive; if for some $i \in \{1,\dots,n\}$ we have $\nu^i > 0$ the LT sells asset Y^i and if $\nu^i < 0$ the LT buys the asset.

As discussed in Section 3.2.2.3, the rate impact of the LT's activity is encoded in the trading function $f\,(x,y)$ and is a function of the trading speed as in traditional execution models (see Cartea et al. (2015)), the pool rate $\boldsymbol{Z}$, and the pool depth κ; see Cartea et al. (2022b).

More precisely, for each asset Y^i, we write the difference between the execution rate $\tilde{Z}^i$ and the instantaneous rate Z^i as

$$\tilde{Z}^i - Z^i = -\frac{\eta}{\kappa^i}\,\left(Z^i\right)^{3/2}\,\nu^i\,, \tag{4.7}$$

where κ^i is the depth of the pool that offers liquidity for the pair $\left(X, Y^i\right)$. Thus, the dynamics of the LT's holdings of asset X are given by

$$d\tilde{x}_t = \tilde{\boldsymbol{Z}}_t^{\intercal}\,\boldsymbol{\nu}_t\,dt = \left(\boldsymbol{Z}_t - \mathfrak{D}\left(\frac{\eta}{\boldsymbol{\kappa}} \odot \boldsymbol{Z}_t^{3/2}\right)\,\boldsymbol{\nu}_t\right)^{\intercal}\boldsymbol{\nu}_t\,dt\,, \tag{4.8}$$

where $\boldsymbol{\kappa}$ is the n-dimensional vector of the depth of the pools, η is a parameter which depends on the LT's trading frequency, $\mathfrak{D}(\boldsymbol{c})$ denotes a diagonal matrix whose diagonal elements are equal to the vector $\boldsymbol{c}$, and $\odot$ denotes the component-wise product. Compared with previous works in the literature that employ multi-OU price dynamics, here market impact parameter is a deterministic function of the stochastic vector $\boldsymbol{Z}$.

Let $\boldsymbol{\mathcal{X}}$ be a $n \times (2n+m)$ matrix with $\boldsymbol{\mathcal{X}}_{ij} = \mathbb{1}_{\{i=j\}}$ which maps the first n elements of a $2n+m$-dimensional vector into an n dimensional vector, and write (4.8) in terms of the vector $\boldsymbol{R}$ as

$$d\tilde{x}_t = \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}_t - \mathfrak{D}\left(\frac{\eta}{\boldsymbol{\kappa}} \odot \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}_t^{3/2}\right)\right)\,\boldsymbol{\nu}_t\right)^{\intercal}\boldsymbol{\nu}_t\,dt\,.$$

4.4.1 Performance criterion and value function

In our model, the LT maximises her expected terminal wealth in units of X while penalising inventory in the risky assets $\boldsymbol{Y}$. The set of admissible strategies is

$$\mathcal{A}_t = \left\{ (\boldsymbol{\nu}_s)_{s\in[t,T]} \;\middle|\; \mathbb{R}^n\text{-valued, } \mathbb{F}\text{-adapted, and } \int_t^T \|\boldsymbol{\nu}_s\|^2\,ds < +\infty, \quad \text{a.s.} \right\}, \tag{4.9}$$

and we write $\mathcal{A} := \mathcal{A}_0$. Let $\boldsymbol{\nu} \in \mathcal{A}$, the performance criterion of the LT who trades at speed $\boldsymbol{\nu}$, is the function $u^{\nu}\colon [0,T] \times \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{2n+m}_{++} \to \mathbb{R}$ given by

$$u^{\boldsymbol{\nu}}(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) = \mathbb{E}_{t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}}\left[\tilde{x}_T^{\boldsymbol{\nu}} + \boldsymbol{R}_T^{\intercal}\,\boldsymbol{\mathcal{X}}^{\intercal}\,\tilde{\boldsymbol{y}}_T^{\boldsymbol{\nu}} - (\tilde{\boldsymbol{y}}_T^{\boldsymbol{\nu}})^{\intercal}\,\boldsymbol{\alpha}\,\tilde{\boldsymbol{y}}_T^{\boldsymbol{\nu}} - \phi\int_t^T (\tilde{\boldsymbol{y}}_s^{\boldsymbol{\nu}})^{\intercal}\,\tilde{\boldsymbol{\Sigma}}\,\tilde{\boldsymbol{y}}_s^{\boldsymbol{\nu}}\,ds\right], \tag{4.10}$$

and the LT's value function is

$$u(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) = \sup_{\nu\in\mathcal{A}} u^{\nu}(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R})\,.$$

The first term on the right-hand side of (4.10) is the amount of asset X that the LT holds at the end of the trading window. The second term represents the terminal value of her remaining inventory marked-to-market using the pool rates $\boldsymbol{Z}$. The third term represents an inventory penalty where the diagonal positive matrix $\boldsymbol{\alpha}$ quantifies the aversion of the LT to holding non-zero inventory at time T; the units of $\boldsymbol{\alpha}$ are such that the penalty is in units of X. Finally, the last term is a running inventory penalty where $\tilde{\boldsymbol{\Sigma}}$ is the covariance matrix of the pool rates $\boldsymbol{Z}$, and the parameter $\phi \geq 0$ quantifies the urgency of the LT to liquidate inventory; the units of ϕ are such that the penalty is in units of X. Note that the third term on the right-hand side of (4.10) is sometimes interpreted in the algorithmic trading literature as the 'cost' of liquidating the final inventory $\tilde{\boldsymbol{y}}_T$ at time T. Using this interpretation, and in light of the results of the previous section, this term should be proportional to $\mathcal{D}\left(\frac{\eta}{\boldsymbol{\kappa}} \odot \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}_t^{3/2}\right)\right)$. Here we interpret this term as penalisation term and we observe that for $\boldsymbol{\alpha}$ sufficiently large, $(\tilde{\boldsymbol{y}}_T^{\boldsymbol{\nu}})^{\intercal}\,\boldsymbol{\alpha}\,\tilde{\boldsymbol{y}}_T^{\boldsymbol{\nu}}$ is greater than expected execution costs of terminal inventory.

The value function solves the Hamilton–Jacobi–Bellman (HJB) equation

$$\begin{aligned} 0 = {} & \partial_t w - \phi\,\tilde{\boldsymbol{y}}^{\intercal}\tilde{\boldsymbol{\Sigma}}\,\tilde{\boldsymbol{y}} + (\boldsymbol{\mu}-\boldsymbol{R})^{\intercal}\,\boldsymbol{\beta}\,\partial_{\boldsymbol{R}} w + \frac{1}{2}\,\mathrm{Tr}\left(\boldsymbol{\Sigma}\,D_{\boldsymbol{RR}} w\right) \\ & + \sup_{\nu\in\mathbb{R}^n}\left(-\boldsymbol{\nu}^{\intercal}\partial_{\tilde{\boldsymbol{y}}} w + \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R} - \mathfrak{D}\left(\frac{\eta}{\boldsymbol{\kappa}} \odot \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}^{3/2}\right)\right)\boldsymbol{\nu}\right)^{\intercal}\boldsymbol{\nu}\,\partial_{\tilde{x}} w\right)\,, \end{aligned} \tag{4.11}$$

with terminal condition

$$w(T,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) = \tilde{x} + \boldsymbol{R}^{\intercal}\,\boldsymbol{\mathcal{X}}^{\intercal}\,\tilde{\boldsymbol{y}} - \tilde{\boldsymbol{y}}^{\intercal}\,\boldsymbol{\alpha}\,\tilde{\boldsymbol{y}}\,.$$

Next, substitute the ansatz

$$w(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) \;=\; \tilde{x} + \boldsymbol{R}^{\intercal}\,\boldsymbol{\mathcal{X}}^{\intercal}\,\tilde{\boldsymbol{y}} + \theta(t,\tilde{\boldsymbol{y}},\boldsymbol{R})\;,$$

into (4.11) and solve the first order condition to obtain the partial differential equation (PDE)

$$\begin{aligned} 0 \;= {} & \partial_t\theta - \phi\,\tilde{\boldsymbol{y}}^{\intercal}\tilde{\boldsymbol{\Sigma}}\,\tilde{\boldsymbol{y}} + (\boldsymbol{\mu}-\boldsymbol{R})^{\intercal}\,\boldsymbol{\beta}\,(\boldsymbol{\mathcal{X}}^{\intercal}\tilde{\boldsymbol{y}} + \partial_{\boldsymbol{R}}\theta) \\ & + \frac{1}{2}\,\mathrm{Tr}\left(\boldsymbol{\Sigma}\,D_{\boldsymbol{RR}}\theta\right) + \partial_{\tilde{\boldsymbol{y}}}\theta^{\intercal}\,\mathfrak{D}\left(\frac{\boldsymbol{\kappa}}{4\,\eta} \odot \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}^{-3/2}\right)\right)\partial_{\tilde{\boldsymbol{y}}}\theta\,, \end{aligned} \tag{4.12}$$

with terminal condition

$$\theta(T,\tilde{\boldsymbol{y}},\boldsymbol{R}) = -\tilde{\boldsymbol{y}}^{\intercal}\,\boldsymbol{\alpha}\,\tilde{\boldsymbol{y}}\,,$$

where the optimal trading speed in feedback form is

$$\nu^{\star} = -\mathfrak{D}\left(\frac{\boldsymbol{\kappa}}{2\,\eta} \odot \left(\boldsymbol{\mathcal{X}}\,\boldsymbol{R}^{-3/2}\right)\right)\partial_{\tilde{\boldsymbol{y}}}\theta\;. \tag{4.13}$$

The semi-linear PDE (4.12) does not admit a closed-form solution. In practice, one uses a numerical scheme to approximate the solution θ and deduce the optimal speed (4.13). In our

model, the dimensionality of the problem grows exponentially with the number of assets. As a consequence, numerical schemes suffer from the curse of dimensionality.

In practice, the performance of an execution or statistical arbitrage strategy heavily relies on the ability to compute and send liquidity taking instructions with minimal latency. Thus, it is crucial for an LT to employ a trading strategy that is relatively fast to compute. Hence, we derive a closed-form approximation strategy for (4.13) which can be implemented and tracked by the LT in real time. Therefore, we extend the method developed in Section 3.3 for a single asset framework.

Subsection 4.4.2 solves an execution problem when convexity costs are fixed for a value of the pool rate Z, and Subsection 4.4.3 defines the closed-form approximation strategy as the uniform limit of a family of these strategies.

4.4.2 Optimal strategy with deterministic execution costs

Here, we derive a trading strategy that assumes a deterministic and fixed cost parameter in (4.7). More precisely, consider here that the execution cost in (4.7) is

$$\tilde{\boldsymbol{Z}} - \boldsymbol{Z} = -\eta\, \mathfrak{D}(\boldsymbol{\zeta})\, \boldsymbol{\nu}\,.$$

Here the n-dimensional vector $\boldsymbol{\zeta} > 0$ is a fixed cost parameter. Our aim is to derive the optimal strategy $\left(\boldsymbol{\nu}_t^{\star,\boldsymbol{\zeta}}\right)_{t\in[0,T]}$ for a given value of the parameter $\boldsymbol{\zeta} > 0$. For each value of $\boldsymbol{\zeta}$, we consider the set of admissible strategies $\mathcal{A}_t^{\boldsymbol{\zeta}}$ similar to that in (4.9) and we write $\mathcal{A}^{\boldsymbol{\zeta}} := \mathcal{A}_0^{\boldsymbol{\zeta}}$. Let $\boldsymbol{\nu}^{\boldsymbol{\zeta}} \in \mathcal{A}^{\boldsymbol{\zeta}}$. For an LT who trades at speed $\left(\boldsymbol{\nu}_t^{\boldsymbol{\zeta}}\right)_{t\in[0,T]}$, the inventory $\left(\tilde{y}_t^{\boldsymbol{\zeta}}\right)_{t\in[0,T]}$ evolves as

$$d\tilde{\boldsymbol{y}}_t^{\boldsymbol{\zeta}} = -\boldsymbol{\nu}_t^{\boldsymbol{\zeta}}\, dt\,,$$

and her holdings of asset X follow the dynamics

$$d\tilde{x}_t^{\boldsymbol{\zeta}} = \left(\boldsymbol{\mathcal{X}}\, \boldsymbol{R}_t - \eta\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)\, \boldsymbol{\nu}_t^{\boldsymbol{\zeta}}\right)^{\intercal} \boldsymbol{\nu}_t^{\boldsymbol{\zeta}}\, dt\,.$$

We consider the same performance criterion as that in (4.10) and let $u_{\boldsymbol{\zeta}}$ be the value function of the LT. To solve the new optimisation problem for each value of $\boldsymbol{\zeta}$, follow the same steps as above to obtain the PDE verified by $\theta_{\boldsymbol{\zeta}}$, where $w_{\boldsymbol{\zeta}}(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) = \tilde{x} + \boldsymbol{R}^{\intercal}\, \boldsymbol{\mathcal{X}}^{\intercal}\, \tilde{\boldsymbol{y}} + \theta_{\boldsymbol{\zeta}}(t,\tilde{\boldsymbol{y}},\boldsymbol{R})$, and write

$$\begin{aligned} 0 \;=& \partial_t\theta_{\boldsymbol{\zeta}} - \phi\, \tilde{\boldsymbol{y}}^{\intercal}\, \tilde{\boldsymbol{\Sigma}}\, \tilde{\boldsymbol{y}} + \left(\boldsymbol{\mu} - \boldsymbol{R}\right)^{\intercal}\, \boldsymbol{\beta}\, \left(\boldsymbol{\mathcal{X}}^{\intercal}\, \tilde{\boldsymbol{y}} + \partial_{\boldsymbol{R}}\theta_{\boldsymbol{\zeta}}\right) \\ &+ \frac{1}{2}\, \mathrm{Tr}\left(\boldsymbol{\Sigma}\, D_{\boldsymbol{RR}}\, \theta_{\boldsymbol{\zeta}}\right) + \frac{1}{4\,\eta}\, \partial_{\tilde{\boldsymbol{y}}}\, \theta_{\boldsymbol{\zeta}}^{\intercal}\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1}\, \partial_{\tilde{\boldsymbol{y}}}\theta_{\boldsymbol{\zeta}}\;, \end{aligned} \tag{4.14}$$

with terminal condition

$$\theta_{\boldsymbol{\zeta}}(T,\tilde{\boldsymbol{y}},\boldsymbol{R}) = -\tilde{\boldsymbol{y}}^{\intercal}\boldsymbol{\alpha}\, \tilde{\boldsymbol{y}}\,,$$

where the LT's optimal trading speed in feedback form is

$$\boldsymbol{\nu}^{\star,\boldsymbol{\zeta}} = -\frac{1}{2\,\eta}\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1}\, \partial_{\tilde{\boldsymbol{y}}}\theta_{\boldsymbol{\zeta}}\;. \tag{4.15}$$

To further study the problem, substitute the ansatz

$$\theta_{\boldsymbol{\zeta}}(t,\tilde{x},\tilde{\boldsymbol{y}},\boldsymbol{R}) = \tilde{\boldsymbol{y}}^{\intercal} A_{\boldsymbol{\zeta}}(t)\, \tilde{\boldsymbol{y}} + \tilde{\boldsymbol{y}}^{\intercal} B_{\boldsymbol{\zeta}}(t)\, \boldsymbol{R} + C_{\boldsymbol{\zeta}}(t)\, \tilde{\boldsymbol{y}} + \boldsymbol{R}^{\intercal} D_{\boldsymbol{\zeta}}(t)\, \boldsymbol{R} + E_{\boldsymbol{\zeta}}(t)\, \boldsymbol{R} + F_{\boldsymbol{\zeta}}(t)\,, \tag{4.16}$$

in (4.14) to obtain the system of ODEs

$$\begin{cases} 0 = A'_\zeta(t) - \phi\,\tilde{\boldsymbol{\Sigma}} + \dfrac{1}{\eta} A_\zeta(t)^\intercal \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} A_\zeta(t)\,, \\ 0 = B'_\zeta(t) - \boldsymbol{\mathcal{X}}\,\boldsymbol{\beta}^\intercal - B_\zeta(t)\,\boldsymbol{\beta}^\intercal + \dfrac{1}{\eta}\, A_\zeta(t)^\intercal\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} B_\zeta(t)\,, \\ 0 = C'_\zeta(t) + \boldsymbol{\mu}^\intercal\,\boldsymbol{\beta}\,\boldsymbol{\mathcal{X}}^\intercal + \boldsymbol{\mu}^\intercal\,\boldsymbol{\beta} B_\zeta(t)^\intercal + \dfrac{1}{\eta}\, C_\zeta(t)^\intercal\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} A_\zeta(t)\,, \\ 0 = D'_\zeta(t) + \dfrac{1}{4\,\eta} B_\zeta(t)^\intercal\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} B_\zeta(t)\,, \\ 0 = E'_\zeta(t) - E_\zeta(t)^\intercal\,\boldsymbol{\beta}^\intercal + \dfrac{1}{2\,\eta}\, C_\zeta(t)^\intercal\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1}\, B_\zeta(t)\,, \\ 0 = F'_\zeta(t) + \boldsymbol{\mu}^\intercal\,\boldsymbol{\beta} E_\zeta(t) + Tr\left(\boldsymbol{\Sigma}\, D_\zeta(t)\right) + \dfrac{1}{4\,\eta}\, C_\zeta(t)^\intercal\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} C_\zeta(t)\,, \end{cases}$$

with terminal condition

$$A_\zeta(T) = -\boldsymbol{\alpha}\,,$$
$$B_\zeta(T) = C_\zeta(T) = D_\zeta(T) = E_\zeta(T) = F_\zeta(T) = 0\,.$$

The ODEs in D, E, F, and the corresponding terminal conditions admit the unique solutions $D_\zeta = E_\zeta = F_\zeta = 0$. Next, use the classical tools for Riccati equations to show that the ODE for $A_\zeta(t)$ admits the unique solution

$$A_\zeta(t) = \eta\,\mathfrak{D}\left(\boldsymbol{\zeta}\right)^{\frac{1}{2}} \left(\boldsymbol{\Psi} + \boldsymbol{\xi}\left(t\right)^{-1}\right) \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{\frac{1}{2}}\,,$$

where

$$\begin{cases} \boldsymbol{\Psi} = \dfrac{\phi}{\eta}\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-\frac{1}{2}} \tilde{\boldsymbol{\Sigma}}\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-\frac{1}{2}}\,, \\ \boldsymbol{\xi}\left(t\right) = -\dfrac{\boldsymbol{\Psi}^{-1}}{2}\left(I - e^{-2\boldsymbol{\Psi}(T-t)}\right) - e^{-\boldsymbol{\Psi}(T-t)}\left(\boldsymbol{\Phi} + \boldsymbol{\Psi}\right)^{-1} e^{-\boldsymbol{\Psi}(T-t)}, \\ \boldsymbol{\Phi} = \dfrac{1}{\eta}\mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-\frac{1}{2}} A_\zeta(T)\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-\frac{1}{2}}\,. \end{cases}$$

Finally, use Theorem 3 in Cartea et al. (2018b) to obtain the unique solution for B_ζ and C_ζ. More precisely, observe that $C_\zeta = -\boldsymbol{\mu}^\intercal B_\zeta^\intercal$ and write

$$B_\zeta(t) = \int_t^T \boldsymbol{:}\, \boldsymbol{e}^{\int_t^u \left(\boldsymbol{\beta}^\intercal - \frac{1}{\eta} A_\zeta(s)^\intercal \mathfrak{D}(\boldsymbol{\zeta})^{-1}\right) ds} \boldsymbol{:}\, \boldsymbol{\mathcal{X}}\boldsymbol{\beta}^\intercal\, du\,,$$

where the notation $\boldsymbol{:}\, \boldsymbol{e}^{\int_t^u \cdot\, ds} \boldsymbol{:}$ denotes the time-ordered exponential.

Thus, we obtain a closed-form solution to the HJB equation and an analytical formula for the optimal trading speed in the case of fixed execution costs. In particular, substitute (4.16) into (4.15) to obtain the optimal strategy in feedback form

$$\boldsymbol{\nu}^{\star,\zeta} = \frac{1}{2\,\eta}\, \mathfrak{D}\left(\boldsymbol{\zeta}\right)^{-1} \left(B_\zeta(t)^\intercal\left(\boldsymbol{\mu} - \boldsymbol{R}\right) - 2\,\boldsymbol{A}_\zeta(t)\,\tilde{\boldsymbol{y}}\right).$$

4.4.3 Closed-form approximation strategy

Here, we use a family of closed-form strategies of the type in (4.15) to derive a piecewise-defined trading strategy which approximates the optimal trading speed in feedback form (4.13). Specifically,

we partition the space of the rate $\boldsymbol{Z}$ into hyperrectangles and define a piecewise strategy which uses a different impact parameter $\boldsymbol{\zeta}$ in each different hyperrectangle. Finally, we show that as the dimension of the hyperrectangles becomes arbitrarily small, the piecewise strategy converges to the closed-form approximation strategy.

For each $i \in \{1, \ldots, n\}$, let $\left\{Z_0^{i,N}, \ldots, Z_N^{i,N}\right\}$ be a partition of $\left[\underline{Z}^i, \overline{Z}^i\right]$, where $0 < \underline{Z}^i < \overline{Z}^i$. Here, $\underline{Z}^i$ and $\overline{Z}^i$ are are such that the LT has high confidence that the rate Z^i will be within the range $\left[\underline{Z}^i, \overline{Z}^i\right]$ during the execution window. We define $\mathfrak{R} := \left[\underline{Z}^1, \overline{Z}^1\right] \times \left[\underline{Z}^n, \overline{Z}^n\right]$. Next, for each $N \in \mathbb{N}$, $i \in \{1, \ldots, n\}$, and $j \in \{0, \ldots, N\}$ we define

$$Z_j^{i,N} := \underline{Z}^i + \frac{j}{N}\left(\overline{Z}^i - \underline{Z}^i\right) \quad \text{and} \quad \zeta_j^{i,N} := \frac{1}{\kappa^i}\left(Z_j^{i,N}\right)^{3/2},$$

and for each multi-index $\boldsymbol{j} = (j_1, \ldots, j_n) \in \{1, \ldots, N\}^n$ we define

$$\boldsymbol{\zeta}_{\boldsymbol{j}}^N := \left(\zeta_{j_1}^{1,N}, \ldots, \zeta_{j_n}^{n,N}\right), \quad \text{and} \quad \boldsymbol{\nu}^{\star,\boldsymbol{j},N} = \left(\nu^{\star,1,j_1,N}, \ldots, \nu^{\star,n,j_n,N}\right) := \boldsymbol{\nu}^{\star,\boldsymbol{\zeta}_{\boldsymbol{j}}^N}.$$

To construct the approximate trading strategy, we first define a strategy $\boldsymbol{\nu}^{\star,N}$ that uses the closed-form optimal trading speed $\boldsymbol{\nu}^{\star,\boldsymbol{j},N}$ to approximate the optimal trading speed whenever the rate is close to $\boldsymbol{\zeta}_{\boldsymbol{j}}^N$. For each $i \in \{1, \ldots, n\}$, we define the piecewise-defined trading speed $\nu^{\star,i,N}\colon [0,T] \times \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{2n+m} \to \mathbb{R}$

$$\begin{aligned}\nu^{\star,i,N}(t, \tilde{y}, \boldsymbol{R}) = \nu^{\star,i,0,N}(t, \tilde{y}, \boldsymbol{R})\, \mathbb{1}_{Z^i < Z_1^{i,N}} + \sum_{j=1}^{N-1} \nu^{\star,i,j,N}(t, \tilde{y}, \boldsymbol{R})\, \mathbb{1}_{Z^i \in [Z_j^{i,N}, Z_{j+1}^{i,N})} \\ + \nu^{\star,i,N,N}(t, \tilde{y}, \boldsymbol{R})\, \mathbb{1}_{Z^i \geq Z_N^{i,N}}.\end{aligned}$$

The strategy $\nu^{\star,i,N}(t, \tilde{y}, \boldsymbol{R})$ has first-type discontinuity points; specifically, for each $j \in \{1, \ldots, N\}$ we have $\nu^{\star,i,j,N}|_{Z^i = Z_{j+1}^{i,N}} \neq \nu^{\star,i,j+1,N}|_{Z^i = Z_{j+1}^{i,N}}$.

The theorem below shows how to partition $\mathfrak{R}$ to make the discontinuities in each $\nu^{\star,i,N}$ arbitrarily small simultaneously. Furthermore, when the distance between points in the partition becomes sufficiently small, the sequence of piecewise-defined optimal strategies $\left\{\boldsymbol{\nu}^{\star,N}\right\}_{N \in \mathbb{N}}$ converges uniformly to a continuous closed-form approximation strategy which we use in our performance study of Section 4.5.

Theorem 3. *For each $\varepsilon > 0$ there exists $N \in \mathbb{N}$ such that*

$$\max_{\substack{j=1,\ldots,N \\ i=0,\ldots,n}} \left\| \nu^{\star,i,j,N}|_{Z^i = Z_{j+1}^{i,N}} - \nu^{\star,i,j+1,N}|_{Z^i = Z_{j+1}^{i,N}} \right\| < \varepsilon.$$

Furthermore, for each $N \in \mathbb{N}$, let $\tilde{\boldsymbol{\nu}}^{\star,N} := \boldsymbol{\nu}^{\star,N}|_{[0,T] \times \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathfrak{R} \times \mathbb{R}^m}$. Then, the sequence $\left\{\tilde{\boldsymbol{\nu}}^{\star,N}\right\}$ converges to $\tilde{\nu}^\star$ uniformly in $[0,T] \times \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathfrak{R} \times \mathbb{R}^m$, where

$$\tilde{\boldsymbol{\nu}}^\star(t, \tilde{y}, \boldsymbol{R}) = \frac{1}{2\eta}\mathfrak{D}(\boldsymbol{\zeta})^{-1}\left(B\left(t, \boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)^\intercal (\boldsymbol{\mu} - \boldsymbol{R}) - 2\boldsymbol{A}\left(t, \boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)\tilde{\boldsymbol{y}}\right). \tag{4.17}$$

and

$$\boldsymbol{A}\left(t, \boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right) = \eta\,\mathfrak{D}\left(\boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)^{\frac{1}{2}}\left(\boldsymbol{\Psi} + \boldsymbol{\xi}(t)^{-1}\right)\mathfrak{D}\left(\boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)^{\frac{1}{2}},$$

$$\boldsymbol{B}\left(t, \boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right) = \int_t^T \boldsymbol{:}\, e^{\int_t^u \left(\boldsymbol{\beta}^\intercal - \frac{1}{\eta}\boldsymbol{A}\left(s, \boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)^\intercal \mathfrak{D}\left(\boldsymbol{\kappa}^{-1} \odot \boldsymbol{Z}^{3/2}\right)^{-1}\right) ds} \boldsymbol{:}\, \boldsymbol{\mathcal{X}}\boldsymbol{\beta}^\intercal\, du, \ .$$

The proof for Theorem 3 is a trivial generalisation of the proof in A.1.

4.5 Performance analysis

We use transaction data to show the performance of the closed-form approximation strategy in (4.17). Similar to Section 4.2, we use transaction data for ETH and BTC in two pools of the CPMM Uniswap v3 and the more liquid exchange Binance.

To estimate the execution costs incurred by the LT in the pools we consider, one needs to track the depth of the pool at every instant. We use historical LP operations to reconstruct the historical depth κ of the two pools, which we use to estimate the execution costs during the execution programmes of the LT.

Here, we perform consecutive in-sample estimation of model parameters using 24 hours of transaction data sampled at the minute frequency, and out-of-sample execution of the closed-form approximation strategy (4.17) using T =12 hours, $\phi = 1 \times 10^{-6}$, and $\boldsymbol{\alpha} = \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix}$.

For each individual run, we compute the optimal strategy and compute the terminal profit and loss of the execution programme, which is $\tilde{\boldsymbol{x}}_T + \tilde{\boldsymbol{y}}_T^\intercal \boldsymbol{Z}_T - \tilde{\boldsymbol{y}}_0^\intercal \boldsymbol{Z}_0$. Next, we shift both the in-sample and the out-of-sample time windows by 12 hours and repeat the process, thus performing 357 runs in total. In each run, the agent must (randomly with probability 1/2) either liquidate 1000 units of ETH, and 100 units of BTC, or buy the same amounts.

To estimate the coefficients of the process driving the joint dynamics of the exchange rates, we use the VAR(1) model

$$\Delta R_t = a + \Pi \, \Delta R_{t-1} + \varepsilon_t \,,$$

which is a time discretisation of our multi-OU process; where $a \in \mathbb{R}^{2n+m}$, $\Pi \in \mathcal{M}_{2n+m}(\mathbb{R})$, and ε is white noise. We use in-sample data and a least squares regression to estimate the coefficients of the VAR model, from which one obtains the mean-reversion matrix $\boldsymbol{\beta}$, the long-term unconditional mean $\boldsymbol{\mu}$, and the covariance matrix $\boldsymbol{\Sigma}$.

As an example, Figure 4.5 shows the optimal inventory when the in-sample data is that of 1 February 2022 and the execution programme is over the following 12 hours. Figure 4.5 shows that the strategy uses the spread between the oracle rates and the pool rates for both pairs to predict a potential decrease or increase in the rate. The depth of the ETH/USDC 0.05% pool is larger than that of the BTC/USDC 0.3%; see Table 4.1, so trading costs caused by the convexity of the level function are low in the first pool and high in the second. Hence, the speculative component of the closed-form approximation strategy (4.17) is less exploited in the less liquid pool even though the spread between the oracle and the pool rate is wide.

In our analysis, we compare the performance of our strategy with that of a classical Almgren-Chriss (AC) strategy and a TWAP strategy; see Guéant (2016). Moreover, we run the strategy (4.17) with zero initial inventory, which corresponds to an arbitrage strategy that uses the cointegration factors to take speculative positions. Table 4.6 shows the distribution of the terminal P&L for the three strategies and shows that the closed-form approximation strategy outperforms TWAP and AC on average, and is profitable for statistical arbitrage.

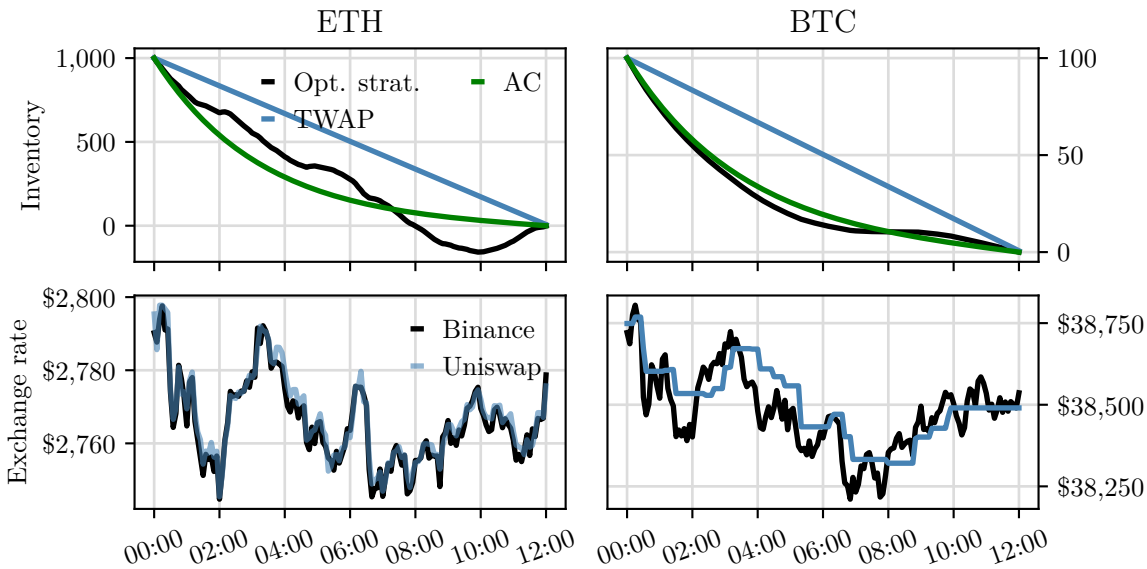


Figure 4.5: Inventory in ETH and BTC during the execution programme (top) and the pool and oracle exchange rates (bottom). Model parameters are obtained using data between 00:00 on 1 February and 00:00 on 2 February 2022, and the execution programme is between 00:00 on 2 February and 12:00 on 2 February 2022.

	Avg. P&L	Std. Dev.
Optimal	$ 2,328	$ 54,805
Optimal (stat. arb.)	$ 2,658	$ 5,819
AC	$ -343	$ 54,823
TWAP	$ -3,705	$ 99,946

Table 4.6: Average and standard deviation of the terminal P&L with strategy (4.17) (liquidation and statistical arbitrage) and from executing TWAP and AC for 357 runs of in-sample estimation of model parameters and out-of-sample execution.

4.6 Conclusions

In this chapter, we derived the optimal strategy for a liquidity taker (LT) who trades in an AMM a basket of crypto-currencies whose constituents co-move. The LT uses market signals and exchange rate information from relevant AMMs and traditional venues to enhance the performance of her strategy. We used stochastic control tools to derive a closed-form strategy that can be computed and implemented by the LT in real time. Finally, we used market data from Uniswap v3 and Binance to study co-movements between crypto-currencies and lead-lag effects between trading venues, and to showcase the performance of the strategy.

Chapter 5

Optimal Liquidity Provision in constant product market makers with concentrated liquidity

5.1 Introduction

Traditional electronic exchanges are organised around limit order books to clear demand and supply of liquidity. In contrast, the takers and providers of liquidity in constant function market makers (CFMMs) interact in liquidity pools; liquidity providers (LPs) deposit their assets in the liquidity pool and liquidity takers (LTs) exchange assets directly with the pool. At present, constant product market makers (CPMMs) with concentrated liquidity (CL) are the most popular type of CFMM, with Uniswap v3 as a prime example; see Adams et al. (2021). In CPMMs with CL, LPs specify the rate intervals (i.e., tick ranges) over which they deposit their assets, and this liquidity is counterparty to trades of LTs when the marginal exchange rate of the pool is within the liquidity range of the LPs. When LPs deposit liquidity, fees paid by LTs accrue and are paid to LPs when they withdraw their assets from the pool. The amount of fees accrued to LPs is proportional to the share of liquidity they hold in each liquidity range of the pool.

Existing research characterises the losses of LPs, but does not offer tools for strategic liquidity provision. In this chapter, we study strategic liquidity provision in CPMMs with CL. We derive the continuous-time dynamics of the wealth of strategic LPs which consists of the position they hold in the pool (position value), fee income, and costs from repositioning their liquidity position. The width of the range where the assets are deposited affects the value of the LP's position in the pool and we show that fee income is subject to a tradeoff between the width of the LP's liquidity range and the volatility of the marginal rate in the pool. More precisely, CL increases fee revenue when the rate is in the range of the LP, but also increases *concentration risk*. Concentration risk refers to the risk that the LP faces when her position is concentrated in narrow ranges; the LP stops collecting fees when the rate exits the range of her position.

We derive an optimal dynamic strategy to provide liquidity in a CPMM with CL. In our model, the LP maximises the expected utility of her terminal wealth, which consists of the accumulated trading fees and the gains and losses from the market making strategy. The dynamic strategy

controls the width and the skew of liquidity that targets the marginal exchange rate. For the particular case of log-utility, we obtain the strategy in closed-form and show how the solution balances the opposing effects between volatility, convexity, and fee collection. When volatility increases, there is an incentive for the LP to widen the range of liquidity provision. In particular, in the extreme case of very high volatility, the LP must withdraw from the pool. Also, when there is an increase in the potential provision fees that the LP may collect because of higher liquidity taking activity, the strategy balances two opposing forces. One, there is an incentive to increase fee collection by concentrating the liquidity of the LP in a tight range around the exchange rate of the pool. Two, there is an incentive to limit the losses due to concentration risk by widening the range of liquidity provision. Finally, when the drift of the marginal exchange rate is stochastic (e.g., a predictive signal), the strategy skews the range of liquidity to increase fee revenue, by capturing the LT trading flow, and to increase the position value, by profiting from the expected changes in the marginal rate.

Finally, we use Uniswap v3 data to motivate our model and to test the performance of the strategy we derive. The LP and LT data are from the pool ETH/USDC (Ethereum and USD coin) between the inception of the pool on 5 May 2021 and 18 August 2022. To illustrate the performance of the strategy we use in-sample data to estimate model parameters and out-of-sample data to test the strategy. Our analysis of the historical transactions in Uniswap v3 shows that LPs have traded at a significant loss, on average, in the ETH/USDC pool. We show that the out-of-sample performance of our strategy is considerably superior to the average LP performance we observe in the ETH/USDC pool.

Early works on AMMs are in Chiu and Koeppl (2019); Angeris et al. (2021); Lipton and Treccani (2021). Some works in the literature study strategic liquidity provision in CFMMs and CPMMs with CL. Heimbach et al. (2022) discuss the tradeoff between risks and returns that LPs face in Uniswap v3, Cartea et al. (2023b) study the predictable losses of LPs in a continuous-time setup, Milionis et al. (2023) study the impact of fees on the profits of arbitrageurs in CFMMs, Fukasawa et al. (2023) study the hedging of the impermanent losses of LPs, and Lı et al. (2023) study the economics of liquidity provision. Closest to our work are the models in Fan et al. (2021) and Fan et al. (2022) which focus on fee revenue and use approximation techniques to obtain dynamic strategies. Finally, there is a growing literature on AMM design for fair competition between LPs and LTs. Goyal et al. (2023) study an AMM with dynamic trading functions that incorporate beliefs of LPs, Lommers et al. (2023) study AMMs where the LP's strategy adjusts dynamically to market information, and Cartea et al. (2023d) generalise CFMMs and propose AMM designs where LPs express their beliefs and risk preferences.

Our work is related to the algorithmic trading and optimal market making literature. Early works on liquidity provision in traditional markets are Ho and Stoll (1983), Biais (1993), and Avellaneda and Stoikov (2008) with extensions in many directions; see Cartea et al. (2014, 2017); Guéant (2017); Bergault et al. (2021); Drissi (2022). We refer the reader to Cartea et al. (2015), Guéant (2016), and Donnelly (2022) for a comprehensive review of algorithmic trading models for

takers and makers of liquidity in traditional markets. Also, our work is related to those in Cartea et al. (2018a); Barger and Lorig (2019); Cartea and Wang (2020); Donnelly and Lorig (2020); Forde et al. (2022); Bergault et al. (2022) which implement market signals in algorithmic trading strategies.

The remainder of the chapter proceeds as follows. Section 5.2 describes CL pools. Section 5.3 studies the continuous-time dynamics of the wealth of LPs as a result of the position value, the fee revenue, and rebalancing costs. In particular, we use Uniswap v3 data to study the fee revenue component of the LP's wealth and our results motivate the assumptions in our model. Section 5.4 introduces our liquidity provision model and uses stochastic control to derive a closed-form optimal strategy. Next, we study how the strategy controls the width and the skew of the liquidity range as a function of the pool's profitability, PL, concentration risk, and the drift in the marginal rate. Finally, Section 5.5 uses Uniswap v3 data to test the performance of the strategy and showcases its superior performance.

5.2 Constant function market makers and concentrated liquidity

5.2.1 Constant function market makers

Consider a reference asset X and a risky asset Y which is valued in units of X. Assume there is a pool that makes liquidity for the pair of assets X and Y, and denote by Z the marginal exchange rate of asset Y in units of asset X in the pool. In a CFMM that charges a fee τ proportional to trade size, the trading function $f(x,y) = \kappa^2$ links the state of the pool before and after a trade is executed, where x and y are the quantities of asset X and Y that constitute the *reserves* in the pool, κ is the depth of the pool, and f is increasing in both arguments. We write $f(x,y) = \kappa^2$ as $x = \varphi_\kappa(y)$ for an appropriate decreasing *level function* φ_κ.

We denote the execution rate for a traded quantity $\pm\Delta y$ by $\tilde{Z}(\pm\Delta y)$, where $\Delta y \geq 0$. When an LT buys Δy of asset Y, she pays $\Delta x = \Delta y \times \tilde{Z}(y)$ of asset X, where

$$f(x + (1-\tau)\Delta x, y - \Delta y) = \kappa^2 \quad \Longrightarrow \quad \tilde{Z}(\Delta y) = \frac{\varphi_\kappa(y - \Delta y) - \varphi_\kappa(y)}{(1-\tau)\,\Delta y}. \tag{5.1}$$

Similarly, when an LT sells y of asset Y, she receives $x = y \times \tilde{Z}(-y)$ of asset X, where

$$f(x + \Delta x, y + (1-\tau)\Delta y) = \kappa^2 \quad \Longrightarrow \quad \tilde{Z}(-\Delta y) = \frac{\varphi_\kappa(y) - \varphi_\kappa(y + (1-\tau)\,\Delta y)}{\Delta y}. \tag{5.2}$$

In CL markets, the marginal exchange rate is $Z = -\varphi'_\kappa(y)$, which is the price of an infinitesimal trade when fees are zero, i.e., when $\Delta y \to 0$ and $\tau = 0$.[1] In traditional CPMMs such as Uniswap v2, the trading function is $f(x,y) = x \times y$, so the level function is $\varphi_\kappa(y) = \kappa^2/y$ and the marginal exchange rate is $Z = x/y$.

Liquidity provision operations in CPMMs do not impact the marginal rate, so when an LP deposits the quantities x and y of assets X and Y, the condition $Z = x/y = (x + \Delta x)/(y + \Delta y)$ must be satisfied; see (3.3).

[1] When fees are not zero, the exchange rates for infinitesimal trades are $\lim_{\Delta y \to 0} \tilde{Z}(\Delta y) = -\frac{1}{1-\tau}\,\varphi'_\kappa(y) = \frac{1}{1-\tau}\,Z$ and $\lim_{\Delta y \to 0} \tilde{Z}(-\Delta y) = -(1-\tau)\,\varphi'_\kappa(y) = (1-\tau)\,Z$; see (5.1) and (5.2).

5.2.2 Concentrated liquidity markets

This chapter focuses on liquidity provision in CPMMs with CL. In pools with CL, the space of rates where an LP provides liquidity is discretised into a set of values $\{\mathcal{Z}(i)\}_{i\in\mathbb{Z}}$ called ticks.[2] For instance, in Uniswap v3 each tick $\mathcal{Z}(i)$ is uniquely identified by an integer $i \in \mathbb{Z}$ according to

$$\mathcal{Z}(i) = 1.0001^i \,. \tag{5.3}$$

The boundaries of the LP's position in the pool take values in the set of ticks. The range between two consecutive ticks defines the smallest range available for LPs. Every tick range $(\mathcal{Z}(i), \mathcal{Z}(i+1)]$ has its own depth which determines the execution cost of LT trades when the instantaneous rate Z is in that tick range.

The LP specifies a lower tick $\mathcal{Z}(L)$ and an upper tick $\mathcal{Z}(U)$ between which she provides liquidity that is evenly spread over all the tick ranges in the position range $(\mathcal{Z}(L), \mathcal{Z}(U)]$. The position of an LP who provides liquidity in a range $(\mathcal{Z}(L), \mathcal{Z}(U)]$ is characterised by the position depth $\tilde{\kappa}$. The value of $\tilde{\kappa}$ represents the depth of the LP's liquidity within every tick range in her position range, and $\tilde{\kappa}/\kappa^i$ represents the portion of liquidity that the LP holds in the tick range $(\mathcal{Z}(i), \mathcal{Z}(i+1)] \subset (\mathcal{Z}(L), \mathcal{Z}(U)]$.

The assets that the LP deposits in a range $(\mathcal{Z}(L), \mathcal{Z}(U)]$ provide the liquidity that supports marginal rate movements between $\mathcal{Z}(L)$ and $\mathcal{Z}(U)$. The quantities $\tilde{x}$ and $\tilde{y}$ that the LP provides verify the key formulae

$$\begin{cases} \tilde{x} = 0 & \text{and} \quad \tilde{y} = \tilde{\kappa}\left(\frac{1}{\sqrt{\mathcal{Z}(L)}} - \frac{1}{\sqrt{\mathcal{Z}(U)}}\right) & \text{if} \quad Z < \mathcal{Z}(L)\,, \\ \tilde{x} = \tilde{\kappa}\left(\sqrt{Z} - \sqrt{\mathcal{Z}(L)}\right) & \text{and} \quad \tilde{y} = \tilde{\kappa}\left(\frac{1}{\sqrt{Z}} - \frac{1}{\sqrt{\mathcal{Z}(U)}}\right) & \text{if} \quad \mathcal{Z}(L) \le Z < \mathcal{Z}(U)\,, \\ \tilde{x} = \tilde{\kappa}\left(\sqrt{\mathcal{Z}(U)} - \sqrt{\mathcal{Z}(L)}\right) & \text{and} \quad \tilde{y} = 0 & \text{if} \quad Z \ge \mathcal{Z}(U)\,. \end{cases} \tag{5.4}$$

When the rate Z changes, the equations in (5.4) and the prevailing marginal rate Z determine the holdings of the LP in the pool, in particular, they determine the quantities of each asset received by the LP when she withdraws her liquidity.

Within each tick range, the constant product formulae (5.1)–(5.2) determine the dynamics of the marginal rate, where the depth κ is the total depth of liquidity in that tick range. To obtain the total depth in a tick range, one sums the depths of the individual liquidity positions in the same tick range; see Drissi (2023). When a liquidity taking trade is large, so the marginal rate crosses the boundary of a tick range, the pool executes two separate trades with potentially different depths for the constant product formula.

The proportional fee τ charged by the pool to LTs is distributed among LPs. More precisely, if an LP's liquidity position with depth $\tilde{\kappa}$ is in a tick range where the total depth of liquidity is κ, then for every liquidity taking trade that pays an amount p of fees, the LP with liquidity $\tilde{\kappa}$ earns the amount

$$\tilde{p} = \frac{\tilde{\kappa}}{\kappa}\, p\, \mathbb{1}_{\{\mathcal{Z}(L) < Z \le \mathcal{Z}(U)\}}\,. \tag{5.5}$$

[2] In traditional limit order books, a tick is the smallest price increment.

Thus, the larger is the position depth $\tilde{\kappa}$, the higher is the proportion of fees that the LP earns.[3]

5.3 The wealth of liquidity providers in CL pools

In this section, we consider a strategic LP who dynamically tracks the marginal rate Z. In our model, the LP's position is self-financed throughout an investment window $[0, T]$, so the LP repeatedly withdraws her liquidity and collects the accumulated fees, then uses her wealth, i.e., the collected fees and the assets she withdraws, to deposit liquidity in a new range. In the remainder of this work, we work in a filtered probability space $\left(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{F} = (\mathcal{F}_t)_{t\in[0,T]}\right)$ that satisfies the usual conditions, where $\mathbb{F}$ is the natural filtration generated by the collection of observable stochastic processes that we define below.

We assume that the marginal exchange rate in the pool $(Z_t)_{t\in[0,T]}$ is driven by a stochastic drift $(\mu_t)_{t\in[0,T]}$ and we write

$$dZ_t = \mu_t \, Z_t \, dt + \sigma \, Z_t \, dW_t \,, \tag{5.6}$$

where the volatility parameter σ is a nonnegative constant and $(W_t)_{t\in[0,T]}$ is a standard Brownian motion independent of μ. We assume that μ is càdlàg with finite fourth moment, i.e., $\mathbb{E}\left[\mu_t^4\right] < +\infty$ for $0 \leq t \leq T$. The LP observes and uses μ to optimise her liquidity positions and improve trading performance.

Consider an LP who provides liquidity in the CPMM with CL during an investment horizon $[0, T]$, with $T > 0$. At time t, she chooses a range $\left(Z_t^L, Z_t^U\right]$ to provide liquidity. Note that Z_t^L, Z_t^U do not necessarily correspond to ticks as defined in (5.3); in practice, when the LP implements her strategy in real time, she rounds Z_t^L, and Z_t^U to the closest ticks.

The LP's initial wealth is V_0, in units of X, and at time $t = 0$, she deposits quantities $(\tilde{x}_0, \tilde{y}_0)$ in the range $\left(Z_0^L, Z_0^U\right]$, so the initial depth of her position is $\tilde{\kappa}_0$, and the value of her initial position, marked-to-market in units of X, is $V_0 = \tilde{x}_0 + \tilde{y}_0 \, Z_0$. The dynamics of the LP's wealth consist of the value of the liquidity position in the pool $(\alpha_t)_{t\in[0,T]}$, the fee revenue $(p_t)_{t\in[0,T]}$, and the rebalancing costs $(c_t)_{t\in[0,T]}$. We introduce the wealth process $(V_t = \alpha_t + p_t + c_t)_{t\in[0,T]}$, which we mark-to-market in units of the reference asset X, with $V_0 > 0$ known. At any time t, the LP uses her wealth V_t to provide liquidity. Next, Subsection 5.3.1 studies the dynamics of the LP's position α in the pool, Subsection 5.3.2 studies the dynamics of the LP's fee revenue p, and Subsection 5.3.3 studies the dynamics of the rebalancing costs c.

5.3.1 Position value

In this section, we focus our analysis on the *position value* α. Throughout the investment window $[0, T]$, the holdings $(\tilde{x}_t, \tilde{y}_t)_{t\in[0,T]}$ of the LP change because the marginal rate Z changes and because she continuously adjusts her liquidity range around Z. More precisely, to make markets optimally,

[3] For instance, if the LP is the only provider of liquidity in the range $(Z^\ell, Z^u]$ then $\kappa = \tilde{\kappa}$, so the LP collects all the fees in that range.

the LP controls the values of $\left(\delta_t^L\right)_{t\in[0,T]}$ and $\left(\delta_t^U\right)_{t\in[0,T]}$ which determine the dynamic liquidity provision boundaries $\left(Z_t^L\right)_{t\in[0,T]}$ and $\left(Z_t^U\right)_{t\in[0,T]}$ as follows:

$$\begin{aligned}\sqrt{Z_t^U} =& \sqrt{Z_t}\left(1-\frac{\delta_t^U}{2}\right)^{-1},\\ \sqrt{Z_t^L} =& \sqrt{Z_t}\left(1-\frac{\delta_t^L}{2}\right),\end{aligned} \tag{5.7}$$

where $\delta^L \in (-\infty, 2]$, $\delta^U \in [-\infty, 2)$, and $\delta^L\,\delta^U/2 < \delta^L + \delta^U$ because $0 \leq Z^L < Z^U < \infty$. We define δ^L and δ^U in (5.7) in terms of $\sqrt{Z}$ to simplify and linearise the CL constant product formulae; see Cartea et al. (2023b) for more details.

In practice, the LP earns fees when the rate Z_t is in the LP's liquidity range $(Z_t^L, Z_t^U]$, so $\delta^L \in (0, 2]$, $\delta^U \in [0, 2)$, and $\delta^L\,\delta^U/2 < \delta^L + \delta^U$.Below, Section 5.4 considers a problem where the controls are not constrained, and values $\delta^L \notin (0, 2]$, $\delta^U \notin [0, 2)$ are those where liquidity provision is unprofitable.

In the remainder of this chapter we define the *spread* δ_t of the LP's position as

$$\delta_t = \delta_t^U + \delta_t^L\,, \tag{5.8}$$

and we consider admissible strategies δ that are $\mathbb{R}$-valued and such that $\int_0^T \delta_t^{-4}\,dt < \infty$, almost surely; see Section 5.4 for more details. For small position spreads, we use the first-order Taylor expansion to write the approximation

$$\frac{Z_t^U - Z_t^L}{Z_t} = \left(1-\frac{\delta_t^U}{2}\right)^{-2} - \left(1-\frac{\delta_t^L}{2}\right)^{2} \approx \delta_t.$$

We assume that the marginal rate process $(Z_t)_{t\in[0,T]}$ follows the dynamics (5.6). Cartea et al. (2023b) show that the holdings in assets X and Y in the pool for an LP who follows an arbitrary strategy $\left(Z_t^L, Z_t^U\right)$ are given by

$$\tilde{x}_t = \frac{\delta_t^L}{\delta_t^L + \delta_t^U}\,\alpha_t \quad \text{and} \quad \tilde{y}_t = \frac{\delta_t^U}{Z_t\left(\delta_t^L + \delta_t^U\right)}\,\alpha_t\,, \tag{5.9}$$

so the value $(\alpha_t)_{t\in[0,T]}$ of her position follows the dynamics

$$d\alpha_t = \; V_t\left(\frac{1}{\delta_t^L + \delta_t^U}\right)\left(-\frac{\sigma^2}{2}\,dt + \mu_t\,\delta_t^U\,dt + \sigma\,\delta_t^U\,dW_t\right). \tag{5.10}$$

The dynamics in (5.10) show that a larger position spread δ reduces the LP's exposure to the volatility of the pool.

For a fixed value of the spread $\delta_t = \delta_t^L + \delta_t^U$, the dynamics in (5.10) show that if $\mu_t \geq 0$, then the LP increases her expected wealth by increasing the value δ^U, i.e., by skewing her range of liquidity to the right. However, note that the quadratic variation of the LP's position value is $d\langle\alpha,\alpha\rangle_t = V_t^2\,\sigma^2\left(\delta_t^U/\delta_t\right)^2\,dt$, so skewing the range to the right also increases the variance of the LP's position. On the other hand, if $\mu \leq 0$, then the LP reduces her losses by decreasing the value of δ^U or equivalently increasing the value of δ^L, i.e., by skewing her range of liquidity to the left. Thus, the LP uses the expected changes in the marginal rate to skew the range of liquidity and to increase her expected terminal wealth.

5.3.2 Fee income

5.3.2.1 Fee income: pool fee rate

The dynamics of the fee income in our model of Section 5.4 uses a fixed depth κ and assumes that the pool generates fee income for all LPs at an instantaneous *pool fee rate* π; clearly, these fees are paid by LTs who interact with the pool, see (5.1)–(5.2). The value of π represents the instantaneous profitability of the pool, akin to the size of market orders and their arrival rate in LOBs.

To analyse the dynamics of the pool fee rate π, we use historical LT transactions in Uniswap v3 as a measure of activity and to estimate the total fee income generated by the pool; Table 4.1 provides descriptive statistics. Figure 5.1 shows the estimated fee rate π in the ETH/USDC pool. For any time t, we use[4]

$$\pi_t = 0.05\% \, \frac{\Xi_t}{2\,\kappa\,\sqrt{Z_t}}\,,$$

where Ξ_t is the volume of LT transactions the day before t, $2\,\kappa\,\sqrt{Z_t}$ is the pool value in terms of asset X at time t, and 0.05% is the fixed fee of the pool.[5] Figure 5.1 suggests that the pool fee rate π generated by liquidity taking activity in the pool is stochastic and mean reverting. Here, we assume that π is independent of the rate Z over the time scales we consider; Table 5.1 shows that the pool fee rate is weakly correlated to the rate Z at different sampling frequencies, especially for the higher frequencies we consider in our numerical tests. In Section 5.4, the pool fee rate π follows Cox-Ingersoll-Ross-type dynamics.

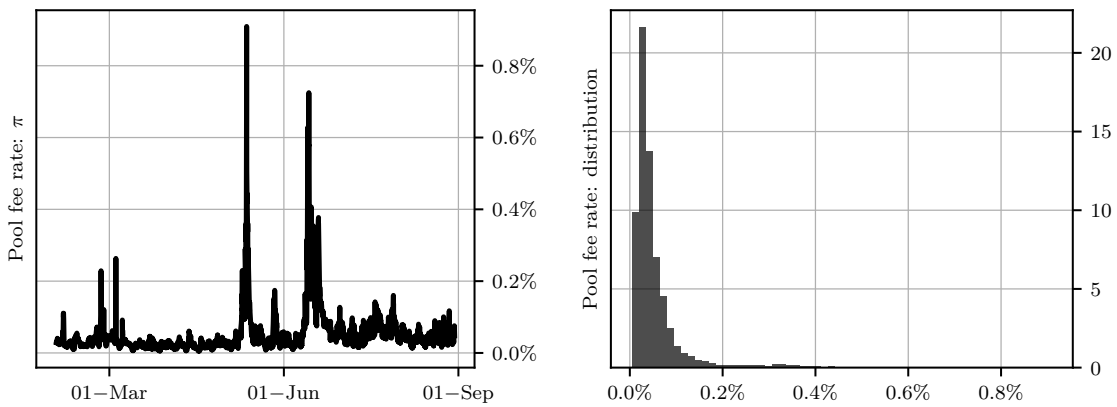


Figure 5.1: Estimated pool fee rate from February to August 2022 in the ETH/USDC pool. For any time t, the pool fee rate is the total fee income, as a percentage of the total pool size, paid by LTs on the period $[t-1\text{ day}, t]$. The pool size at time t is $2\,\kappa\,\sqrt{Z_t}$ in terms of asset X, where Z_t is the active rate in the pool at time t. Left panel: historical values of the pool fee rate. Right panel: distribution of the pool fee rate.

5.3.2.2 Fee income: spread and concentration risk

In the three cases of (5.4), increasing the spread reduces the depth $\tilde{\kappa}$ of the LP's position. Recall that the LP fee income is proportional to $\tilde{\kappa}/\kappa$, where κ is the pool depth. Thus, decreasing the

[4]Pools have different fee rates in Uniswap v3: 0.01%, 0.05%, and 0.1%.

[5]The value of a CPMM pool is given by the active pool depth. In particular, the size of a pool with quantity x of asset X and quantity y of asset Y is $x + Z\,y = 2\,x = 2\,\kappa\,Z^{1/2}$ units of X because $x\,y = \kappa^2$ and $Z = x/y$ so $x^2 = \kappa^2\,Z$.

	$\Delta t = 1$ minute	$\Delta t = 5$ minutes	$\Delta t = 1$ hour	$\Delta t = 1$ day
Correlation	−2.1%	−2.4%	−2.6%	−10.9%
Two-tailed p-value	2.0%	8.9%	8.2%	16.7%
R^2 regression	0.01%	1.1%	2.9%	5.0%

Table 5.1: First row: correlation of the returns of the rate Z and the fee rate π, i.e., $(Z_{t+\Delta t} - Z_t)/Z_t$ and $(Z_{t+\Delta t} - Z_t)/Z_t$ for $\Delta t = 1$ minute, five minutes, one hour, and one day, using data of the ETH/USDC pool between 5 May 2021 and 18 August 2022. Second row: two-tailed p-value of the t-test. Final row: R-squared of regression of the pool fee rate's returns against the marginal exchange rate's returns.

value of $\tilde{\kappa}$ potentially reduces LP fee income. Figure 5.2 shows the value of $\tilde{\kappa}$ as a function of the spread δ.

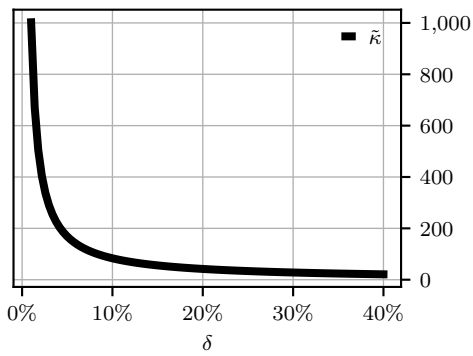


Figure 5.2: Value of the depth $\tilde{\kappa}$ of the LP's position in the pool as a function of the spread δ. The spread is in percentage of the marginal exchange rate; recall that $\left(Z_t^U - Z_t^L\right)/Z_t \approx \delta_t$.

However, although narrow ranges increase the potential fee income, they also increase concentration risk; a wide spread (i.e., a lower value of the depth $\tilde{\kappa}$) decreases fee income per LT transaction but reaps earnings from a larger number of LT transactions because the position is active for longer periods of time (i.e., it takes longer, on average, for Z to exit the LP's range). Thus, the LP must strike a balance between maximising the depth $\tilde{\kappa}$ around the rate and minimising the concentration risk, which depends on the volatility of the rate Z.

In our model, the LP continuously adjusts her position around the current rate Z, so we write the continuous-time dynamics of (5.5), conditional on the rate not exiting the LP's range, as

$$dp_t = \underbrace{(\tilde{\kappa}_t / \kappa)}_{\text{Position depth}} \underbrace{\pi_t}_{\text{Fee rate}} \underbrace{2\,\kappa\,\sqrt{Z_t}}_{\text{Pool size}}\, dt\,, \tag{5.11}$$

where $(\tilde{\kappa}_t)_{t\in[0,T]}$ models the depth of the LP's position and p is the LP's fee income for providing liquidity with depth $\tilde{\kappa}$ in the pool. The fee income is proportional to the pool size, i.e., proportional to $2\,\kappa\,\sqrt{Z_t}$. Next, use the second equation in (5.4) and equations (5.7)–(5.9) to write the dynamics of the LP's position depth $\tilde{\kappa}_t$ as

$$\tilde{\kappa}_t = 2\,V_t \left(\frac{1}{\delta_t^L + \delta_t^U}\right) \frac{1}{\sqrt{Z_t}}\,,$$

so the dynamics in (5.11) become

$$dp_t = \left(\frac{4}{\delta_t^L + \delta_t^U} \right) \pi_t \, V_t \, dt \, . \tag{5.12}$$

In practice, there is latency in the market and the LP cannot reposition her liquidity position and rebalance her assets continuously. Thus, the LP faces concentration risk in between the times she repositions her liquidity; narrow spreads generate less fee income because the rate Z may exit the range of the LP's liquidity, especially in volatile markets.

To model the losses due to concentration risk in the continuous-time dynamics (5.12) of the LP's fee revenue, we introduce an instantaneous concentration cost that reduces the fees collected by the LP as a function of the spread; the concentration cost increases (decreases) when the spread narrows (widens). We modify the dynamics of the fees collected by the LP in (5.12) as follows

$$dp_t = \left(\frac{4}{\delta_t^L + \delta_t^U} \right) \pi_t \, V_t \, dt \, - \gamma \left(\frac{1}{\delta_t^L + \delta_t^u} \right)^2 V_t \, dt \, , \tag{5.13}$$

where $\gamma > 0$ is the concentration cost parameter and V_t is the wealth invested by the LP in the pool at time t.

To justify the form of the concentration cost, we study the realised fee revenue in the ETH/USDC pool as a function of the spread δ in (5.8) for rebalancing frequencies $m = 1$ minute and $m = 5$ minutes. We denote by $\widehat{p}_{\delta,m}$ the average realised fee revenue earned by an LP in the ETH/USDC pool who provides liquidity with wealth $V = 1$, in a range with spread δ, around the marginal rate Z throughout a time window of length m. The left panel of Figure 5.3 shows that the value of δ which maximises the fee revenue $\widehat{p}_{\delta,m}$ is strictly positive for both values of m that we consider. In particular, narrow ranges reduce fee revenue due to concentration risk.

The form of the fee revenue dynamics (5.13) is a second order Taylor approximation that captures the specific shape of the fee revenue in CL markets; see left panel of Figure 5.3. The LP uses the regression model

$$\delta^2 \, \widehat{p}_{\delta,m} = 4 \, \pi \, m \, \delta - \gamma \, m \, ,$$

which is based on the dynamics (5.13), to estimate the concentration cost parameter γ. The right panel of Figure 5.3 shows that $\delta^2 \, \widehat{p}_{\delta,m}$ is affine in δ and that the estimated concentration cost parameter γ depends on the rebalancing frequency of the LP. In particular, the figure shows that the dynamics (5.13) and the second order approximation are suitable to describe the realised fee revenue in CL markets. The performance study of Section 5.5 uses this methodology to set the value of the concentration cost parameter γ.

5.3.2.3 Fee income: drift and asymmetry

The stochastic drift μ indicates the future expected changes of the marginal exchange rate in the pool. In practice, the LP may use a predictive signal so that μ represents the belief that the LP holds over the future marginal rate. For an LP who maximises fee revenue, it is natural to consider

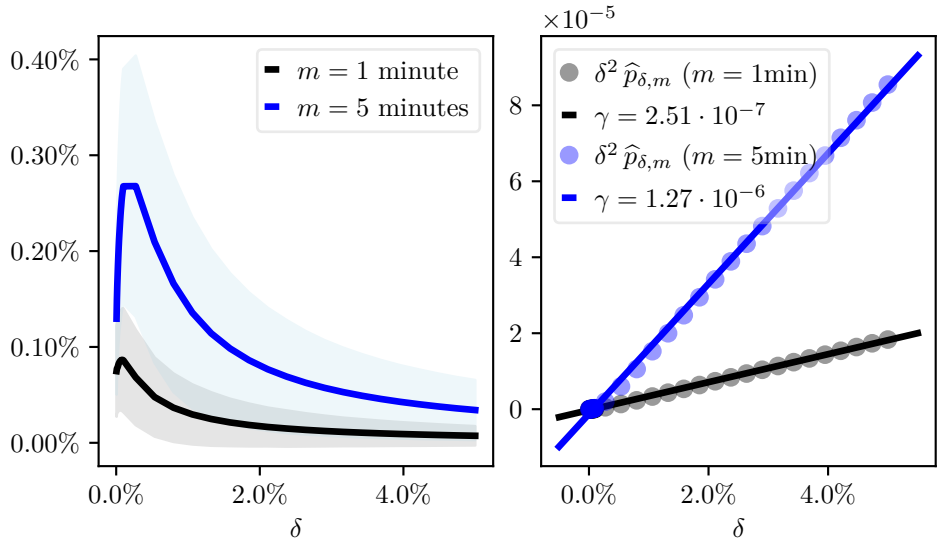


Figure 5.3: Left panel: mean and standard deviation of the fee revenue of hypothetical liquidity positions as a function of the position spread δ for the rebalancing frequencies $m = 1$ minute and $m = 5$ minutes. The fee revenue of each position considers a wealth $V = 1$ and uses historical LT transactions of the ETH/USDC pool of Uniswap v3. Right panel: $\delta^2\,\widehat{p}_{\delta,m}$ as a function of the position spread δ for the rebalancing frequencies $m = 1$ minute and $m = 5$ minutes. The concentration cost parameter is estimated as $-\iota/m$ where ι is the intercept of the regression of $\delta^2\,\widehat{p}_{\delta,m}$ on δ.

asymmetric liquidity positions that capture the liquidity taking flow. We define the *asymmetry* of a position as

$$\rho_t = \frac{\delta_t^U}{\delta_t^U + \delta_t^L} = \frac{\delta_t^U}{\delta_t}\,, \tag{5.14}$$

where δ_t^U and δ_t^L are defined in (5.7). In one extreme, when the asymmetry $\rho \to 0$, then $Z^U \to Z$ and the position consists of only asset X, and in the other extreme, when $\rho \to 1$, then $Z^L \to Z$ and the position consists of only asset Y.

In the remainder of this work, the asymmetry of the LP's position is a function of the observed drift:

$$\rho_t = \rho\left(\delta_t, \mu_t\right) = \frac{1}{2} + \frac{\mu_t}{\delta_t} = \frac{1}{2} + \frac{\mu_t}{\delta_t^U + \delta_t^L}\,, \quad \forall t \in [0, T]\,. \tag{5.15}$$

The asymmetry (5.15) adapts the skew of the position to the expected drift of the marginal rate. When $\mu = 0$, the position is symmetric around the marginal rate and $\rho_t = 1/2$, so $\delta_t^L = \delta_t^U$. When $\mu > 0$, the position is skewed to the right (i.e., $\delta_t^U > \delta_t^L$) to capture more LT trades and fee revenue, and similarly when $\mu < 0$, the position is skewed to the left (i.e., $\delta_t^U < \delta_t^L$). Also, when $\mu > 0$ and the position is skewed to the right according to (5.15), the proportion of asset Y increases and the position profits from rate appreciation, and when the position is skewed to the left, the proportion of asset Y decreases and the position is protected from rate depreciation.

Optimal liquidity provision is the dynamic choice of δ^U and δ^L over a trading window, or equivalently, the dynamic choice of δ and ρ. Our model assumes that the LP fixes the asymmetry ρ

of her position at time t according to (5.15), so we reduce the trading problem to a one-dimensional dynamic optimisation problem, which significantly simplifies calculations.

To further justify the form of the asymmetry (5.15), we use Uniswap v3 data to study how the asymmetry and the width of the LP's range of liquidity relate to fee revenue. First, we estimate the realised drift μ in the pool ETH/USDC over a rolling window of $T = 5$ minutes.[6] Next, for any time t, the fee income for different positions of the LP's liquidity range is computed for various values of the spread δ and for various values of the asymmetry ρ. For each value of the realised drift μ during the investment horizon, and for each fixed value of the spread δ, we record the asymmetry that maximises fee income. Figure 5.4 shows the optimal (on average) asymmetry ρ as a function of the spread δ of the position for multiple values of the realised drift μ.

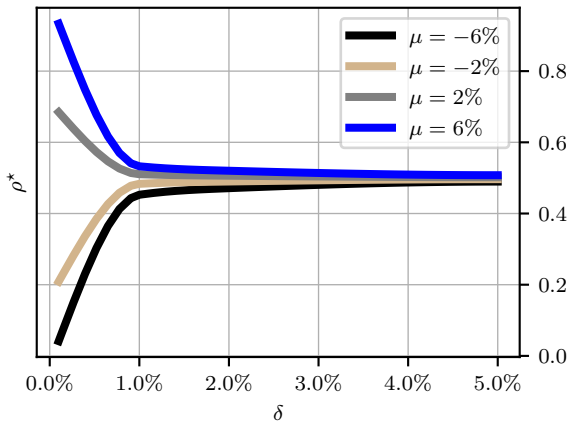


Figure 5.4: Optimal position asymmetry $\rho^\star$ in (5.14) as a function of the spread δ of the position, for multiple values of the drift μ. For each historical value of the drift in our data, we compute the performance of hypothetical liquidity positions with asymmetry values in the range $[0, 1]$ and position spreads in the range $[0.05\%, 5\%]$. The asymmetry $\rho^\star$ in the figure is the value of ρ that maximises the average fee income for each value of the pair (δ, μ). The drift is computed as the mean of 5-minute returns during one hour of trading, and the performance of the liquidity positions is computed as the total fee revenue if the position is held for an hour.

Figure 5.4 suggests that there exists a preferred asymmetry of the position for a given value of the spread δ and a given value of the drift μ. First, for all values of the spread δ, the LP skews her position to the right when the drift is positive ($\rho^\star > 0.5$) and she skews her position to the left when the drift is negative ($\rho^\star < 0.5$). Second, for narrow spreads, the liquidity position requires more asymmetry than for large spreads when the drift is not zero.

In our liquidity provision model of Section 5.4, the LP holds a belief over the future exchange rate throughout the investment window and controls the spread $\delta = \delta^U + \delta^L$ of her position. Thus, she strategically chooses the asymmetry of her position as a function of δ and μ. We approximate the relationship exhibited in Figure 5.4 with the asymmetry function (5.15).

[6] The values of the drift in this section are normalised to reflect daily estimates. In particular, we use $\mu = \tilde{\mu} \,/\, \Delta t$ where $\tilde{\mu}$ is the average of the observed log returns and Δt is the observed average LT trading frequency.

5.3.3 Rebalancing costs and gas fees

Our model considers a strategic LP who continuously repositions her liquidity position in the pool which requires rebalancing of the LP's assets. Specifically, repositioning typically leads to different holdings (5.9) in the pool, in which case the LP trades one asset for the other in the pool or in another trading venue. However, rebalancing assets to reposition liquidity is costly.

Let $(c_t)_{t\in[0,T]}$ denote the cost of rebalancing in terms of asset X. Similar to Fan et al. (2021) and Fan et al. (2022), we model rebalancing costs as proportional to the quantity $\tilde{y}_t$ of asset Y that the LP deposits in the pool. At any time t, we assume that the LP uses all her wealth V_t when repositioning her liquidity position, so we use (5.9) to write

$$c_t = -\zeta\, \tilde{y}_t\, Z_t = -\zeta\, \frac{\delta_t^U}{\delta_t^L + \delta_t^U}\, V_t\,, \quad c_0 = 0\,, \tag{5.16}$$

where $\zeta > 0$ is a constant that models the execution costs.

Transactions sent to the pool also bear gas fees. Gas fees are a flat fee paid to the blockchain and do not depend on the size of the LP's transaction; see Lı et al. (2023) for more details on the impact of gas fees on liquidity provision. Thus, gas fees scale with the frequency at which the LP sends transactions to the pool. Our model considers continuous trading, so gas fees do not influence the optimisation problem, but should be considered when the strategy is implemented.

In the next section, we derive an optimal liquidity provision strategy, and prove that the profitability of liquidity provision is subject to a tradeoff between fee revenue, PL, and concentration risk.

5.4 Optimal liquidity provision in CL pools

5.4.1 The problem

Consider an LP who provides liquidity in a CPMM with CL throughout the investment window $[0,T]$. We work on the filtered probability space $\big(\Omega, \mathcal{F}, \mathbb{P}; \mathbb{F} = (\mathcal{F}_t)_{t\in[0,T]}\big)$ where $\mathcal{F}_t$ is the natural filtration generated by the collection (Z, μ, π).

The dynamics of the LP's wealth consist of the fees earned, the position value, and the rebalancing costs. Similar to Section 5.3.1, we denote the wealth process of the LP by $(V_t = \alpha_t + p_t + c_t)_{t\in[0,T]}$, with $V_0 > 0$ known, where α is the value of the LP's position, p is the fee revenue, and c is the rebalancing cost. At any time t, the LP uses her wealth V_t to provide liquidity. Now, use (5.10), (5.13), and (5.16) to write the dynamics of the LP's wealth as

$$dV_t = V_t \left(\frac{1}{\delta_t^L + \delta_t^U}\right) \left[\left(4\,\pi_t - \frac{\sigma^2}{2} + (\mu_t - \zeta\,)\,\delta_t^U\right) dt + \sigma\,\delta_t^U\, dW_t\right] - \gamma \left(\frac{1}{\delta_t^L + \delta_t^U}\right)^2 V_t\, dt\,, \tag{5.17}$$

where $\gamma \geq 0$ is the concentration cost parameter, see Subsection 5.3.2, and π follows the dynamics defined below in (5.18). To simplify notation, we set $\zeta = 0$; our results hold when replacing μ by $\mu - \zeta$.

Next, use $\delta_t = \delta_t^U + \delta_t^L$ and $\delta_t^U \,/\, \delta_t = \rho\,(\mu_t, \delta_t)$ in (5.17) to write the dynamics of the LP's wealth as

$$dV_t = \frac{1}{\delta_t}\left(4\,\pi_t - \frac{\sigma^2}{2}\right) V_t\,dt + \mu_t\,\rho\,(\delta_t, \mu_t)\; V_t\,dt + \sigma\,\rho\,(\delta_t, \mu_t)\; V_t\,dW_t - \frac{\gamma}{\delta_t^2}\,V_t\,dt\,.$$

Next, following the discussions of Section 5.3.2, we denote by η the *profitability threshold* and we assume that the pool fee rate π follows the dynamics

$$d(\pi_t - \eta_t) = \Gamma\;(\overline{\pi} + \eta_t - \pi_t)\,dt + \psi\,\sqrt{\pi_t - \eta_t}\,dB_t\,, \tag{5.18}$$

where $\Gamma > 0$ denotes the mean reversion speed, $\overline{\pi} > 0$ is the long-term mean of $(\pi_t - \eta_t)_{t\in[0,T]}$, $\psi > 0$ is a non-negative volatility parameter, $(B_t)_{t\in[0,T]}$ is a Brownian motion independent of $(W_t)_{t\in[0,T]}$, and $\pi_0 - \eta_0 > 0$ is known. To solve the LP's optimal liquidity provision problem, we introduce the following assumption.

Assumption 1. *The profitability threshold η in the dynamics (5.18) is given by*

$$\eta_t = \frac{\sigma^2}{8} - \frac{\mu_t}{4}\left(\mu_t - \frac{\sigma^2}{2}\right) + \frac{\varepsilon}{4}\,. \tag{5.19}$$

From Assumption 1 and the CIR dynamics (5.18) it follows that

$$\pi_t - \eta_t \geq 0 \implies 4\,\pi_t - \frac{\sigma^2}{2} + \mu_t\left(\mu_t - \frac{\sigma^2}{2}\right) \geq \varepsilon > 0\,, \quad \forall t \in [0,T]\,. \tag{5.20}$$

Assumption 1 and condition (5.20) ensure that the spread δ of the optimal strategy is admissible. Financially, the inequality in (5.20) is a profitability condition $\pi_t \geq \eta_t$ that guarantees that the LP's fee income π is greater than the PL faced by the LP, adjusted by the drift in the marginal rate.

We impose specific dynamics for the fee rate π such that it satisfies a profitability condition (5.20) that allows us to obtain an admissible strategy. While we use this constraint to solve the optimal liquidity provision problem, it also represents an adequate and natural measure for LPs to assess the profitability of liquidity provision in different pools before depositing their assets.

Finally, note that the dynamics in (5.6) and (5.18) imply that the LP also observes W, B, and the profitability threshold η is determined by μ, so the LP observes all the stochastic processes of this problem.

5.4.2 The optimal strategy

The LP controls the spread δ of her position to maximise the expected utility of her terminal wealth in units of X. To define the set of admissible strategies $\mathcal{A}$, note that if the LP assumes that the asymmetry function ρ is that in (5.15), then for each δ, we need

$$\int_0^T \rho\,(\delta_t, \mu_t)^2\;dt < \infty \quad \mathbb{P}\text{–a.s.}\,. \tag{5.21}$$

Observe that

$$\int_0^T \rho\,(\delta_t, \mu_t)^2\;dt \;=\; \int_0^T \left(\frac{1}{2} + \frac{\mu_t}{\delta_t}\right)^2 dt \;\leq\; \frac{T}{4} + \frac{1}{2}\int_0^T \mu_t^4\,dt + \frac{1}{2}\int_0^T \frac{1}{\delta_t^4}\,dt\,.$$

Thus, to satisfy (5.21) and to ensure that the control problem below is well-posed, we define the set of admissible strategies

$$\mathcal{A}_t = \left\{ (\delta_s)_{s\in[t,T]},\ \mathbb{R}\text{-valued},\ \mathbb{F}\text{-adapted, and } \int_t^T \frac{1}{\delta_s^4}\, ds < +\infty\ \mathbb{P}\text{–a.s.} \right\},$$

where $\mathcal{A} := \mathcal{A}_0$.

Let $\delta \in \mathcal{A}$. The performance criterion of the LP is a function $u^\delta \colon [0,T] \times \mathbb{R}^4 \to \mathbb{R}$ given by

$$u^\delta(t,v,z,\pi,\mu) = \mathbb{E}_{t,v,z,\pi,\mu}\left[U\left(V_T^\delta\right)\right],$$

where U is a concave utility function, and the value function $u : [0,T] \times \mathbb{R}^4 \to \mathbb{R}$ of the LP is

$$u(t,v,z,\pi,\mu) = \sup_{\delta\in\mathcal{A}_t} u^\delta(t,v,z,\pi,\mu)\,. \tag{5.22}$$

The following results solve the optimal liquidity provision model when the LP assumes a general stochastic drift μ and her preferences are given by a logarithmic utility function.

Proposition 5. *Assume the asymmetry function ρ is as in (5.15) and that $U(v) = \log(v)$. Then,*

$$\begin{aligned} w\,(t,v,z,\pi,\mu) =& \log\,(v) + (\pi-\eta)^2 \int_t^T \mathbb{E}_{t,\mu}\left[\frac{8}{2\,\gamma + \mu_s^2\,\sigma^2}\right] \exp\left(-2\,\Gamma\,(s-t)\right)\, ds \\ &+ (\pi-\eta)\left(2\,\Gamma\,\overline{\pi} + \psi^2\right) \int_t^T \mathbb{E}_{t,\mu}\left[C\,(s,\mu_s)\right] \exp\left(-\Gamma\,(s-t)\right)\, ds \\ &- (\pi-\eta) \int_t^T \mathbb{E}_{t,\mu}\left[\frac{4\,\varepsilon}{2\,\gamma + \sigma^2\,\mu_s^2}\right] \exp\left(-\Gamma\,(s-t)\right)\, ds \\ &+ \int_t^T \left(\Gamma\,\overline{\pi}\,\mathbb{E}_{t,\mu}\left[E\,(s,\mu_s)\right] + \psi^2\,\mathbb{E}_{t,\mu}\left[\eta_s\,C\,(s,\mu_s)\right]\right) ds \\ &- \frac{1}{2}\int_t^T \left(\mathbb{E}_{t,\mu}\left[\frac{\varepsilon^2}{2\,\gamma + \sigma^2\,\mu_s^2} + \mu_s\right]\right) ds - \pi\frac{\sigma^2}{8}\,(T-t) \end{aligned} \tag{5.23}$$

solves the HJB equation associated with problem (5.22). Here, $\eta_s = \frac{\sigma^2}{8} - \frac{\mu_s}{4}\left(\mu_s - \frac{\sigma^2}{2}\right) + \frac{\varepsilon}{4}$ for $s \geq t$, $\eta_t = \eta$, and $\mathbb{E}_{t,\mu}$ represents expectation conditioned on $\mu_t = \mu$, and

$$C\,(t,\mu) = \mathbb{E}_{t,\mu}\left[\int_t^T \frac{8}{2\,\gamma + \mu_s^2\,\sigma^2} \exp\left(-2\,\Gamma\,(s-t)\right)\, ds\right],$$

and

$$E\,(t,\mu) = \mathbb{E}_{t,\mu}\left[\int_t^T \left(\left(2\,\Gamma\,\overline{\pi} + \psi^2\right) C\,(s,\mu) + \frac{4\,\varepsilon}{2\,\gamma + \sigma^2\,\mu_s^2}\right) \exp\left(-\Gamma\,(s-t)\right)\, ds\right].$$

For a proof, see B.1.

Theorem 4. *Let Assumption 1 hold and assume that the asymmetry function ρ is as in (5.15) and that $U(v) = \log(v)$. Then, the solution in Proposition 5 is the unique solution to the optimal control problem (5.22), and the optimal spread $(\delta_s)_{s\in[t,T]} \in \mathcal{A}_t$ is given by*

$$\delta_s^\star = \frac{2\,\gamma + \mu_s^2\,\sigma^2}{4\,\pi_s - \frac{\sigma^2}{2} + \mu_s\left(\mu_s - \frac{\sigma^2}{2}\right)} = \frac{2\,\gamma + \mu_s^2\,\sigma^2}{4\,(\pi_s - \eta_s) + \varepsilon},$$

where $\eta_s = \frac{\sigma^2}{8} - \frac{\mu_s}{4}\left(\mu_s - \frac{\sigma^2}{2}\right) + \frac{\varepsilon}{4}$.

For a proof, see B.2.

5.4.3 Discussion: profitability, PL, and concentration risk

In this section, we study the strategy when $\mu = 0$, in which case the position is symmetric, so $\rho = 1/2$ and $\delta_t^U = \delta_t^L = \delta_t/2$ and the optimal spread (3.20) becomes[7]

$$\delta_t^{L\star} = \delta_t^{U\star} = \frac{2\,\gamma}{8\,\pi_t - \sigma^2} \qquad \Longrightarrow \qquad \delta_t^\star = \frac{4\,\gamma}{8\,\pi_t - \sigma^2}, \tag{5.24}$$

so the inequality in (5.20) becomes

$$4\,\pi_t - \frac{\sigma^2}{2} \geq \varepsilon > 0\,, \quad \forall t \in [0, T]\,. \tag{5.25}$$

The inequality in (5.25) guarantees that the optimal control (5.24) does not explode, and ensures that fee income is large enough for LP activity to be profitable. In particular, it ensures that $\pi > \sigma^2/8 + \varepsilon$. When $\varepsilon \to 0$, i.e., $\sigma^2/4 \to \pi$, the spread $\delta \to +\infty$. However, we require that the spread $\delta = \delta^U + \delta^L \leq 4$, so the conditions $\delta^L \leq 2$ and $\delta^U \leq 2$ become

$$\frac{\gamma}{4\,\pi - \frac{\sigma^2}{2}} \leq 2 \implies \pi - \frac{\gamma}{8} \geq \frac{\sigma^2}{8}\,. \tag{5.26}$$

When $\delta^L = \delta^U = 2$, the LP provides liquidity in the maximum range $(0\,, +\infty)$, so the depth of her liquidity position $\tilde{\kappa}$ is minimal, and the LP's position is equivalent to providing liquidity in CPMMs without CL; see Cartea et al. (2023b) for more details.

On the other hand, when $\delta \leq 4$, the depreciation rate of the LP's position value in (5.10) is higher than $\sigma^2/8$. In particular, if $\delta = \delta^{\text{tick}}$, where δ^{tick} is the spread of a liquidity position concentrated within a single tick range, then the depth of the LP's liquidity position $\tilde{\kappa}$ is maximal and the LP's exposure to volatility is maximal.

LPs should track the profitability of the pools they consider before depositing their assets in the pool. When $\mu = 0$, we propose that LPs use $\sigma^2/8$ as a lower bound rule-of-thumb for the pool's rate of profitability because $\sigma^2/8$ is the lowest rate of depreciation of their wealth in the pool.

Condition (5.26) ensures that the profitability $\pi - \gamma/8$, which is the pool fee rate adjusted by the concentration cost, is higher than the depreciation rate of the LP's assets in the pool. Thus, the condition imposes a minimum profitability level of the pool, so LP activity is viable. An optimal control $\delta^\star > 4$ indicates non-viable LP activity because fees are not enough to compensate the losses borne by the LP. Figure 5.5 shows the estimated pool fee rate and the estimated depreciation rate in the ETH/USDC pool (from January to August 2022). In particular, the CIR model captures the dynamics of $\pi_t - \sigma^2/8$.

Next, we study the dependence of the optimal spread on the value of the concentration cost coefficient γ, the fee rate π, and the volatility σ. The concentration cost coefficient γ scales the spread linearly in (5.24). Recall that the cost term penalises small spreads because there is a risk that the rate will exit the LP's range. Thus, large values of γ generate large values of the spread. Figure 5.6 shows the optimal spread as a function of the pool fee rate π. Large potential fee income pushes the strategy towards targeting more closely the marginal rate Z to profit from fees. Lastly, Figure 5.7 shows that the optimal spread increases as the volatility of the rate Z increases.

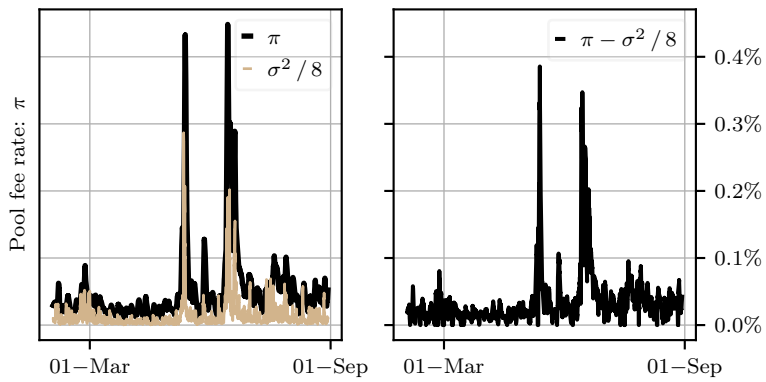


Figure 5.5: Estimated pool fee rate from February to August 2022 in the ETH/USDC pool. For any time t, the pool fee rate is the total fee income, as a percentage of the total pool size, paid by LTs in the period $[t-1 \text{ day}, t]$. The pool size at time t is $2\,\kappa\,\sqrt{Z_t}$ where Z_t is the active rate in the pool at time t.

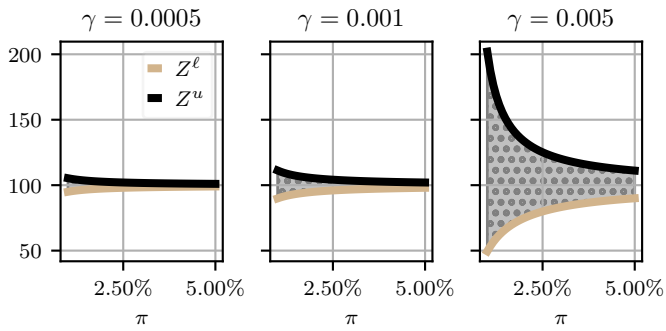


Figure 5.6: Optimal LP position range $\left(Z^L, Z^U\right]$ as a function of the pool fee rate π for different values of the concentration cost parameter γ, when $Z = 100$, $\sigma = 0.02$, and $\mu = 0$.

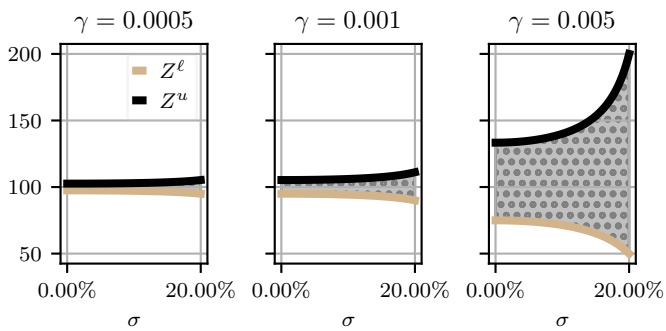


Figure 5.7: Optimal LP position range $\left(Z^L, Z^U\right]$ as a function of the volatility σ for different values of the concentration cost parameter γ, when $Z = 100$, $\pi = 0.02$, and $\mu = 0$.

Finally, the optimal spread does not depend on time or on the terminal date T. The LP marks-to-market her wealth in units of X, but does not penalise her holdings in asset Y. In particular, the LP's performance criterion does not include a running penalty or a final liquidation penalty (to turn

[7] The position range is approximately symmetric around the position rate Z because $\delta_t^U = \delta_t^L$ does not imply that $Z - Z^L = Z^U - Z$; see (5.7). However, for small values of δ^L and δ^U, one can write the approximation $Z - Z^L \approx Z^U - Z$, in which case the position is symmetric around the rate Z.

assets into cash or into the reference asset). For example, if at the end of the trading window the holdings in asset Y must be exchanged for X, then the optimal strategy would skew, throughout the trading horizon, the liquidity range to convert holdings in Y into X through LT activity.[8]

5.4.4 Discussion: drift and position skew

In this section, we study how the strategy depends on the stochastic drift μ. Use $\delta_t = \delta_t^L + \delta_t^U$ and $\rho(\delta_t, \mu_t) = \delta_t^U / \delta_t$ to write the two ends of the optimal spread as

$$\delta_t^{U\star} = \frac{2\,\gamma + \mu_t^2\,\sigma^2}{8\,\pi_t - \sigma^2 + 2\,\mu_t\left(\mu_t - \frac{\sigma^2}{2}\right)} + \mu_t \quad \text{and} \quad \delta_t^{L\star} = \frac{2\,\gamma + \mu_t^2\,\sigma^2}{8\,\pi_t - \sigma^2 + 2\,\mu_t\left(\mu_t - \frac{\sigma^2}{2}\right)} - \mu_t\,. \tag{5.27}$$

The inequality in (5.20) guarantees that the optimal control in (5.27) does not explode and ensures that fee income is large enough for LP activity to be profitable. The profitability condition in (5.26) becomes

$$\pi_t - \frac{\gamma}{8} \geq \frac{\sigma^2}{8}\left(\frac{\mu_t^2}{2} + 1\right) - \frac{\mu_t}{4}\left(\mu_t - \frac{\sigma^2}{2}\right),$$

so LPs that assume a stochastic drift in the dynamics of the exchange rate Z should use this simplified measure of the depreciation rate due to PL as a rule-of-thumb before considering depositing their assets in the pool.

Next, we study the dependence of the optimal spread on the value of the drift μ. First, recall that the controls in (5.27) must obey the inequalities[9]

$$0 < \delta_t^L \leq 2 \quad \text{and} \quad 0 \leq \delta_t^U < 2\,,$$

because $0 \leq Z^L < Z^U < \infty$ and $Z_t \in (Z_t^L, Z_t^U]$, which together with (5.8) implies $0 \leq \delta_t \leq 4$. Next, the asymmetry function satisfies

$$0 < \rho(\delta_t, \mu) = \frac{\delta_t^U}{\delta_t} < 1\,, \tag{5.28}$$

which implies

$$0 \leq \rho(\delta_t, \mu)\;\delta_t < 2 \quad \text{and} \quad 0 \leq (1 - \rho(\delta_t, \mu))\;\delta_t < 2\,. \tag{5.29}$$

Now, use (5.15) and (5.29) to write

$$0 \leq \left(\frac{1}{2} + \frac{\mu}{\delta_t}\right)\delta_t < 2 \quad \text{and} \quad 0 \leq \left(\frac{1}{2} - \frac{\mu}{\delta_t}\right)\delta_t < 2\,. \tag{5.30}$$

Finally, use (5.28) and (5.30) to obtain the inequalities

$$2\,|\mu| \leq \delta_t \leq 4 - 2\,|\mu|\,, \tag{5.31}$$

so μ must be in the range $[-1, 1]$ for the LP to provide liquidity. If μ is outside this range, concentration risk is too high so the LP must withdraw her holdings from the pool. Recall that the dynamics of Z are geometric and μ is a percentage drift, so values of μ outside the range $[-1, 1]$

[8] In LOBs, one usually assumes that final inventory is liquidated with adverse price impact and that there is a running inventory penalty, thus market making strategies in LOBs depend on the terminal date T.

[9] The admissible set of controls is not restricted to these ranges. However, values outside these range cannot be implemented in practice.

are unlikely. Moreover, when $\mu = -1$, the drift of the exchange rate Z is large and negative, so the optimal range is $(0, Z]$, i.e., the largest possible range to the left of Z. When $\mu = 1$, the drift of the exchange rate Z is large and positive, so the optimal range is $(Z, +\infty)$, which is the largest possible range to the right of Z. Condition (5.31) is always verified when we study the performance of the strategy in the ETH/USDC pool. Figure 5.8 shows how the optimal spread adjusts to the value of the drift μ. Finally, note that

$$\frac{\partial \delta^{U\star}}{\partial \sigma} = \frac{\partial \delta^{L\star}}{\partial \sigma} = \frac{2\,\mu^2\,\sigma\,(4\,\pi - 4\,\eta + \varepsilon) + 4\,\sigma\,(1+\mu)\left(2\,\gamma + \mu^2\,\sigma^2\right)}{(4\,\pi - 4\,\eta + \varepsilon)^2} > 0\,, \qquad \forall \mu \in [-1, 1]\,,$$

shows that the optimal range is strictly increasing in the volatility σ of the rate Z, which one expects as increased activity that exposes the position value to more PL, and increases the concentration risk.

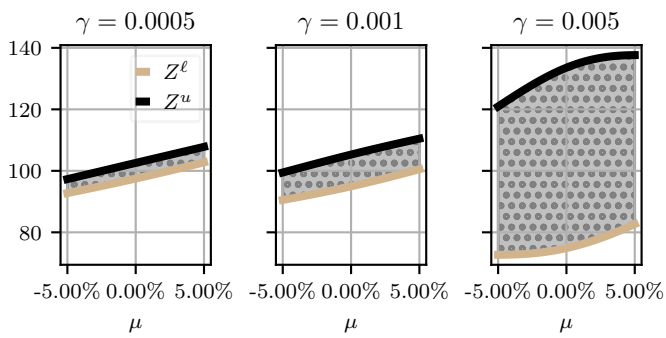


Figure 5.8: Optimal LP position range $\left(Z^L, Z^U\right]$ as a function of the drift μ for different values of the concentration cost parameter γ, when $Z = 100$, $\pi = 0.02$, and $\sigma = 0.02$.

5.5 Performance of strategy

5.5.1 Methodology

In this section, we use Uniswap v3 data between 1 January and 18 August 2022 to study the performance of the strategy of Section 5.4. We consider execution costs and discuss how gas fees and liquidity taking activity in the pool affect the performance of the strategy.[10]

Our strategy in Section 5.4 is solved in continuous time. In our performance study, we discretise the trading window in one-minute periods and the optimal spread is fixed at the beginning of each time-step. That is, let t_i be the times where the LP interacts with the pool, where $i \in \{1, \ldots, N\}$ and $t_{i+1} - t_i = 1$ minute. For each time t_i, the LP uses the optimal strategy in (5.27) based on information available at time t_i, and she fixes the optimal spread of her position throughout the period $[t_i, t_{i+1})$; recall that the optimal spread is not a function of time.

To determine the optimal spread (5.27) of the LP's position at time t_i, we use in-sample data $[t_i - 1 \text{ day}, t_i]$ to estimate the parameters. The volatility σ of the rate Z is given by the standard

[10] In practice, LPs pay gas fees when using the Ethereum network to deposit liquidity, withdraw liquidity, and adjust their holdings. Gas is paid in Ether, the native currency of the Ethereum network, and measures the computational effort of the LP operation; see Cartea et al. (2022b).

deviation of one-minute log returns of the rate Z, which is multiplied by $\sqrt{1440}$ to obtain a daily estimate. The pool fee rate π_t is given by the total fee income generated by the pool during the in-sample period, divided by the pool size $2\,\kappa\,\sqrt{Z_t}$ at time t, where κ is the active depth at time t. We run the linear regression described in Section 5.3.2.2 to estimate the concentration cost parameter, which is set to $\gamma = 5 \times 10^{-7}$. Finally, prediction of the future marginal rate Z is out of the scope of this work, thus we set $\mu = 0$ and $\rho = 0.5$.

To compute the LP's performance as a result of changes in the value of her holdings in the pool (position value), and as a result of fee income, we use out-of-sample data $[t_i, t_{i+1}]$. We use equation (5.10) to determine the one-minute out-of-sample changes in the position value. For fee income, we use LT transactions in the pool at rates included in the range $\left(Z_t^L, Z_t^U\right]$ and equation (5.5). The income from fees accumulates in a separate account in units of X with zero risk-free rate.[11] At the end of the out-of-sample window, the LP withdraws her liquidity and collects the accumulated fees, and we repeat the in-sample estimation and out-of-sample liquidity provision described above. Thus, at times t_i, where $i \in \{1, \ldots, N-1\}$, the LP consecutively withdraws and deposits liquidity in different ranges. Between two consecutive operations (i.e., reposition liquidity provision), the LP may need to take liquidity in the pool to adjust her holdings in asset X and Y. In that case, we use results in Cartea et al. (2022b) to compute execution costs.[12] In particular, we consider execution costs when the LP trades asset Y in the pool to adjust her holdings between two consecutive operations. More precisely, we consider that for every quantity y of asset Y bought or sold in the pool, a transaction cost $y\,Z_t^{3/2}/\kappa$ is incurred. We assume that the LP's taking activity does not impact the dynamics of the pool. Finally, we obtain 331,858 individual LP operations from 1 January to 18 August 2022.

5.5.2 Benchmark

We compare the performance of our strategy with the performance of LPs in the pool we consider. We select operation pairs that consist of first providing and then withdrawing the same depth of liquidity $\tilde{\kappa}$ at two different points in time by the same LP.[13] The operations that we select represent approximately 66% of all LP operations. Figure 5.9 shows the distribution of key variables that describe how LPs provide liquidity. The figure shows the distribution of the number of operations per LP, the changes in the position value, the length of time the position is held in the pool, and the position spread.

Finally, Table 5.2 shows the average and standard deviation of the distributions in Figure 5.9. Notice that the bulk of liquidity is deposited in small ranges, and positions are held for short periods of time; 20% of LP positions are held for less than five minutes and 30% for less than one hour. Table 5.2 also shows that, on average, the performance of the LP operations in the pool and the period we consider is $-1.49\,\%$ per operation.

[11] In practice, fees accumulate in both assets X and Y.

[12] Cartea et al. (2022b) show that execution costs in the pool are a closed-form function of the rate Z, the pool depth κ, and the transaction size.

[13] In blockchain data, every transaction is associated to a unique wallet address.

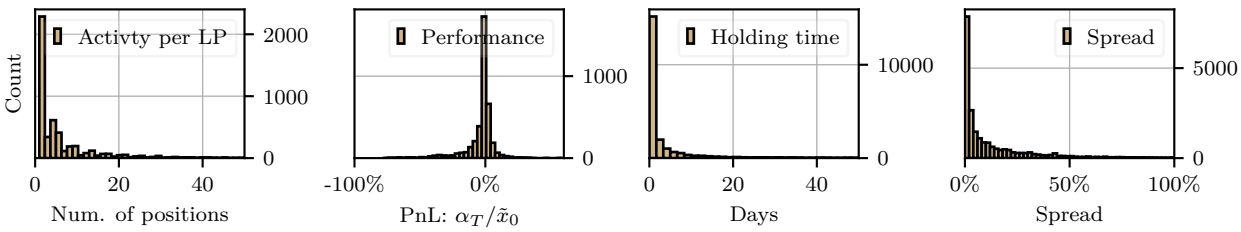


Figure 5.9: From left to right: distribution of the number of operations per LP, changes in the holdings value as a percentage of initial wealth, position hold time, and position spread. ETH/USDC pool with selected operations from 5,156 LPs between 5 May 2021 and 18 August 2022.

	Number of transactions per LP	Position value performance ($\alpha_T/V_0 - 1$)	Fee income ($p_T/V_0 - 1$)	Hold time	Spread
Average	11.5	−1.64%	0.155%	6.1 days	18.7%
Standard deviation	40.2	7.5%	0.274%	22.4 days	43.2%

Table 5.2: LP operations statistics in the ETH/USDC pool using operation data of 5,156 different LPs between 5 May 2021 and 18 August 2022. Performance includes transaction fees and excludes gas fees. The position value performance and the fee income are not normalised by the hold time.

5.5.3 Performance results

This subsection focuses on the performance of our strategy when gas fees are zero — at the end of the section we discuss the profitability of the strategy when gas fees are included. Figure 5.10 shows the distribution of the optimal spread (5.24) posted by the LP. The bulk of liquidity is deposited in ranges with a spread δ below 1%. Table 5.2 compares the average performance of the components of the optimal strategy with the performance of LP operations observed in the ETH/USDC pool.[14] Table 5.3 suggests that the position of the LP loses value in the pool (on average) because of PL; however, the fee income would cover the loss, on average, if one assumes that gas fees are zero. Finally, the results show that our optimal strategy significantly improves the PnL of LP activity in the pool and the performance of the assets themselves.

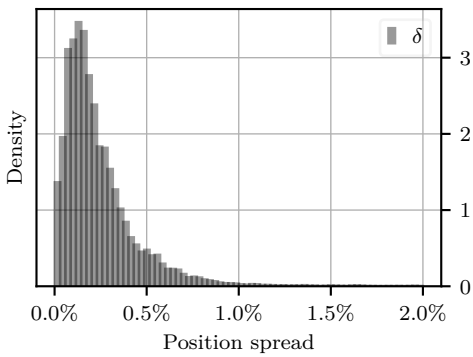


Figure 5.10: Distribution of the position spread δ.

[14] In particular, performance is given for the selected operations shown in Figure 5.9.

	Position value performance per operation	Fee income per operation	Total performance per operation (with transaction costs, without gas fees)
Optimal strategy	−0.015% (0.0951%)	0.0197% (0.005%)	0.0047% (0.02%)
Market	−0.0024% (0.02%)	0.0017% (0.005%)	−0.00067% (0.02%)
Hold	n.a.	n.a.	−0.00016% (0.08%)

Table 5.3: **Optimal strategy**: Mean and standard deviation of the one-minute performance of the LP strategy (5.24) and its components. **Market**: Mean and standard deviation of one-minute performance of LP activity in the ETH/USDC pool using data between 1 January and 18 August 2022. **Hold**: Mean and standard deviation of the one-minute performance of holding the assets. In all cases, the performance includes transaction costs (pool fee and execution cost), but does not include gas fees.

The results in Table 5.3 do not consider gas fees. Gas cost is a flat fee, so it does not depend on the position spread or size of transaction. If the activity of the LP does not affect the pool and if the fees collected scale with the wealth that the LP deposits in the pool, then the LP should consider an initial amount of X and Y that would yield enough fees to cover the flat gas fees. An estimate of the average gas cost gives an estimate of the minimum amount of initial wealth for a self-financing strategy to be profitable. Recall that, at any point in time t, the LP withdraws her liquidity, adjusts her holdings, and then deposits new liquidity. In the data we consider, the average gas fee is 30.7 USD to provide liquidity, 24.5 USD to withdraw liquidity, and 29.6 USD to take liquidity. Average gas costs are obtained from blockchain data which record the gas used for every transaction, and from historical gas prices. The LP pays a flat fee of 84.8 USD per operation when implementing the strategy in the pool we consider, so the LP strategy is profitable, on average, if the initial wealth deposited in the pool is greater than 1.8×10^6 USD.

Fee income of the LP strategy is limited by the volume of liquidity taking activity in the pool, so one should not only consider increasing the initial wealth to make the strategy more profitable. There are 4.7 LT transactions per minute and the average volume of LT transactions is $477,275$ USD per minute, so if the LP were to provide 100% of the liquidity, the average fee income per operation would be 1,431 USD.

5.6 Discussion: modelling assumptions

This section summarises our modelling assumptions and discusses their implications, strengths, and weaknesses.

Continuous trading. Our model assumes continuous repositioning of the LP's position. However, when interacting with blockchains, updates occur at the block validation frequency. For instance, the Ethereum network's blocks are validated every 13 seconds, on average. Moreover,

within each block, the transactions form a (random) queue that determines the priority with which they are executed. Our model can be extended to include delays inherent to blockchains and we refer the reader to the work in Cartea and Sánchez-Betancourt (2021) and Cartea et al. (2021).

Rebalancing. Continuous repositioning requires rebalancing the LP's assets which incurs costs as discussed in Section 5.3.3. To model this aspect, we assume that the LP pays costs that are proportional to the quantity of asset Y held in the pool. In practice, the exact costs depend on variations in the holdings between two consecutive liquidity positions. Thus, liquidity provision strategies should balance large variations in the holdings with fee revenue, PL, and concentration cost. However, the nonlinearity in the CL constant product formula complicates the mathematical modelling of this aspect of trading costs. Moreover, rebalancing costs depend on the cost structure of the trading venue where rebalancing trades are executed.

Gas fees. Our model assumes that gas fees paid by the LP to interact with the blockchain are flat and constant. In practice, gas fees are stochastic and depend on exogenous factors such as the price of electricity and network congestion. These could be included in our model by considering a stochastic gas fee that is observed by the LP.

Concentration costs. The specific microstructure of CL markets features a new type of investment risk which we refer to as concentration risk. To capture the losses due to concentration risk in a continuous-time framework, we introduced an instantaneous cost which is inversely proportional to the spread of the LP's position, and we showed that it captures the losses due to concentration risk accurately. In practice, LPs must tailor the estimation of the concentration cost parameter γ in (5.13) to the rebalancing frequency and to the volatility of the marginal rate Z.

Asymmetry. Our model assumes a fixed relation between the asymmetry of the LP's position and the drift in the marginal rate. This relation fits observed data but also leads the LP to skew the position to capture more LT trades and to profit from expected rate movements. Future work will consider a richer characterization of the asymmetry because it may be desirable for LPs to adjust the asymmetry of their position as a function of other state variables or as a controlled variable

Fee dynamics. Our model assumes that the distribution of fee revenue π among LPs is stochastic and proportional to the size of the pool to reflect that large pools attract more trading flow because trading is cheaper. We also make the simplifying assumption that fees are uncorrelated to the price. In practice, the dynamics of fee revenue may be correlated to those of the volatility of the rate, which is also related to the concentration cost parameter γ. Future work will consider more complex relations among these variables.

Constant volatility. At present, CFMMs with CL mainly serve as trading venues for crypto-assets which are better described with a diffusion model with stochastic volatility. It is straightforward to extend our strategy to this type of models.

Market impact. Finally, our analysis does not take into account the impact of liquidity provision on liquidity taking activity, however, we expect liquidity provision in CPMMs with CL to be profitable in pools where the volatility of the marginal rate is low, where liquidity taking activity is high, and when the gas fee cost to interact with the liquidity pools is low. These conditions ensure that the fees paid to LPs in the pool, adjusted by gas fees and concentration cost, exceed PL so liquidity provision is viable.

5.6.1 Discussion: related work

Closest to this work are the strategic liquidity provision models proposed in Fan et al. (2021) and Fan et al. (2022) which also consider CL markets. Both models allow LPs to compute liquidity positions over different intervals centered around the marginal rate within a given time horizon; Fan et al. (2021) only consider static LP strategies which do not use of reallocations, and Fan et al. (2022) reposition liquidity whenever the price is outside of the position range. Both approaches only focus on maximising fee revenue and rely on approximations of the LP's objective and use Neural Networks to obtain context-aware approximate strategies. In contrast, our model leads to closed-form formulae that explicitly balance fee revenue with concentration risk, fee revenue, and rebalancing costs, while allowing LPs to use price signals (potentially based on exogenous information) to improve trading performance.

5.7 Conclusions

We studied the dynamics of the wealth of an LP in a CPMM with CL who implements a self-financing strategy that dynamically adjusts the range of liquidity. The wealth of the LP consists of the position value and fee revenue. We showed that the position value depreciates due to PL and the LP widens her liquidity range to minimise her exposure to PL. On the other hand, the fee revenue is higher for narrow ranges, but narrow ranges also increase concentration risk.

We derived the optimal strategy to provide liquidity in a CPMM with CL when the LP maximises expected utility of terminal wealth. This strategy is found in closed-form for log-utility of wealth, and it shows that liquidity provision is subject to a profitability condition. In particular, the potential gains from fees, net of gas fees and concentration costs, must exceed PL. Our model shows that the LP strategically adjusts the spread of her position around the reference exchange rate; the spread depends on various market features including tthe volatility of the rate, the liquidity taking activity in the pool, and the drift of the rate.

Chapter 6

Deep Liquidity Provision

6.1 Introduction

Constant function market makers (CFMMs) are trading venues where the takers and providers of liquidity interact in liquidity pools; liquidity providers (LPs) deposit their assets in the liquidity pool and liquidity takers (LTs) exchange assets directly with the pool. At present, constant product market makers (CPMMs) with concentrated liquidity (CL) are the most popular type of CFMM, with Uniswap v3 as a prime example; see Adams et al. (2021). In CPMMs with CL, LPs specify the rate ranges over which they deposit their assets, and this liquidity is counterparty to trades of LTs when the marginal exchange rate of the pool is within the liquidity range of the LPs. When LPs deposit liquidity, fees paid by LTs accrue and are paid to LPs when they withdraw their assets from the pool. The amount of fees accrued to LPs is proportional to the share of liquidity they hold in each liquidity range of the pool.

Current research characterises the losses of LPs and offers tools for strategic liquidity provision in a single liquidity pool with CL; see Cartea et al. (2022c, 2023b). In this chapter, we study strategic liquidity provision in multiple liquidity pools with CL. Here, each pool trades a risky asset and a common reference asset used by the LP to value her wealth.

We derive the discrete-time dynamics of the wealth of a strategic LP who dynamically provides liquidity in the various pools. At each time step, the LP specifies the proportion of her wealth to deposit in each pool and the range where she wants to provide liquidity. Her wealth consists of the position she holds in each pool and fee income.

Here, we do not specify a model for the stochastic processes observed by the LP. Instead, we use a model-free approach to solve the liquidity provision problem. Specifically, we approximate the optimal liquidity provision strategy with a long short-term memory (LSTM) network, whose output at each time step determines how the LP spreads her wealth across the pools and the liquidity provision range in each pool. The LP use Sharpe Ratio and mean-variance criterion to train the LSTM strategy on simulated data and market data.

We test the LSTM strategy on simulated data to showcase the performance of the strategy under different market conditions. Firstly, we consider an LP who provides liquidity in a single liquidity pool with CL and we recover results from the literature; see Cartea et al. (2022c). Specifically, when volatility is high, regardless of the performance criterion used, the LSTM strategy provides

liquidity in wide ranges of the pool, as in see Cartea et al. (2023b). The optimal strategy increases the range posted by the LP as the fee tier decreases, and the LP skews her position to align provision of liquidity with the trend of the pools rate. Secondly, we consider an LP who provides liquidity in two liquidity pools with CL. The behaviour of the LSTM strategy in each individual pool is qualitatively the same as that in the single pool case, and the LP provides more liquidity in pools with lower volatility, higher fee tier, and where the rate between the risky asset and the reference asset trends upwards.

Finally, we use Uniswap v3 and Binance data to test the LSTM strategy. We use LP and LT data from the two trading venues between 1 July 2021 and 30 September 2023 for the pair ETH/USDC and the pair BTC/USDC. ETH is the ticker of *Ether*, BTC is the ticker of *Bitcoin* and USDC is the ticker of *USD coin*, which is a digital currency fully backed by U.S. Dollars (USD). The input to the LSTM strategy is a set of features which we build with data from both venues. Overall, we find that gas fees have a considerable impact on the performance of the strategy. Specifically, the average returns of the LSTM strategy before gas fees decrease with the LP's initial wealth and increase with the LP's trading frequency; in contrast, the average returns of the strategy after gas fees increase with the LP's initial wealth and decrease with the LP's trading frequency.

Early works on automated market makers (AMMs) include Chiu and Koeppl (2019); Angeris et al. (2021); Lipton and Treccani (2021). Cartea et al. (2023b) study the losses faced by LPs in CFMMs and then introduce predictable loss, which measures the unhedgeable losses of LPs stemming from the depreciation of their holdings in the pool and from the opportunity costs from locking their assets in the pool. Predictable loss is similar to loss-versus-rebalancing in Milionis et al. (2022) which describes the unhedgeable losses of LPs in traditional CFMMs due to losses to arbitrageurs.[1] Previous works on strategic liquidity provision include Heimbach et al. (2022) which discusses the tradeoff between risks and returns that LPs face in Uniswap v3. Cartea et al. (2022c) introduces a continuous-time model for optimal liquidity provision, Milionis et al. (2023) study the impact of fees on the profits of arbitrageurs in CFMMs, Fukasawa et al. (2023) study the hedging of the impermanent losses of LPs, and Lı et al. (2023) studies the economics of liquidity provision. Close to our work are the models in Fan et al. (2021) and Fan et al. (2022) which focus on fee revenue and use approximation techniques to obtain dynamic strategies for liquidity provision. Finally, there is a growing literature on AMM design for fair competition between LPs and LTs. Goyal et al. (2023) study an AMM with dynamic trading functions that incorporate beliefs of LPs, Lommers et al. (2023) studies AMMs where the LP's strategy adjusts dynamically to market information, and Cartea et al. (2023d) generalises CFMMs and propose AMM designs where LPs express their beliefs and risk preferences.

Our work is related to the algorithmic trading and optimal market making literature. Early works on liquidity provision in traditional markets are Ho and Stoll (1983), Biais (1993), and

[1] Milionis et al. (2022) introduced loss-versus-rebalancing in August 2022 as a measure that quantifies the unhedgeable losses of LPs in CFMMs to arbitrageurs. Contemporaneously, Cartea et al. (2023b) introduced predictable loss in November 2022 as a component in the wealth of strategic LPs that quantifies predictable loss due to the convexity of the trading function and due to opportunity costs in CFMMs and in CL markets.

Avellaneda and Stoikov (2008) with extensions in many directions; see Cartea et al. (2014, 2017); Guéant (2017); Bergault et al. (2021); Drissi (2022). We refer the reader to Cartea et al. (2015), Guéant (2016), and Donnelly (2022) for a comprehensive review of algorithmic trading models for takers and makers of liquidity in traditional markets.

LSTM neural networks are a type of neural network typically employed for sequential data; see Hochreiter and Schmidhuber (1997). In financial applications, LSTM networks are typically used for unsupervised learning tasks; see Nelson et al. (2017); Fischer and Krauss (2018); Zhang et al. (2019); Moghar and Hamiche (2020). Here, we employ LSTM networks to approximate an optimal strategy which maximises a performance criterion. Close to our work are Buehler et al. (2019); Becker et al. (2020); Limmer and Horvath (2023) where the authors train neural networks on market data to hedge a basket of derivatives. Also, our work is related to the reinforcement learning and market making literature; see Chan and Shelton (2001); Balch et al. (2019); Gašperov and Kostanjčar (2021); Jerome et al. (2023).

The remainder of the chapter proceeds as follows. Section 6.2 describes CL pools and Section 6.3 studies the discrete-time dynamics of the wealth of LPs who provides liquidity in a CPMM with N pools with CL. Finally, Section 6.4 introduces the framework for the model-free optimisation, Section 6.5 tests the optimal strategy on synthetic data, and Section 6.6 tests the strategy on market data.

6.2 Automated Market Making

In this section, we focus on CPMMs with CL and describe the revenue and the risks of liquidity provision in these venues.

6.2.1 Constant product market makers

Traditional electronic exchanges are organised around limit order books where liquidity taking and liquidity provision orders are matched. AMMs are a new type of trading venues where takers and providers of liquidity interact through liquidity pooling. LPs deposit assets in a common liquidity pool, and LTs trade directly with the pool. The liquidity pool consists of quantity $x > 0$ of asset X and quantity $y > 0$ of asset Y. Asset Y is valued in terms of the reference asset X, and Z denotes the (exchange) rate in the pool. CPMMs are a particular type of AMMs and they impose an LT trading condition and an LP trading condition, both of which define the state of the pool after an LT transaction and after an LP operation is completed.

LT trading condition Let (x, y) be the state of the pool before the arrival of an LT buy order for a quantity $\Delta y > 0$ of asset Y. The quantity $\Delta x > 0$ of asset X that the LT pays to the pool, in exchange for $\Delta y > 0$, is determined by the LT trading condition

$$x\,y = (x + (1-\tau)\,\Delta x)\,(y - \Delta\Delta y) = \kappa^2\,, \tag{6.1}$$

where $\kappa > 0$ is the *depth* of the pool and $\tau \geq 0$ is the *fee tier*. For a sell order, the LT trading condition is

$$x\,y = (x - \Delta x)\,(y + (1-\tau)\,\Delta y) = \kappa^2\,.$$

Chapter 3 shows that the instantaneous exchange rate is given by

$$Z = \frac{x}{y}\,,$$

and that the execution rate for an LT trade of quantity y is

$$\tilde{Z}\,(\Delta y) = Z - Z^{3/2}\,\Delta y\,/\,\kappa\,.$$

LP trading condition LPs deposit assets in the pool or withdraw assets from the pool. The LP trading condition requires that LP operations do not impact the rate Z of the pool. However, LP operations do change the depth κ of the pool. More precisely, let (x, y) be the state of the pool before an LP deposits liquidity (x, y),[2] so $(x + x, y + y)$ is the state of the pool after the deposit, which increases the depth of the pool because

$$(x + \Delta x)\,(y + \Delta y) = \overline{\kappa}^2 > x\,y = \kappa^2\,.$$

The LP trading condition requires that when an LP deposits the quantities $(\Delta x, \Delta y)$, then the following equality must hold

$$Z = \frac{x}{y} = \frac{x + \Delta x}{y + \Delta y}\,. \tag{6.2}$$

In practice, LPs usually invest an initial wealth V in terms of the reference asset X, so the amounts x and y are obtained with the LP trading condition and the equality $V = \Delta x + Z\,\Delta y$. In contrast, LTs usually specify a quantity x to trade, and receive or pay a quantity y which is determined by the LT trading condition.

6.2.2 Concentrated liquidity

In CPMMs without CL, liquidity in the pool is spread over all rates, and the LP trading condition ensures that the quantities (x, y) deposited or withdrawn by LPs verify $(\Delta x, \Delta y) = (\rho\,x, \rho\,y)$ where $\rho \geq 0$; see Cartea et al. (2022b) and Cartea et al. (2023a). In pools with CL, the space of rates where LPs provide liquidity is discretised into a set of values $\{\mathcal{Z}(i)\}_{i\in\mathbb{Z}}$ called ticks.[3] Each tick $\mathcal{Z}(i)$ is uniquely identified by an integer $i \in \mathbb{Z}$ according to

$$\mathcal{Z}(i) = 1.0001^i\,. \tag{6.3}$$

The boundaries of an LP's position in the pool take values in the set of ticks. The range between two consecutive ticks defines the smallest range available for LPs. Every tick range $(\mathcal{Z}(i), \mathcal{Z}(i+1)]$ has its own depth which determines the execution cost of LT trades when the instantaneous rate Z

[2] Here, $x \geq 0$ and $y \geq 0$.
[3] In contrast, a tick in LOBs refers to the smallest possible price increment.

is in that tick range. Thus, the depth κ can change as a result of the instantaneous rate Z crossing the boundary of a tick range.

LPs specify a lower tick $\mathcal{Z}(L)$ and an upper tick $\mathcal{Z}(U)$ between which they provide liquidity that is evenly spread over all the tick ranges in the position range $(\mathcal{Z}(L), \mathcal{Z}(U)]$. The position of an LP who provides liquidity in a range $(\mathcal{Z}(L), \mathcal{Z}(U)]$ is characterised by the position depth $\tilde{\kappa}^{L,U}$. The value of $\tilde{\kappa}^{L,U}$ represents the depth of the LP's liquidity within every tick range in her position range, and $\tilde{\kappa}^{L,U}/\kappa^i$ represents the portion of liquidity that the LP holds in the tick range $(\mathcal{Z}(i), \mathcal{Z}(i+1)] \subset (\mathcal{Z}(L), \mathcal{Z}(U)]$.

The following key formulae define the LP trading condition in CPMMs with CL, i.e., they give the quantities (x, y) that the LP must deposit in or withdraw from the range $(\mathcal{Z}(L), \mathcal{Z}(U)]$ so that the equality in (6.2) holds

$$\begin{cases} x = 0 & \text{and} \quad y = \tilde{\kappa}^{L,U}\left(\frac{1}{\sqrt{\mathcal{Z}(L)}} - \frac{1}{\sqrt{\mathcal{Z}(U)}}\right) & \text{if} \quad Z < \mathcal{Z}(L)\,, \\ x = \tilde{\kappa}^{L,U}\left(\sqrt{Z} - \sqrt{\mathcal{Z}(L)}\right) & \text{and} \quad y = \tilde{\kappa}^{L,U}\left(\frac{1}{\sqrt{Z}} - \frac{1}{\sqrt{\mathcal{Z}(U)}}\right) & \text{if} \quad \mathcal{Z}(L) \leq Z < \mathcal{Z}(U)\,, \\ x = \tilde{\kappa}^{L,U}\left(\sqrt{\mathcal{Z}(U)} - \sqrt{\mathcal{Z}(L)}\right) & \text{and} \quad y = 0 & \text{if} \quad Z \geq \mathcal{Z}(U)\,, \end{cases} \tag{6.4}$$

and the depth available in the tick range $(\mathcal{Z}(i), \mathcal{Z}(i+1)]$ is given by the sum of the depth of the positions of all LPs who provide liquidity in that tick range. We write

$$\kappa^i = \sum_{j \in \{1, \cdots, M\}} \tilde{\kappa}^{L,U,j}\, \mathbb{1}_{\{\mathcal{Z}(L_j) \leq \mathcal{Z}(i) < \mathcal{Z}(U_j)\}}\,,$$

where $\tilde{\kappa}^{L,U,j}$ is the liquidity provided by the jth LP in the range $(\mathcal{Z}(L_j), \mathcal{Z}(U_j)]$, and $\mathbb{1}$ is the indicator function.

When the rate Z is in a tick range, fee revenue is distributed to LPs who provide liquidity in that tick range, i.e., for a fee p paid by an LT, the jth LP with position depth $\tilde{\kappa}^{L,U,j}$ earns[4]

$$\frac{\tilde{\kappa}^{L,U,j}}{\kappa}\, p\, \mathbb{1}_{\{\mathcal{Z}(L_j) < Z \leq \mathcal{Z}(U_j)\}}\,. \tag{6.5}$$

6.3 Optimal liquidity provision

In this section, we formalise the problem of an LP who provides liquidity in $n = 1, \ldots, N$ pools. Each pool provides liquidity in the reference asset X and a risky asset Y_n. We denote by $Z^1, \ldots, Z^N$ the marginal rates in these pools.

We fix a filtered probability space $\left(\Omega, \mathcal{F}, \mathbb{P}, \mathbb{F} = \{\mathcal{F}_t\}_{t=0,\ldots,T}\right)$ satisfying the usual conditions. Here, $\mathbb{F}$ is the natural filtration generated by the $\mathbb{R}^D$-valued stochastic process $\{\mathfrak{I}_t\}_{t=0,\ldots,T}$ that represents the information available to the LP at each time. Finally, we assume that each stochastic process defined below is adapted to $\mathbb{F}$.

For each $n \in \{1, \ldots, N\}$, the marginal rate of the pool to trade X and Y_n is the $\mathbb{R}_+$-valued stochastic process $\{Z_t^n\}_{t=0,\ldots,T}$. The index of the active tick in the pool is the $\mathbb{Z}$-valued stochastic

[4] If the volume of the LT trade makes the rate Z exit the tick range, then the transaction is subdivided into sub-transactions which get executed with different values of the depth.

process $\{i_t^n\}_{t=0,\dots,T}$, where n is the pool, t is the time, and i is the tick index of the pool as defined in (6.3). In particular, for each $t \in \{0,\dots,T\}$ the rate

$$Z_t^n \in \left(\mathcal{Z}\left(i_t^n\right), \mathcal{Z}\left(i_t^n+1\right)\right] .$$

Similarly, the market depth is the $\mathbb{R}_+$-valued stochastic process $\{\kappa_t^n\}_{t=0,\dots,T}$, where n is the pool, t is the time and κ is the market depth as defined in (6.1).

6.3.1 Stochastic control problem

The LP provides liquidity in the N pools between times 0 and T, and her wealth, in units of asset X, is the $\mathbb{R}_+$-valued stochastic process $\{V_t\}_{t=0,\dots,T}$. She provides liquidity in the CPMM by readjusting her position every Δt time-steps, where $\Delta t \in \{1,2,3,\dots\}$ is fixed. We denote by $\mathbb{T}$ the set of times at which the LP readjusts her position, which is given by

$$\mathbb{T} := \{t = k\,\Delta t \,|\, k \in \{0,1,2,\dots\} \text{ and } k\,\Delta t < T\} .$$

For each $t \in \mathbb{T}$, to optimally provide liquidity between t and $t+\Delta t$, the LP chooses optimal controls $\boldsymbol{w}_t = \left(w_t^1,\dots,w_t^N\right)$ such that

$$\sum_{n=1}^{N} w_t^n = 1 \quad \text{and} \quad w_t^n \geq 0\,,\ \forall\, n \in \{1,\dots,N\}\,. \tag{6.6}$$

Here, w_t^n represents the proportion of wealth that she deposits into the pool that trades assets X and Y_n. Simultaneously, the LP chooses optimal controls $\boldsymbol{\ell}_t = \left(\ell_t^1,\dots,\ell_t^N\right) \in \{0,\dots,\mathcal{T}\}^N$ and $\boldsymbol{u}_t = \left(u_t^1,\dots,u_t^N\right) \in \{0,\dots,\mathcal{T}\}^N$, where $\mathcal{T}$ is a fixed positive integer. For each n, the quantity ℓ_t^n is the *lower spread*, and $\mathcal{Z}(i_t^n - \ell_t^n)$ is the lower tick of the LP's range. Similarly, the quantity u_t^n is the *upper spread*, and $\mathcal{Z}(i_t^n + u_t^n + 1)$ is the upper tick of the LP's range. Next, for each n the LP deposits a fraction w_t^n of her wealth into the range $(\mathcal{Z}(i_t^n - \ell_t^n), \mathcal{Z}(i_t^n + u_t^n + 1)]$ of the pool that trades assets X and Y_n. The quantity $\ell_t^n + u_t^n$ is the *spread* of the strategy in the pool of assets X and Y_n at time t and we have

$$Z_t^n \in (\mathcal{Z}(i_t^n - \ell_t^n), \mathcal{Z}(i_t^n + u_t^n + 1)]$$

because ℓ_t^n, $u_t^n \geq 0$.

A liquidity provision strategy is a sequence $\{(\boldsymbol{w}_t, \boldsymbol{\ell}_t, \boldsymbol{u}_t)\}_{t\in\mathbb{T}}$ which specifies how the LP provides liquidity in the N pools of the CPMM between time 0 and T. The set of admissible liquidity provision strategies is given by

$$\mathcal{A} = \left\{ \{(\boldsymbol{w}_t, \boldsymbol{\ell}_t, \boldsymbol{u}_t)\}_{t\in\mathbb{T}} \;\middle|\; \begin{array}{c} \mathbb{R}_+^N \times \{0,\dots,\mathcal{T}\}^N \times \{0,\dots,\mathcal{T}\}^N\text{-valued,} \\ \mathbb{F}\text{-adapted, and } \boldsymbol{w_t} \text{ satisfies (6.6)} \end{array} \right\} .$$

6.3.2 Liquidity provision

At time $t = 0$, the LP chooses a control $\boldsymbol{w}_0$ which determines the proportion of her wealth that she allocates to each pool. Simultaneously, she chooses a pair of controls $\boldsymbol{\ell}_0, \boldsymbol{u}_0$ which specify the range where she provides liquidity between time 0 and Δt. For pool n, the quantities w_0^n, $\mathcal{Z}(i_0^n - \ell_0^n)$, and $\mathcal{Z}(i_0^n + u_0^n + 1)$, determine the LP's market depth $\tilde{\kappa}_0^n$, which is given by

$$\tilde{\kappa}_0^n = w_0^n \, V_0 \times \left(2\sqrt{Z_0^n} - \sqrt{\mathcal{Z}(i_0^n - \ell_0^n)} - \frac{Z_0^n}{\sqrt{\mathcal{Z}(i_0^n + u_0^n + 1)}} \right)^{-1}, \tag{6.7}$$

see Adams et al. (2021) for more details. The quantities $(\tilde{x}_0^n, \tilde{y}_0^n)$ of assets X and Y that the LP deposits in the liquidity pool are determined by $\mathcal{Z}(i_0^n - \ell_0^n)$, $\mathcal{Z}(i_0^n + u_0^n + 1)$, Z_0^n, and $\tilde{\kappa}_0^n$. Specifically, we have

$$\begin{cases} \tilde{x}_0^n = \tilde{\kappa}_0^n \left(\sqrt{Z_0^n} - \sqrt{\mathcal{Z}(i_0^n - \ell_0^n)} \right), \\ \tilde{y}_0^n = \tilde{\kappa}_0^n \left(\dfrac{1}{\sqrt{Z_0^n}} - \dfrac{1}{\sqrt{\mathcal{Z}(i_0^n + u_0^n + 1)}} \right), \end{cases}$$

where $\tilde{x}_0^n + \tilde{y}_0^n \, Z_0^n = w_0 \, V_0$.

For simplicity, we assume that before providing liquidity in the pool, the LP trades in an alternative trading venue to obtain the exact amounts of asset X and assets $Y_1, \ldots, Y_N$ that she will deposit in the N pools of the CPMM, we also assume that the LP does not pay transaction fees and trades at the instantaneous rate $Z_0^1, \ldots, Z_0^N$.

Between time 0 and Δt, the instantaneous rate in the nth pool moves from Z_0^n to $Z_{\Delta t}^n$ and the LP's quantities of assets X and Y_n in the CPMM change. We denote by $(\mathring{x}_{\Delta t}^n, \mathring{y}_{\Delta t}^n)$ the quantities of assets X and Y_n held by the LP in the pool at the beginning of time Δt. In this way, we distinguish these quantities from $(\tilde{x}_{\Delta t}^n, \tilde{y}_{\Delta t}^n)$, which denote the LP's holdings in the same pool after choosing new controls $w_{\Delta t}^n$, $\ell_{\Delta t}^n$, and $u_{\Delta t}^n$. In particular, the quantities $\mathring{x}_{\Delta t}^n$ and $\mathring{y}_{\Delta t}^n$ are given by

$$\mathring{x}_{\Delta t}^n = \begin{cases} 0 & \text{if} \quad Z_{\Delta t}^n < \mathcal{Z}(i_0^n - \ell_0^n), \\ \tilde{\kappa}_0^n \left(\sqrt{Z_{\Delta t}^n} - \sqrt{\mathcal{Z}(i_0^n - \ell_0^n)} \right) & \text{if} \quad \mathcal{Z}(i_0^n - \ell_0^n) \le Z_{\Delta t}^n < \mathcal{Z}(i_0^n + u_0^n + 1), \\ \tilde{\kappa}_0^n \left(\sqrt{\mathcal{Z}(i_0^n + u_0^n + 1)} - \sqrt{\mathcal{Z}(i_0^n - \ell_0^n)} \right) & \text{if} \quad Z_{\Delta t}^n \ge \mathcal{Z}(i_0^n + u_0^n + 1), \end{cases}$$

$$\mathring{y}_{\Delta t}^n = \begin{cases} \tilde{\kappa}_0^n \left(\dfrac{1}{\sqrt{\mathcal{Z}(i_0^n - \ell_0^n)}} - \dfrac{1}{\sqrt{\mathcal{Z}(i_0^n + u_0^n + 1)}} \right) & \text{if} \quad Z_{\Delta t}^n < \mathcal{Z}(i_0^n - \ell_0^n), \\ \tilde{\kappa}_0^n \left(\dfrac{1}{\sqrt{Z_{\Delta t}^n}} - \dfrac{1}{\sqrt{\mathcal{Z}(i_0^n + u_0^n + 1)}} \right) & \text{if} \quad \mathcal{Z}(i_0^n - \ell_0^n) \le Z_{\Delta t}^n < \mathcal{Z}(i_0^n + u_0^n + 1), \\ 0 & \text{if} \quad Z_{\Delta t}^n \ge \mathcal{Z}(i_0^n + u_0^n + 1), \end{cases}$$

see Adams et al. (2021) for more details.

Between time 0 and Δt, the LP's liquidity deposited in the pool that trades assets X and Y_n generates $\Phi_{0,\Delta t}^n$, in units of X, and more generally, $\Phi_{s,t}^n$ denotes the amount of fees that the LP's position generates between time s and t, also in units of X. Thus, the mark-to-market value of the

LP's wealth at time Δt is

$$V_{\Delta t} = \sum_{n=1}^{N} \mathring{x}^n_{\Delta t} + \mathring{y}^n_{\Delta t}\, Z^n_{\Delta t} + \Phi^n_{0,\Delta t}\,.$$

Next, the LP withdraws her assets from the N pools and reinvests all her wealth $V_{\Delta t}$ to provide liquidity between time Δt and time $2\Delta t$.

In general, for each $t \in \mathbb{T}$ the mark-to-market value of the LP's wealth is equal to

$$V_t = \sum_{n=1}^{N} \mathring{x}^n_t + \mathring{y}^n_t\, Z^n_t + \Phi^n_{t-\Delta t,t}$$

units of X. Here, $\mathring{x}^n_t + \mathring{y}^n_t\, Z^n_t$ is the mark-to-market value at the beginning of time t of the LP's assets deposited in the pool that trades assets X and Y_n, and $\Phi^n_{t-\Delta t,t}$ are the fees that the LP's position generates between $t-\Delta t$ and t. Then, at time t she chooses optimal controls $\boldsymbol{w}_t, \boldsymbol{\ell}_t, \boldsymbol{u}_t$ to provide liquidity between t and $t+\Delta t$. For each n, the quantities w^n_t, $\mathcal{Z}(i^n_t - \ell^n_t)$, and $\mathcal{Z}(i^n_t + u^n_t + 1)$ determine the LP's liquidity depth $\tilde{\kappa}^n_t$, which is given by

$$\tilde{\kappa}^n_t = w^n_t\, V_t \times \left(2\sqrt{Z^n_t} - \sqrt{\mathcal{Z}(i^n_t - \ell^n_t)} - \frac{Z^n_t}{\sqrt{\mathcal{Z}(i^n_t + u^n_t + 1)}} \right)^{-1}, \tag{6.8}$$

while $\mathcal{Z}(i^n_t - \ell^n_t)$, $\mathcal{Z}(i^n_t + u^n_t + 1)$, Z^n_t, and $\tilde{\kappa}^n_t$ determine the quantities $(\tilde{x}^n_t, \tilde{y}^n_t)$ of assets X and Y_n that the LP needs to deposit in the pool. Specifically, we have

$$\begin{cases} x^n_t = \tilde{\kappa}^n_t \left(\sqrt{Z^n_t} - \sqrt{\mathcal{Z}(i^n_t - \ell^n_t)} \right), \\ y^n_t = \tilde{\kappa}^n_t \left(\dfrac{1}{\sqrt{Z^n_t}} - \dfrac{1}{\sqrt{\mathcal{Z}(i^n_t + u^n_t + 1)}} \right). \end{cases}$$

6.3.3 Performance criterion

The LP uses a performance criterion $\boldsymbol{J} \colon \mathcal{A} \to \mathbb{R}$ to evaluate a strategy $(\boldsymbol{w}, \boldsymbol{\ell}, \boldsymbol{u}) \in \mathcal{A}$. We study two optimal criteria: **Sharpe Ratio**

$$J(\boldsymbol{w}, \boldsymbol{\ell}, \boldsymbol{u}) = \frac{\mathbb{E}[V_T] - V_0}{\sqrt{\mathbb{V}[V_T]}}\,,$$

and **mean-variance**

$$J(\boldsymbol{w}, \boldsymbol{\ell}, \boldsymbol{u}) = \mathbb{E}[V_T] - \gamma\, \mathbb{V}[V_T]\,, \tag{6.9}$$

where γ is a risk aversion parameter, $\mathbb{E}[\,\cdot\,]$ and $\mathbb{V}[\,\cdot\,]$ are the expectation and variance operators, respectively. The goal of the LP is to find a liquidity provision strategy $(\boldsymbol{w}^\star, \boldsymbol{\ell}^\star, \boldsymbol{u}^\star)$ such that

$$(\boldsymbol{w}^\star, \boldsymbol{\ell}^\star, \boldsymbol{u}^\star) = \underset{(\boldsymbol{w},\boldsymbol{\ell},\boldsymbol{u}) \in \mathcal{A}}{\arg\max}\; J(\boldsymbol{w}, \boldsymbol{\ell}, \boldsymbol{u})\,. \tag{6.10}$$

6.4 Model-free optimisation

In this section, we use LSTM networks to solve the optimisation problem in (6.10). LSTM is a type of neural network employed for sequential data; see Hochreiter and Schmidhuber (1997). Specifically, LSTM networks are a particular type of recurrent neural network (RNN).

RNN networks take an $\mathbb{R}^p$-valued input sequence $\boldsymbol{x} = \{x_t\}_{t=0,\dots,T}$ and use information from the past of the sequence to produce an $\mathbb{R}^q$-valued output sequence $\left\{h_t^\theta\right\}_{t=0,\dots,T}$ through a self loop. Typically, an RNN is implemented as a feedforward neural network whose input at time t is the vector x_t and the output h_{t-1}^θ of the RNN at the previous time step. This self loop allows information from the input sequence to persist over time and makes RNN networks particularly suited for sequential data.

However, RNN networks suffer from the vanishing gradient problem which prevents them from learning long term dependencies; see Bengio et al. (1994). LSTM networks solve the problem of long term dependencies by introducing a custom layer, called LSTM layer, which controls the flow of information over time. Typically, LSTM networks are implemented as one or more stacked LSTM layers which take an $\mathbb{R}^p$-valued input sequence $\{x_t\}_{t=0,\dots,T}$ and sequentially produce the $\mathbb{R}^q$-valued output sequence $\left\{h_t^\theta\right\}_{t=0,\dots,T}$ and the $\mathbb{R}^r$-valued hidden *cell* sequence $\left\{c_t^\theta\right\}_{t=0,\dots,T}$. At time t, the inputs to the LSTM network are x_t, the output at the previous time step h_{t-1}^θ, and the hidden cell c_{t-1}^θ; then, the outputs are h_t^θ and c_t^θ. These two self loops and the LSTM layer structure allow long and short term memory to persist over time. Specifically, the output h_t^θ carries short-term information while the hidden cell c_t^θ stores long term information. The amount of information retained by the hidden cell at each time is determined by the *forget gate* inside each LSTM layer; see Goodfellow et al. (2016).

Learning long-term dependencies is particularly relevant in financial applications because information in financial time series tends to persist over time. For instance, in Cartea et al. (2022b) and Cartea et al. (2023a) the authors show that the past values of the instantaneous rate in Binance can be used to forecast the instantaneous rate in Uniswap v3.

6.4.1 Implementation

Next, we provide details about the neural network architecture we use to approximate the solution to the optimisation problem in (6.10). Specifically, we consider a class of LSTM networks with input dimension $\mathbb{R}^D$ and output dimension $\mathbb{R}^N \times \mathbb{R}^{(\mathcal{T}+1)N} \times \mathbb{R}^{(\mathcal{T}+1)N}$.

At time t, the input to the LSTM network is the D-dimensional vector $\mathfrak{I}_t$ representing the information available to the LP up to time t. The network's output is the vector $\boldsymbol{h}_t = (\boldsymbol{h}_t^{\theta,w}, \boldsymbol{h}_t^{\theta,\ell}, \boldsymbol{h}_t^{\theta,u})$, where θ are the network parameters, $\boldsymbol{h}_t^{\theta,w} \in \mathbb{R}^N$, $\boldsymbol{h}_t^{\theta,\ell} \in \mathbb{R}^{(\mathcal{T}+1)N}$, and $\boldsymbol{h}_t^{\theta,u} \in \mathbb{R}^{(\mathcal{T}+1)N}$. Next, to obtain the controls $\boldsymbol{w}_t^\theta, \boldsymbol{\ell}_t^\theta, \boldsymbol{u}_t^\theta$, we further apply a custom layer to the output vector $\boldsymbol{h}_t^\theta$.

Allocation weights The allocation weights $\boldsymbol{w}_t^\theta$ are given by

$$\boldsymbol{w}_t^\theta = \text{softmax}\left(\boldsymbol{h}_t^{\theta,w}\right),$$

and

$$\left[\boldsymbol{w}_t^\theta\right]_n = \left[\text{softmax}\left(\boldsymbol{h}_t^{\theta,w}\right)\right]_n = \frac{\exp\left(\left[\boldsymbol{h}_t^{\theta,w}\right]_n\right)}{\sum_{k=1}^N \exp\left(\left[\boldsymbol{h}_t^{\theta,w}\right]_k\right)},$$

for each $n \in \{0,\dots,N\}$, where $[\,\cdot\,]_n$ is the nth entry of the input vector. The softmax activation function enforces the condition in (6.6).

Spreads To obtain the spreads $\boldsymbol{\ell}_t^\theta$ and $\boldsymbol{u}_t^\theta$, we split the $(\mathcal{T}+1)N$-dimensional output vectors $\boldsymbol{h}_t^{\theta,\ell}$ and $\boldsymbol{h}_t^{\theta,u}$ into N vectors in $\mathbb{R}^{\mathcal{T}+1}$

$$\boldsymbol{h}_t^{\theta,\ell} = \left(\boldsymbol{h}_t^{1,\theta,\ell},\dots,\boldsymbol{h}_t^{N,\theta,\ell}\right) \quad \text{and} \quad \boldsymbol{h}_t^{\theta,u} = \left(\boldsymbol{h}_t^{1,\theta,u},\dots,\boldsymbol{h}_t^{N,\theta,u}\right),$$

and then convert the $\mathcal{T}+1$-dimensional vectors $\boldsymbol{h}_t^{n,\theta,\ell}$ and $\boldsymbol{h}_t^{n,\theta,u}$ into the controls $\ell_t^{n,\theta}$ and $u_t^{n,\theta}$, which are given by

$$\ell_t^{n,\theta} = \underset{i\in\{0,\dots,\mathcal{T}\}}{\arg\max}\left\{\left[\boldsymbol{h}_t^{n,\theta,\ell}\right]_i\right\} \quad \text{and} \quad u_t^{n,\theta} \underset{i\in\{0,\dots,\mathcal{T}\}}{\arg\max}\left\{\left[\boldsymbol{h}_t^{n,\theta,u}\right]_i\right\}.$$

Finally, for each set of parameters θ, we obtain an LSTM liquidity provision strategy $\left(\boldsymbol{w}^\theta,\boldsymbol{\ell}^\theta,\boldsymbol{u}^\theta\right)$.

Next, the LP trains the LSTM network using the performance criterion J to approximate the optimal liquidity provision strategy in (6.10) with $\left(\boldsymbol{w}^{\theta^\star},\boldsymbol{\ell}^{\theta^\star},\boldsymbol{u}^{\theta^\star}\right)$, where

$$\theta^\star = \underset{\theta\in\Theta}{\arg\max}\, J\left(\boldsymbol{w}^\theta,\boldsymbol{\ell}^\theta,\boldsymbol{u}^\theta\right).$$

6.5 LSTM strategy with simulated data

In this section, we test the LSTM strategy on simulated data to showcase the performance of the strategy under various market conditions and in Section 6.6 we test the LSTM strategy on market data. Here, for each experiment, we train an LSTM network of five LSTM layers with 24 neurons each. The dropout rate is 0.1 and we use the Adam optimiser for training; see Kingma and Ba (2015). For each experiment we draw 256,000 samples and use 60% of them for training, 20% for validation, and 20% for testing. We use the mean-variance criterion in (6.9) to train our model.

6.5.1 Model

We introduce a continuous-time model for the N liquidity pools where the LP provides liquidity. We use this model to generate samples to train the LSTM network, and obtain the LSTM liquidity provision strategy.

We fix a filtered probability space $\left(\Omega,\mathcal{F},\mathbb{P},\tilde{\mathbb{F}}=\{\mathcal{F}_t\}_{t\in[0,T]}\right)$ satisfying the usual conditions. Here, $\tilde{\mathbb{F}}$ is the natural filtration generated by the $\mathbb{R}^D$-valued stochastic process $\{\mathfrak{I}_t\}_{t\in[0,T]}$ which represents the information available to the LP at each time. Note that $\mathbb{F}\subset\tilde{\mathbb{F}}$, where $\mathbb{F}=\{\mathcal{F}_t\}_{t=0,\dots,T}$ is the natural filtration introduced in Section 6.3.

When a liquidity taking order arrives in the pool, the instantaneous rate is updated according to the LT condition in Section 6.2, and the volume of the order is distributed among LPs according to

their contribution to market depth and on the fee tier τ_n. Here, we distinguish between the market depth of the LP and the aggregated market depth of the other LPs. The market depth of the LP at time t in pool n is denoted by $\tilde{\kappa}_t^n$ and it is given by (6.8), and the aggregated market depth of the other LPs in the pool is the $\mathbb{R}_{++}$-valued stochastic process κ_t^n. We assume that during the trading horizon there is no liquidity provision activity from other LPs and we further assume that their market depth κ_t^n is constant across different ticks. Therefore, κ_t^n is constant during the entire trading horizon. We write κ_n to denote the constant aggregated market depth of the other LPs, so that the total market depth of the pool at time t is $\kappa_n + \tilde{\kappa}_t^n$.

Liquidity taking orders arrive in the pool according to a Poisson process N_t^n with parameter λ_n. A liquidity taking order is a buy order with probability $p_n \in (0,1)$ and its size, in units of asset X, is normally distributed with mean $\mu_n > 0$ and standard deviation $\xi_n > 0$. Let $\{B_t^n\}_{t\in[0,T]}$ be i.i.d. Bernoulli random variables with parameter p_n, and let $\{S_t^n\}_{t\in[0,T]}$ be i.i.d. normal random variables with mean μ_n and standard deviation ξ_n, independent of $\{B_t^n\}_{t\in[0,T]}$. Then, the liquidity taking order flow in units of asset X of buy and sell orders during the trading horizon is given by

$$\begin{aligned} dO_t^{n,\text{buy}} &= B_t^n \, S_t^n \, dN_t^n \,, \\ dO_t^{n,\text{sell}} &= (1 - B_t^n) \, S_t^n \, dN_t^n \,. \end{aligned}$$

Here, we assume that the arrival of liquidity taking orders is independent between pools, so the Poisson processes $N_t^1, \ldots, N_t^N$ are independent.

In our experiments, we simulate the Poisson process N_t^n, the random variables $\{B_t^n\}_{t\in[0,T]}$, and $\{S_t^n\}_{t\in[0,T]}$ to obtain the order flow processes $O_t^{n,\text{buy}}$ and $O_t^{n,\text{sell}}$. We update the instantaneous rate Z_t^n and distribute trading fees among LPs according to the LT condition in Section 6.2.

6.5.2 Single pool

Here, we consider an LP who provides liquidity in one liquidity pool, i.e., $N = 1$, that trades the reference asset X and the risky asset Y, and the instantaneous rate at time $t = 0$ is $Z_0 = 2{,}200$. The LP provides liquidity for 24 hours and readjusts her position every 30 minutes, so $T = 1{,}440$ minutes and $\Delta t = 30$ minutes. The LP's initial wealth is 500,000 units of asset X and the maximum value for the lower and upper spreads that the LP considers is $\mathcal{T} = 500$. At each time $t \in \mathbb{T}$ the input to the LSTM strategy is $\mathfrak{I}_t = \left(w_{t-\Delta t}^\theta, \ell_{t-\Delta t}^\theta, u_{t-\Delta t}^\theta, Z_t, \Phi_{t-\Delta t,t}\right)$.

Liquidity taking orders arrive in the pool at a Poisson rate of $\lambda \in \mathbb{R}_{++}$ and the distribution of the order size is normal with mean $\mu_{\text{size}} = 132{,}030$ and standard deviation $\sigma_{\text{size}} = 20{,}000$ units of asset X. These choices of parameters are based on market data from the ETH/USDC 0.3% pool in Uniswap v3 between 1 July 2021 and 30 September 2023; see Table 6.9.

6.5.2.1 Trading frequency

Here, we train the LSTM network for various values of the arrival rate λ of liquidity taking orders, while keeping the remaining parameters fixed. The market depth is $\kappa = 15{,}000{,}000$, the fee tier $\tau = 0.3\%$, and $p = 0.5$ is the probability that the liquidity taking order is a buy. When the value

of λ is large, orders arrive more frequently to the pool and the volatility of the instantaneous rate increases; Table 6.1 shows the volatility of 30-minute returns as a function of the arrival rate λ.

Arrival rate λ	$1\ \text{min}^{-1}$	$\frac{1}{3}\ \text{min}^{-1}$	$\frac{1}{5}\ \text{min}^{-1}$	$\frac{1}{10}\ \text{min}^{-1}$	$\frac{1}{20}\ \text{min}^{-1}$
Volatility $\sigma_{\mathbf{30min}}$	7.70%	4.64%	3.49%	2.51%	1.77%

Table 6.1: Volatility of 30-minute returns $\sigma_{30\text{min}}$ as a function of the arrival rate λ. Market depth $\kappa = 15{,}000{,}000$, fee tier $\tau = 0.3\%$, and $p = 0.5$ is the probability that a liquidity taking order is a buy order.

Table 6.2 reports the performance of the LSTM strategy on the test set with simulated data. The table shows the average lower and upper spreads posted by the strategy, and it shows the Sharpe Ratio and the average returns of the strategy, both before and after gas fees; whenever the LP adjusts her position she pays 73.3 units of asset X for gas fees, which is the average USD amount of gas fees paid by LPs for withdrawing or depositing liquidity from the ETH/USDC 0.3% pool in Uniswap v3 between 1 July 2021 and 30 September 2023.

Arrival rate λ	**Sharpe Ratio (after gas fees)**	**Returns (after gas fees)**	**Average lower spread**	**Average upper spread**
1	0.22 (−13.80)	0.01% (−0.71%)	496.00	8.00
1/3	3.74 (−1.26)	0.54% (−0.18%)	88.95	7.75
1/5	4.63 (2.41)	1.51% (0.78%)	24.76	9.20
1/10	6.14 (4.15)	2.23% (1.51%)	15.00	4.00
1/20	**6.37** (**4.80**)	**2.93%** (**2.21%**)	13.43	4.00

Table 6.2: Performance of the LSTM strategy on test set of simulated data. The LP trades in a CPMM with one liquidity pool with CL. We assume market depth $\kappa = 15{,}000{,}000$, fee tier $\tau = 0.3\%$, and $p = 0.5$ is the probability that a liquidity taking order is a buy order.

The position of the LP generates fees only when the instantaneous rate is within the range where she provides liquidity. When volatility is high, the spread of the LSTM strategy is high to ensure that the instantaneous rate remains within the range where the LP is providing liquidity. On the other hand, when volatility is low, the strategy posts a narrow spread, which increases the value of the LP's market depth $\tilde{\kappa}$, to increase the amount of fees that she earns; see (6.7). This result aligns with the finding in Cartea et al. (2023b). They show that LPs widen (resp. tighten) the range where they provide liquidity when volatility is higher (resp. lower).

If the value of the lower spread ℓ is higher than the value of the upper spread u, then the majority of the LP's wealth is allocated to the reference asset X; see (6.4). On the other hand, when $u > \ell$ the LP allocates most of her wealth to the risky asset Y. If the LP allocates most of her liquidity to asset Y, then she is more exposed to the fluctuations of the instantaneous rate; the value of the position in Y depends on the instantaneous rate. Table 6.2 shows that when the volatility of the instantaneous rate is high (i.e., when the value of the arrival rate λ of liquidity taking orders is

high), the mean-variance criterion pushes the LP to provide liquidity in a range which is skewed towards the reference asset X to reduce the standard deviation of the LP's returns.

Figure 6.1 shows the LP's liquidity provision spread as a function of the volatility. Specifically, the red line is the upper spread u and the blue line is the negative lower spread $-\ell$. The distance between the red and the blue line is the average spread of the LSTM strategy. As volatility increases, the LP widens the range where she is providing liquidity and skews her position towards the reference asset X.

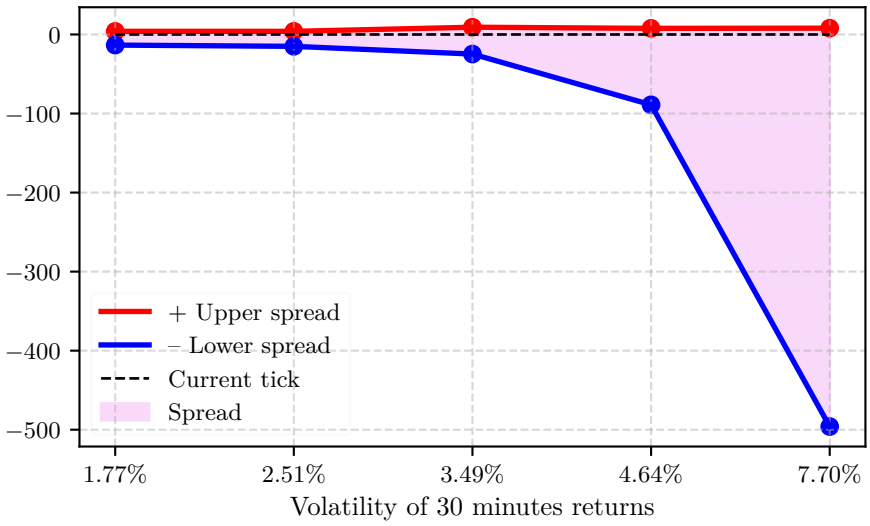


Figure 6.1: Average upper and lower spreads posted by the LSTM strategy as functions of the volatility $\sigma_{30\text{min}}$.

Average returns and Sharpe Ratio, both before and after gas fees, increases when the arrival rate λ decreases. When $\lambda = 1/20$, the average returns of the LSTM strategy are 2.93% before gas fees and 2.21% after gas fees. The Sharpe Ratio of the strategy is 6.37 before gas fees and 4.80 after gas fees. When $\lambda = 1$, the average returns of the LSTM strategy are 0.01% before gas fees and −0.71% after gas fees, and Sharpe Ratio of the strategy is 0.22 before gas fees and −13.80 after gas fees.

6.5.2.2 Fee tier

Next, we train the LSTM network for various values of the fee tier τ, while keeping the remaining parameters fixed, arrival rate $\lambda = 1/3$ min^{-1}, market depth $\kappa = 15{,}000{,}000$, and probability of liquidity taking order to be a buy order $p = 0.5$.

Table 6.3 reports the performance of the LSTM strategy on the test set. It shows the average lower and upper spreads posted by the strategy and shows the Sharpe Ratio and the average returns of the strategy, both before and after gas fees. As above, whenever the LP adjusts her position she pays 73.3 units of asset X for gas fees.

Fee Tier τ	Sharpe Ratio (after gas fees)	Returns (after gas fees)	Average lower spread	Average upper spread
0.1%	0.09 (−32.84)	0.00% (−0.72%)	382.88	12.16
0.3%	3.74 (−1.26)	0.54% (−0.18%)	88.95	7.75
1.0%	**8.60** (**7.32**)	**4.83%** (**4.10%**)	12.83	3.92

Table 6.3: Performance of the LSTM strategy on test set of simulated data. We assume arrival rate is $\lambda = 1/3$ min^{-1}, market depth is $\kappa = 15{,}000{,}000$, and the probability of a buy liquidity taking order is $p = 0.5$.

Figure 6.2 shows the LP's range as a function of the fee tier τ. As the fee tier increases, we see that the LP narrows the range where she provides liquidity and increases her exposure to the risky asset Y. This result aligns with the finding of Cartea et al. (2023b), who show that LPs widen the range where they provide liquidity when the average fee income over time is lower.

Figure 6.2 also shows that as the value of the fee tier decreases, the LP skews her position more towards the reference asset X. Indeed, the exposure to the risks associated with Y benefits the LP only if the fee yield is high enough.

Average returns of the LSTM strategy are highest when $\tau = 1.0\%$; they are equal to 4.83% and 4.10%, respectively, before and after gas fees. When $\tau = 1.0\%$, the LP provides liquidity in a narrow range around the instantaneous rate such that the average lower and upper spreads $\ell^\theta = 12.83$ and $u^\theta = 3.92$. Returns are lowest when $\tau = 0.1\%$ and they are 0.00% before gas fees and −0.72% after gas fees. The LP skews her position towards the reference asset X and the average lower and upper spreads are $\ell^\theta = 382.88$ and $u^\theta = 12.16$.

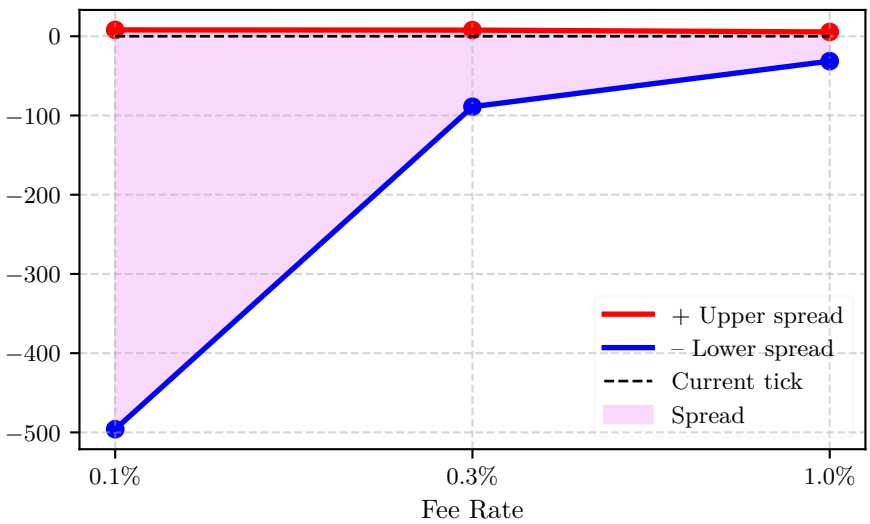


Figure 6.2: Average upper and lower spread posted by the LSTM strategy as functions of the fee tier τ.

6.5.2.3 Trend

Here, we train the LSTM network for various values of the probability p of a buy liquidity taking, while keeping the remaining parameters fixed. The arrival rate is $\lambda = 1/3$ min^{-1}, market depth is $\kappa = 15{,}000{,}000$, and fee tier is $\tau = 0.3\%$. When $p < 1/2$, sell orders arrive more frequently than buy orders, so the pool's marginal rate trends downwards, on average. Similarly, when $p > 1/2$, the pool marginal rate moves upwards, on average. Table 6.4 shows how the average 30-minute returns $\widehat{\mu}_{30\text{min}}$ changes as a function of the probability p.

Probability of buy order p	0.10	0.25	0.50	0.75	0.90
Average Returns $\widehat{\mu}_{\mathbf{30min}}$	−0.29%	−0.18%	0.00%	0.19%	0.31%

Table 6.4: Average 30-minute returns $\widehat{\mu}_{30\text{min}}$ as a function of the probability p of a buy liquidity taking order. Market depth is $\kappa = 15{,}000{,}000$, fee tier is $\tau = 0.3\%$, and arrival rate is $\lambda = 1/3$ min^{-1}.

Table 6.5 reports the performance of the LSTM strategy on the test set. It shows the average lower and upper spreads posted by the strategy, and shows the Sharpe Ratio and the average returns of the strategy, both before and after gas fees. As above, gas fees are 73.3 units of asset X.

Trend p	**Sharpe Ratio (after gas fees)**	**Returns (after gas fees)**	**Average lower spread**	**Average upper spread**
0.10	−7.86 (−21.18)	−0.43% (−1.15%)	495.00	2.00
0.25	−3.40 (−21.11)	−0.14% (−0.86%)	492.00	2.00
0.50	3.74 (−1.26)	0.54% (−0.18%)	88.95	7.75
0.75	14.44 (13.35)	9.59% (8.87%)	54.85	170.41
0.90	**25.40** (**24.30**)	**16.70%** (**15.97%**)	69.14	217.83

Table 6.5: Performance of the LSTM strategy on test set of simulated data. The LP trades in a CPMM with one liquidity pool with CL. Arrival rate is $\lambda = 1/3$ min^{-1}, market depth is $\kappa = 15{,}000{,}000$, and fee tier is $\tau = 0.3\%$.

Figure 6.3 shows the LP's range as a function of the average 30-minute returns $\widehat{\mu}_{30\text{min}}$. When $\widehat{\mu}_{30\text{min}} < 0$, the LP skews her position towards the reference asset X to reduce her losses associated to the drop of the instantaneous rate. Conversely, when $\widehat{\mu}_{30\text{min}} > 0$, the LP skews her position towards the risky asset Y to take advantage of the upward trend in the instantaneous rate Z. This result aligns with the finding of Cartea et al. (2023b), who show LPs skew their position in the direction of the rate's trend.

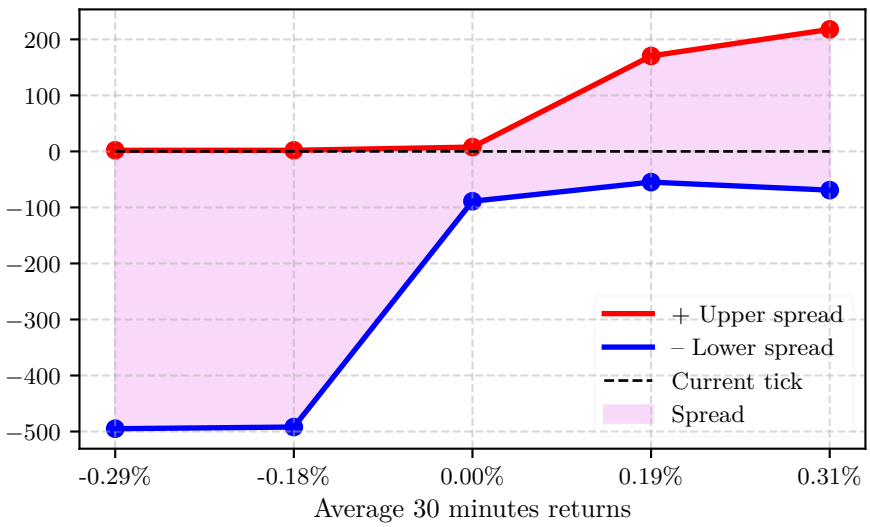


Figure 6.3: Average upper and lower spread posted by the LSTM strategy as functions of the average 30-minute returns $\widehat{\mu}_{30\text{min}}$.

When there is a negative trend, the LP keeps very little inventory in the risky asset Y, while when there is a positive trend, she keeps a considerable part of her inventory in the reference asset X; see Figure 6.3. This is a consequence of using the mean-variance criterion, where it is optimal to increase this LP's investment in the risk-free reference asset X.

Table 6.5 shows that the returns of the LSTM strategy are highest when $p = 0.9$; returns are 16.70% and 15.97%, before and after gas fees, respectively. When $p = 0.9$ the LP skews her position towards the risky asset Y to capitalise on the upward trend. The average lower and upper spreads are $\ell^\theta = 69.14$ and $u^\theta = 217.83$. Average returns are lowest when $p = 0.1$ and the LP skews her position towards the reference asset X to reduce losses related to the negative trend in the value of Y. Average returns are equal to -0.43% and -1.15% before and after gas fees, respectively. The average lower and upper spreads are $\ell^\theta = 495.00$ and $u^\theta = 2.00$.

6.5.3 Two pools

Next, consider an LP who provides liquidity in two liquidity pools, i.e., $N = 2$. One trades the pair X and Y_1, and the other trades the pair X and Y_2, and the instantaneous rates at time $t = 0$ are $Z_0^1 = Z_0^2 = 2{,}200$. The LP provides liquidity for 24 hours and readjusts her position every 30 minutes, i.e., $T = 1{,}440$ minutes and $\Delta t = 30$ minutes. The LP's initial wealth is 500,000 units of asset X and the maximum value for the lower and upper spreads that the LP considers is $\mathcal{T} = 500$. At each time $t \in \mathbb{T}$ the input to the LSTM strategy is $\mathfrak{I}_t = \left(\boldsymbol{w}^\theta_{t-\Delta t}, \boldsymbol{\ell}^\theta_{t-\Delta t}, \boldsymbol{u}^\theta_{t-\Delta t}, \boldsymbol{Z}_t, \boldsymbol{\Phi}_{t-\Delta t,t}\right)$.

Liquidity taking orders arrive in the pools at Poisson rates of $\lambda_1, \lambda_2 \in \mathbb{R}_{++}$ and the distribution of the order size in both pools is normal with mean $\mu_{\text{size}} = 132{,}030$ and standard deviation $\sigma_{\text{size}} = 20{,}000$ units of asset X. These choices of parameters are based on market data from the ETH/USDC 0.3% pool in Uniswap v3 between 1 July 2021 and 30 September 2023; see Table 6.9.

6.5.3.1 Trading frequency

Here, we train the LSTM network for various values of the arrival rates λ_1 and λ_2, while keeping the remaining parameters fixed. Market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, the fee tiers are $\tau_1 = \tau_2 = 0.3\%$, and $p_1 = p_2 = 0.5$ are the probabilities of a buy liquidity taking order in each pool.

Table 6.6 reports the performance of the LSTM strategy on the test set, measured by returns and Sharpe Ratio, both before and after gas fees; at each time step the LP pays 73.3 units of asset X for each pool where she adjusts her position. The table also shows the average lower and upper spreads posted by the strategy and the average proportion of wealth that the LP provides in each pool.

Overall, Table 6.6 reports similar results to those of the experiments in Section 6.5.2.1. Indeed, the behaviour of the LSTM strategy in each individual pool is qualitatively the same as that of the single pool presented in Table 6.2. Specifically, the LP provides liquidity in wide ranges when the value of λ is big and in narrow ranges when the value of λ is small. Moreover, when the value of λ is big, the LP skews her position towards the reference asset X, and when the value of λ is small, she increases her exposure to the risky asset Y. Note that the LP skews her position because the mean-variance criterion pushes the LP to reduce her exposure to the risky asset Y to reduce the standard deviation of the strategy returns.

Table 6.6 shows that the LSTM strategy provides more liquidity to the pool where volatility is lower. For instance, when the arrival rate $\lambda_2 = 1$ and $\lambda_1 < \lambda_2$, then the LP deposits more than 99% of her wealth in the first pool.

Arrival rate λ		Sharpe Ratio (after gas fees)	Returns (after gas fees)	Average lower spread		Average upper spread		Weights	
Pool 1	Pool 2			Pool 1	Pool 2	Pool 1	Pool 2	Pool 1	Pool 2
1	1	0.35 (−35.34)	0.01% (−1.43%)	494.00	480.62	12.73	7.75	43.69%	56.31%
1/3	1	3.57 (−2.77)	0.81% (−0.63%)	51.89	480.65	8.51	34.72	99.43%	0.57%
1/5	1	4.60 (−1.03)	1.18% (−0.26%)	37.14	480.60	6.68	75.63	97.66%	2.34%
1/10	1	6.98 (1.52)	1.85% (0.40%)	19.40	497.00	6.98	1.06	99.15%	0.85%
1/20	1	9.64 (4.58)	2.75% (1.31%)	74.53	492.00	5.00	425.25	**99.85%**	0.15%
1/3	1/3	5.42 (0.60)	1.63% (0.18%)	15.08	24.00	6.37	7.00	42.50%	57.50%
1/5	1/3	6.38 (2.03)	2.12% (0.67%)	14.08	9.20	2.96	3.86	67.28%	32.72%
1/10	1/3	6.44 (2.16)	2.18% (0.73%)	45.50	471.00	5.33	5.15	94.05%	5.95%
1/20	1/3	**11.46** (**6.62**)	**3.42%** (**1.97%**)	26.94	93.29	7.66	3.43	81.97%	18.03%
1/5	1/5	5.20 (−0.50)	1.32% (−0.13%)	28.51	175.90	5.37	431.18	98.97%	1.03%
1/10	1/5	5.81 (0.67)	1.64% (0.19%)	167.41	9.34	21.14	5.03	51.69%	48.31%
1/20	1/5	9.46 (4.79)	2.93% (1.48%)	48.05	234.57	7.35	5.48	74.30%	25.70%
1/10	1/10	7.63 (2.30)	2.07% (0.62%)	209.04	13.55	9.98	4.49	23.63%	**76.37%**
1/20	1/10	8.12 (4.51)	3.25% (1.81%)	42.26	32.19	3.85	4.00	62.30%	37.70%
1/20	1/20	9.49 (4.10)	2.55% (1.10%)	79.80	351.76	4.24	296.61	74.33%	25.67%

Table 6.6: Performance of the LSTM strategy on test set of simulated data, in a CPMM with two liquidity pools with CL. Market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, fee tiers are $\tau_1 = \tau_2 = 0.3\%$, and the probabilities of a buy order are $p_1 = p_2 = 0.5$.

6.5.3.2 Fee tier

Here, we train the LSTM network for various values of the fee tiers τ_1 and τ_2, while keeping the remaining parameters fixed. Arrival rates are $\lambda_1 = \lambda_2 = 1/3$, market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, and $p_1 = p_2 = 0.5$ are the probabilities that of buy orders.

Table 6.7 reports the performance of the LSTM strategy on the test set, measured by returns and Sharpe Ratio, both before and after gas fees; as above, she pays 73.3 units of asset X in each pool for every adjustment of liquidity. The table also shows the average lower and upper spreads posted by the strategy and the average proportion of wealth that the LP provides in each pool.

Overall, Table 6.7 reports similar results to those of the experiments in Section 6.5.2.2. Indeed, the behaviour of the LSTM strategy in each individual pool is qualitatively the same as that of the single pool presented in Table 6.3. Specifically, the LP provides liquidity in wide ranges when the value of the fee tier τ is small and in narrow ranges when the value of τ is big. Moreover, when the value of the fee tier is low, the LP skews her position towards the reference asset X.

Table 6.7 shows that the LSTM strategy provides more liquidity to the pool where the fee tier τ is higher. For instance, when $\tau_1 = 1.0\%$ and $\tau_2 = 0.1\%$, the LP provides all her liquidity to the first pool.

The returns of the LSTM strategy are highest when $\tau_1 = \tau_2 = 1.0\%$ and the LP splits her wealth almost equally between the two pools. Returns are equal to 17.67% before gas fees and to 16.22% after gas fees.

Fee Tier τ		**Sharpe Ratio (after gas fees)**	**Returns (after gas fees)**	**Average lower spread**		**Average upper spread**		**Weights**	
Pool 1	**Pool 2**			**Pool 1**	**Pool 2**	**Pool 1**	**Pool 2**	**Pool 1**	**Pool 2**
0.1%	0.1%	0.12 (−60.00)	−0.00% (−1.44%)	480.75	481.28	14.00	19.15	57.13%	42.87%
0.3%	0.1%	3.19 (−2.61)	0.80% (−0.65%)	84.86	480.32	4.00	33.86	96.40%	3.60%
1.0%	0.1%	7.78 (4.04)	3.01% (1.57%)	70.66	491.00	5.46	397.69	**99.49%**	0.51%
0.3%	0.3%	5.42 (0.60)	1.63% (0.18%)	15.08	24.00	6.37	7.00	42.50%	**57.50%**
1.0%	0.3%	8.39 (4.26)	2.94% (1.49%)	71.32	21.55	6.78	13.96	82.31%	17.69%
1.0%	1.0%	**11.05** (**8.51**)	**6.29%** (**4.84%**)	42.31	23.58	11.95	11.78	53.73%	46.27%

Table 6.7: Performance of the LSTM strategy on test set of simulated data. The LP trades in a CPMM with two liquidity pools with CL. Arrival rates are $\lambda_1 = \lambda_2 = 1/3$, market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, and the probabilities of a buy order are $p_1 = p_2 = 0.5$.

6.5.3.3 Trend

Here, we train the LSTM network for various values of the probabilities p_1 and p_2 of liquidity taking orders to be buy orders in each pool, while keeping the remaining parameters fixed. Arrival rates are $\lambda_1 = \lambda_2 = 1/3$ min^{-1}, market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, and fee tiers are $\tau_1 = \tau_2 = 0.3\%$.

Table 6.8 reports the performance of the LSTM strategy on the test set, measured by returns and Sharpe Ratio, both before and after gas fees; as above, she pays 73.3 units of asset X in each pool for every adjustment of liquidity. The table also shows the average lower and upper spreads posted by the strategy and the average proportion of wealth that the LP provides in each pool.

Overall, Table 6.8 reports similar results to those of the experiments in Section 6.5.2.3. Indeed, the behaviour of the LSTM strategy in each individual pool is qualitatively the same as that of presented in Table 6.5. Specifically, when there is a downward trend in the instantaneous rate, the LP skews her position towards the reference asset X. Conversely, when the instantaneous rate trends upwards, the LP skews her position towards the risky asset. Similarly to the results of Section 6.5.2.3, the mean-variance criterion pushes the LP to keep a considerable position in the reference asset even when the instantaneous rate has a strong upwards trend.

Table 6.6 shows that the LSTM strategy provides more liquidity to the pool when the probability p of a buy order is higher. For instance, when $p_1 = 0.25$ and $p_2 = 0.1$, the LP provides all of her liquidity to the first pool.

The returns of the LSTM strategy are highest when $p_1 = p_2 = 0.9$ and there is a strong upward trend in both pools. Returns are equal to 17.67% before gas fees and to 16.22% after gas fees.

Trend p		**Sharpe Ratio (after gas fees)**	**Returns (after gas fees)**	**Average lower spread**		**Average upper spread**		**Weights**	
Pool 1	**Pool 2**			**Pool 1**	**Pool 2**	**Pool 1**	**Pool 2**	**Pool 1**	**Pool 2**
10%	10%	−7.48 (−34.91)	−0.39% (−1.84%)	500.00	500.00	1.00	1.00	0.00%	**100.00%**
25%	10%	−3.12 (−40.16)	−0.12% (−1.57%)	500.00	500.00	1.00	1.00	**100.00%**	0.00%
50%	10%	3.71 (−5.43)	0.59% (−0.86%)	62.01	425.47	4.02	16.41	98.24%	1.76%
75%	10%	20.21 (17.13)	9.49% (8.04%)	55.86	483.00	71.55	5.00	99.95%	0.05%
90%	10%	34.65 (31.63)	16.57% (15.12%)	1.00	483.00	261.14	2.00	93.04%	6.96%
25%	25%	−3.12 (−40.48)	−0.12% (−1.57%)	500.00	500.00	1.00	1.00	0.00%	100.00%
50%	25%	3.21 (−1.73)	0.94% (−0.51%)	32.47	499.10	6.15	16.00	98.56%	1.44%
75%	25%	16.50 (13.95)	9.35% (7.91%)	5.62	478.12	67.00	17.78	92.16%	7.84%
90%	25%	**42.15** (**38.52**)	16.80% (15.35%)	8.48	483.00	64.80	4.00	89.26%	10.74%
50%	50%	5.42 (0.60)	1.63% (0.18%)	24.00	15.08	7.00	6.37	57.50%	42.50%
75%	50%	15.97 (13.57)	9.64% (8.20%)	26.81	50.90	139.64	38.95	96.91%	3.09%
90%	50%	33.24 (30.23)	16.00% (14.55%)	43.19	45.54	289.96	356.38	95.72%	4.28%
75%	75%	22.70 (19.36)	9.84% (8.39%)	7.42	33.99	65.05	82.86	46.88%	53.12%
90%	75%	28.23 (25.81)	16.87% (15.43%)	4.38	83.83	106.30	500.00	87.94%	12.06%
90%	90%	25.31 (23.24)	**17.67%** (**16.22%**)	26.44	9.85	332.94	176.75	47.70%	52.30%

Table 6.8: Performance of the LSTM strategy on test set of simulated data. The LP trades in a CPMM with two liquidity pools with CL. Arrival rates are $\lambda_1 = \lambda_2 = 1/3$, market depths are $\kappa_1 = \kappa_2 = 15{,}000{,}000$, and fee tiers are $\tau_1 = \tau_2 = 0.3\%$.

6.6 Numerical experiments with market data

6.6.1 Data

6.6.1.1 Uniswap v3

We use LP and LT data from the Uniswap v3 pools for the pair ETH/USDC and the pair BTC/USDC between 1 July 2021 and 30 September 2023 to train the LSTM strategy introduced in Section 6.4. The fee tier in both pools is 0.3%.

	ETH/USDC 0.3%		BTC/USDC 0.3%	
	LT	LP	LT	LP
Number of instructions	380,162	115,170	136,365	16,307
Average daily number of instructions	462	140	166	20
Total USD volume	$\approx$ \$ 50.2$\times 10^9$	$\approx$ \$ 160.9 $\times 10^9$	$\approx$ \$ 10.3 $\times 10^9$	$\approx$ \$ 29.9 $\times 10^9$
Average daily USD volume	\$ 61,061,745	\$ 195,700,068	\$ 12,560,091	\$ 36,361,862
Average LT transaction or LP operation size	\$ 132,030	\$ 1,396,765	\$ 75,711	\$ 1,832,921
Average trading frequency	3 minutes	10 minutes	9 minutes	72 minutes
Average pool depth	14,443,050 $\sqrt{\text{ETH} \cdot \text{USDC}}$		1,342,009 $\sqrt{\text{BTC} \cdot \text{USDC}}$	

Table 6.9: LT and LP activity details in the ETH/USDC 0.3% and ETH/USDC 0.3% pools between 1 July 2021 and 30 September 2023.

Table 6.9 shows statistics for both pools. Overall, Table 6.9 shows that there is more trading activity in the ETH/USDC than in the BTC/USDC pool. The total number of both LT and LP instructions in the ETH/USDC pool is significantly higher than in the BTC/USDC pool. Here, an LT instruction is a buy or a sell liquidity taking order, while an LP instruction is to deposit or to withdraw liquidity into and from the pool, respectively. The frequency of arrival of LT and LP instructions is significantly higher in the ETH/USDC pool compared with the BTC/USDC pool. Specifically, on average in the ETH/USDC pool there is an LT instruction every 3 minutes, compared to one every 9 minutes in the BTC/USDC pool. Similarly, on average in the ETH/USDC pool there is an LP instruction every 10 minutes, while in the BTC/USDC pool there is one every 72 minutes.

The total USD volume traded in the two pools is significantly different. in the ETH/USDC pool, the total USD volume traded by LTs is around \$ 50.2×10^9, which is roughly five times greater than the total USD volume traded by LTs in the BTC/USDC pool, which is approximately \$ 10.3×10^9.

6.6.1.2 Binance

Cartea et al. (2022b) solves the problem of an LT who liquidates a large position in a CPMM with CL. The authors backtest their strategy on market data from Uniswap v3 and use the instantaneous rate from Binance as market signals to enhance the performance of their strategy. At present, Binance is the largest centralised exchange for cryptocurrencies and it is organised as a limit order book market. Here, alongside data from Uniswap v3, we use data from Binance to train the LSTM strategy introduced in Section 6.4. Specifically, we use volume and rate data for the pair ETH/USDC and the pair BTC/USDC between 1 July 2021 and 30 September 2023; Table 6.10 shows statistics for both pairs.

A comparison between Table 6.9 and 6.10 highlights that in Uniswap v3 there is significantly more activity in the ETH/USDC pair than in BTC/USDC, while in Binance there is more activity in BTC/USDC than in ETH/USDC. Specifically, the total USD volume traded by LTs in the

BTC/USDC pair is approximately \$ 1,386.2$\times 10^9$, which is roughly 7 times higher than the total USD volume traded in the ETH/USDC pair, equal to \$ 201.6$\times 10^9$.

In Binance, the average trading sizes of ETH/USDC and BTC/USDC are \$ 1,079 and \$ 953, respectively. These values are significantly lower than the average trading sizes of the same pairs in Uniswap v3, which are equal to \$ 132,030 and \$ 75,711; see Table 6.9. Gas fees are the main reason behind the difference in the average trading sizes between the two trading venues. Indeed, in Uniswap v3 users pay a high flat fee, which prevents them from sending multiple small orders to the pool and to opt instead for sending individual large orders.

	ETH/USDC	BTC/USDC
Total USD volume	$\approx$ 201.6 $\times 10^9$	$\approx$ 1,386.2 $\times 10^9$
Average daily USD volume	245,365,037	1,687,095,461
Average daily number of trades	227,385	1,770,836
Average USD trade size	\$ 1,079	\$ 953
Average trading frequency	0.38 seconds	0.05 seconds

Table 6.10: Binance activity in ETH/USDC and BTC/USDC between 1 July 2021 and 30 September 2023.

6.6.2 Features

We train the LSTM strategy from Section 6.4 with market data from Uniswap v3 and Binance between 1 July 2021 and 30 September 2023. At each time t the input to the strategy $\mathfrak{I}_t$ uses the following features from the two trading venues:

- **Pool current status:** pool rate Z_t, tick $\mathcal{Z}(i)$, market depth κ, and gas price.
- **Past returns, volatility, and fees:** percentage change between the current price and prices 5, 10, and 30 minutes before, and the volatility of the time series of these returns over the past 180, 360, and 720 minutes. Total fees earned by LPs and the pool fee rate of the past 180, 360, and 720 minutes.
- **LP activity:** total USD amount deposited and withdrawn by LPs over the past 10 and 30 minutes. Total number of LPs transactions and the net USD amount over the same time horizons.
- **Binance activity:** total USD volume of buy and sell orders and the number of liquidity taking orders in Binance over the past 30 minutes. Total USD amount of limit orders posted in Binance and current spread between Binance and Uniswap v3.
- **Signals:** trading flow imbalance in Uniswap v3 and Binance. Moving average convergence/divergence (MACD) signal and exponential moving average of the MACD signal with periods 9, 14, and 26; see C.1.

6.6.3 Results

Next, we use market data from Uniswap v3 and Binance to showcase the performance of the LSTM strategy of Section 6.4. We train an LSTM network of five LSTM layers with 120 neurons each. The dropout rate is 0.1 and we use the Adam optimiser for training. Note that in Section 6.5, each LSTM layer has 24 neurons. Here, we increase the number of neurons per layer because we use more input features and because market data are more noisy than the simulated data, so the LSTM network needs more parameters to learn the optimal strategy.

We split our dataset into non-overlapping daily time series and further split our dataset between train, validation, and test data. Then, to avoid look ahead bias, we use the first 30% of daily time series to train the model, the next 30% to validate the model, and the remaining 40% to test the model. In Section 6.5, we use 60% of the data for training, 20% for validation, and 20% for testing. In this section, a higher percentage of the dataset is used for testing because the amount of market data available is low and we need to increase the number of datapoints in the test set to ensure that there is enough variability among them. Indeed, we want the empirical distribution of the test set to be as close as possible to the empirical distribution of the entire dataset to ensure that the performance of the LSTM strategy on the test set is unbiased.

In our experiments, we consider three values for the initial wealth of the LP and five values for her trading frequency Δt, and use Sharpe Ratio and the mean-variance as criteria to train the LSTM network. We execute the LSTM strategy for five choices of trading frequency and three initial wealth positions. Figure 6.4 shows the returns of the LSTM strategy, before and after gas fees, and Table 6.11 reports the results.

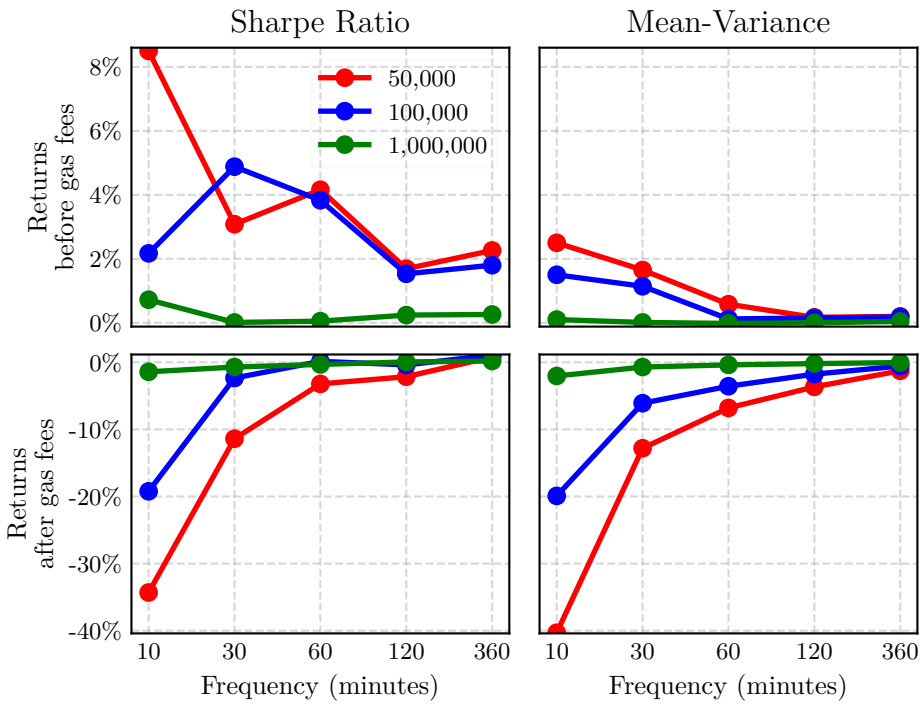


Figure 6.4: **Left**: average returns of the LSTM strategy before (top) and after (bottom) gas fees using Sharpe Ratio to train the LSTM network. **Right**: average returns of the LSTM strategy before (top) and after (bottom) gas fees using the mean-variance criterion to train the LSTM network.

Overall, Table 6.11 shows that gas fees have a considerable impact on the performance of the strategy. The average returns of the LSTM strategy before gas fees are negative only for one of our experiments. In contrast, the average returns after gas fees are positive only in 5 of the 30 experiments. Specifically, returns after gas fees are negative for all the experiments where we use the mean-variance criterion to train the LSTM network.

Figure 6.4 shows that average returns are a decreasing function of the LP's initial wealth. This follows from the LP's remuneration rule in (6.5) and the definition of the LP's market depth $\tilde{\kappa}$, which is a linear function of the LP's wealth; see (6.7). The remuneration rule in (6.5) says that for each fee p paid by LTs, the LP earns a percentage of p equal to

$$\frac{\tilde{\kappa}}{\kappa + \tilde{\kappa}}, \tag{6.11}$$

where κ is the aggregated market depth of the other LPs. Note that (6.11) is an increasing function of $\tilde{\kappa}$ which is bounded by 1 and grows sub-linearly. Therefore, from (6.5), (6.7), (6.11), it follows that the LP's absolute returns per trade are bounded by the amount of fees paid by LTs and percentage returns are a decreasing function of the initial wealth.

In contrast, Figure 6.4 shows that average returns after gas fees are an increasing function of the LP's initial wealth because gas fees are a flat fee and thus they impact small values of the initial wealth more. In

Average returns before gas fees are highest and equal to 8.50% when the LP's initial wealth is 50,000 USDC, the trading frequency is $\Delta t = 10$ minutes and Sharpe Ratio is used as performance

criterion. Here, 50,000 USDC and $\Delta t = 10$ are, respectively, the lowest and the highest values for the initial wealth and the trading frequency. After gas fees, average returns drop to -34.31%, which is the lowest value for returns after gas fees when the LP maximises Sharpe Ratio. This noticeable drop in the performance of the strategy is due to the relatively low initial wealth and the high frequency at which she readjusts her positions; when $\Delta t = 10$ minutes, the LP adjusts her position too frequently and thus pays more gas fees. The relative impact of gas fees is maximised because her initial wealth is the at the minimum value considered in our experiments.

Sharpe Ratio is highest and equal to 0.34 when the LP's initial wealth is 100,000 USDC, the trading frequency is $\Delta t = 360$ minutes, and Sharpe Ratio is used as performance criterion. This shows that gas fees are a bottleneck for dynamic liquidity provision in Uniswap v3.

Performance Criterion	**Frequency**	**Initial Wealth (USDC)**	**Returns (after gas fees)**		**Sharpe Ratio (after gas fees)**	
Sharpe Ratio	10min	50,000	**8.50%**	(−34.31%)	1.24	(−5.01)
	10min	100,000	2.17%	(−19.23%)	0.91	(−8.07)
	10min	1,000,000	0.72%	(−1.42%)	1.01	(−1.99)
	30min	50,000	3.08%	(−11.39%)	0.92	(−3.39)
	30min	100,000	4.88%	(−2.35%)	**1.37**	(−0.66)
	30min	1,000,000	0.01%	(−0.71%)	0.01	(−0.51)
	60min	50,000	4.16%	(−3.22%)	1.06	(−0.82)
	60min	100,000	3.83%	(0.14%)	1.19	(0.04)
	60min	1,000,000	0.05%	(−0.32%)	0.03	(−0.19)
	120min	50,000	1.69%	(−2.15%)	0.47	(−0.61)
	120min	100,000	1.53%	(−0.39%)	0.51	(−0.13)
	120min	1,000,000	0.24%	(0.05%)	0.17	(0.03)
	360min	50,000	2.26%	(0.79%)	0.60	(0.21)
	360min	100,000	1.80%	(**1.06%**)	0.58	(**0.34**)
	360min	1,000,000	0.26%	(0.19%)	0.18	(0.13)
Mean-Variance	10min	50,000	2.50%	(−40.30%)	0.67	(−10.82)
	10min	100,000	1.50%	(−19.91%)	0.82	(−10.88)
	10min	1,000,000	0.10%	(−2.04%)	0.33	(−6.72)
	30min	50,000	1.65%	(−12.81%)	0.81	(−6.31)
	30min	100,000	1.14%	(−6.10%)	1.16	(−6.24)
	30min	1,000,000	0.01%	(−0.71%)	0.04	(−2.14)
	60min	50,000	0.58%	(−6.80%)	0.41	(−4.80)
	60min	100,000	0.12%	(−3.57%)	0.24	(−7.10)
	60min	1,000,000	−0.02%	(−0.39%)	−0.06	(−1.03)
	120min	50,000	0.17%	(−3.67%)	0.21	(−4.56)
	120min	100,000	0.15%	(−1.77%)	0.22	(−2.66)
	120min	1,000,000	−0.00%	(−0.20%)	−0.01	(−0.45)
	360min	50,000	0.20%	(−1.27%)	0.21	(−1.34)
	360min	100,000	0.18%	(−0.56%)	0.20	(−0.62)
	360min	1,000,000	0.04%	(−0.04%)	0.06	(−0.07)

Table 6.11: Performance of the LSTM strategy on test set of market data.

Appendix A

Appendix for Chapter 3

A.1 Proof of Theorem 1

Recall that for each fixed values of N and j

$$\nu^{\star,j,N}\left(t,\tilde{y},Z,S\right) = -\frac{1}{\eta\,\zeta_N^j}A_{j,N}(t)\,\tilde{y} + \frac{1}{2\,\eta\,\zeta_N^j}B_{j,N}(t)(S-Z)\ ,$$

where

$$\zeta_N^j := \frac{1}{\kappa}\left(Z_j^N\right)^{3/2}\ ,$$

and

$$A_{j,N}(t) := A_{\zeta_j^N}(t) = \sqrt{\phi\,\eta\,\zeta_N^j}\tanh\left(\frac{\sqrt{\phi}}{\sqrt{\eta\,\zeta_N^j}}t + \operatorname{arctanh}\left(-\frac{\alpha}{\sqrt{\phi\,\eta\,\zeta_N^j}}\right)\right)\ ,$$

$$B_{j,N}(t) := B_{\zeta_j^N}(t) = \int_0^t \beta\exp\left(\int_s^t\left(\beta - \frac{1}{\eta\,\zeta_N^j}A_{j,N}(u)\right)du\right)ds\ .$$

Moreover, recall that

$$A(t,Z) = \sqrt{\frac{\phi\,\eta\,Z^{3/2}}{\kappa}}\tanh\left(\frac{\sqrt{\phi\,\kappa}}{\sqrt{\eta\,Z^{3/2}}}t + \operatorname{arctanh}\left(-\frac{\alpha\sqrt{\kappa}}{\sqrt{\phi\,\eta\,Z^{3/2}}}\right)\right)\ ,$$

$$B(t,Z) = \int_0^t \beta\exp\left(\int_s^t\left(\beta - \frac{\kappa}{\eta\,Z^{3/2}}A(u,Z)\right)du\right)ds\ .$$

To prove (3.25), take $(t,\tilde{y},S)$ and write

$$\begin{aligned}
&\left|\nu^{\star,j,N}\left(t,\tilde{y},Z_{j+1}^N,S\right) - \nu^{\star,j+1,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)\right| \\
&\quad= \left|-\frac{1}{\eta\,\zeta_N^j}A_{j,N}(t)\,\tilde{y} + \frac{1}{\eta\,\zeta_N^{j+1}}A_{j+1,N}(t)\,\tilde{y} + (S-Z_{j+1}^N)\left(\frac{1}{2\,\eta\,\zeta_N^j}B_{j,N}(t) - \frac{1}{2\,\eta\,\zeta_N^{j+1}}B_{j+1,N}(t)\right)\right| \\
&\quad\le \frac{|\tilde{y}|}{\eta}\left|-\frac{1}{\zeta_N^j}A_{j,N}(t) + \frac{1}{\zeta_N^{j+1}}A_{j+1,N}(t)\right| + \frac{|S|+\overline{Z}}{\eta}\left|\frac{1}{2\,\zeta_N^j}B_{j,N}(t) - \frac{1}{2\,\zeta_N^{j+1}}B_{j+1,N}(t)\right| \\
&\quad= \frac{|\tilde{y}|}{\eta}\left|-\frac{\kappa}{\left(Z_j^N\right)^{3/2}}A\left(t,Z_j^N\right) + \frac{\kappa}{\left(Z_{j+1}^N\right)^{3/2}}A\left(t,Z_{j+1}^N\right)\right| \\
&\qquad+ \frac{|S|+\overline{Z}}{2\,\eta}\left|\frac{\kappa}{\left(Z_j^N\right)^{3/2}}B\left(t,Z_j^N\right) - \frac{\kappa}{\left(Z_{j+1}^N\right)^{3/2}}B\left(t,Z_{j+1}^N\right)\right|\ .
\end{aligned}$$

Observe that for a fixed $t \in [0,T]$ the functions

$$Z \mapsto \frac{\kappa}{Z^{3/2}} A(t,Z) \quad \text{and} \quad Z \mapsto \frac{\kappa}{Z^{3/2}} B(t,Z)$$

are uniformly continuous on $\left[\underline{Z}, \overline{Z}\right]$ because they are both compositions of continuous functions defined over a closed interval. By definition of the partition in (3.24), $\left|Z_j^N - Z_{j+1}^N\right| = 1/N$ so for each $\varepsilon > 0$ there exists $N \in \mathbb{N}$ such that

$$\max_{j=1,\dots,N} \left|\nu^{\star,j,N}\left(t,\tilde{y},Z_{j+1}^N,S\right) - \nu^{\star,j+1,N}\left(t,\tilde{y},Z_{j+1}^N,S\right)\right| \leq \varepsilon\,.$$

To prove that $\{\tilde{\nu}^{\star,N}\}$ converges uniformly to $\tilde{\nu}^\star$ in $[0,T] \times \mathbb{R} \times \left[\underline{Z}, \overline{Z}\right] \times \mathbb{R}$, take $(t,\tilde{y},Z,S) \in [0,T] \times \mathbb{R}^2 \times \left[\underline{Z}, \overline{Z}\right] \times \mathbb{R}$, and $N \in \mathbb{N}$, and observe that there exists $\bar{j} \in \{1,\dots,N\}$ such that $Z \in \left[Z_{\bar{j}}^N, Z_{\bar{j}+1}^N\right)$ and thus

$$\begin{aligned}
&\left|\tilde{\nu}_N^*\left(t,\tilde{y},Z,S\right) - \tilde{\nu}^*\left(t,\tilde{y},Z,S\right)\right| \\
&\quad = \left|\tilde{\nu}_{\bar{j},N}^*\left(t,\tilde{y},Z,S\right) - \tilde{\nu}^*\left(t,\tilde{y},Z,S\right)\right| \\
&\quad = \left|-\frac{1}{\eta\,\zeta_N^{\bar{j}}} A_{\bar{j},N}(t)\,\tilde{y} + \frac{\kappa}{\eta\,Z^{3/2}} A(t,Z)\,\tilde{y} + (S-Z)\left(\frac{1}{2\,\eta\,\zeta_N^{\bar{j}}} B_{\bar{j},N}(t) - \frac{\kappa}{2\,\eta\,Z^{3/2}} B(t,Z)\right)\right| \\
&\quad \leq \frac{|\tilde{y}|\,\kappa}{\eta\,\underline{Z}^{3/2}} \left|-A\left(t,Z_{\bar{j}}^N\right) + A(t,Z)\right| + \frac{|S| + \overline{Z}}{2\,\eta\,\underline{Z}^{3/2}}\,\kappa \left|B\left(t,Z_{\bar{j}}^N\right) - B(t,Z)\right|\,.
\end{aligned}$$

The uniform convergence of $\{\tilde{\nu}_N^*\}$ to $\tilde{\nu}^*$ follows from the uniform continuity of $A(t,Z)$ and $B(t,Z)$ on $[0,T] \times \left[\underline{Z}, \overline{Z}\right]$. □

A.2 Example for the liquidation strategy

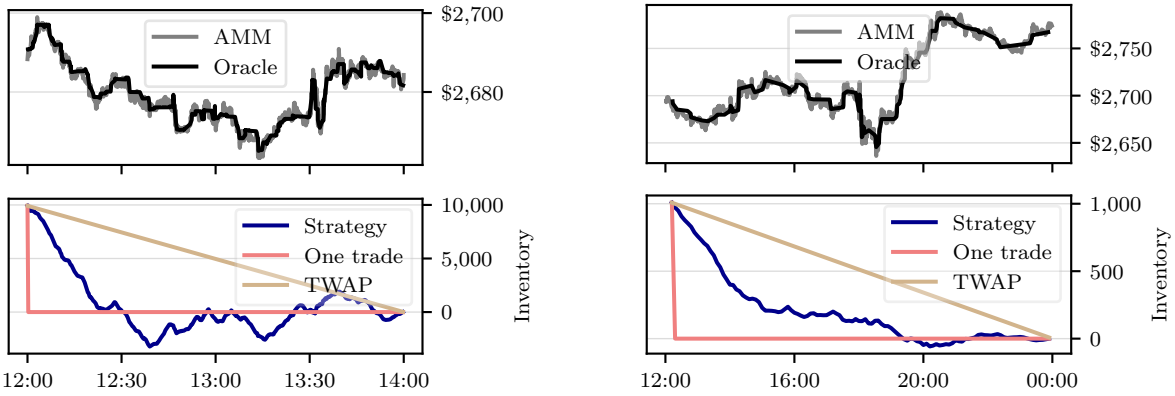
In this appendix, we describe the parameters and strategy performance for a specific run of the liquidation strategy. Assume the LT will start trading at noon on 16 March 2022, so she uses the data between noon 15 March 2022 and noon 16 March to estimate model parameters.

For the 24 hours before noon 16 March 202, there are, on average, one liquidity taking order every 13 seconds in the liquid pool and one every 360 seconds in the illiquid pool; i.e., the time steps in the regressions (3.41) are $\Delta t = 13$ for ETH/USDC and $\Delta t = 360$ for ETH/DAI. Table A.1 shows parameter estimates.

	ETH/USDC	ETH/DAI
$\hat{\sigma}$	0.045 day$^{-1/2}$	0.053 day$^{-1/2}$
$\hat{\gamma}$	0.034 day$^{-1/2}$	0.027 day$^{-1/2}$
$\hat{\beta}$	657.9 day^{-1}	14.78 day^{-1}

Table A.1: Parameter estimates for dynamics of Z and S with data between noon 15 March 2022 and noon 16 March 2022.

The parameter η of the execution costs in (3.7) is also set to 13 seconds $= 17.3 \times 10^{-5}$ days and 360 seconds $= 41 \times 10^{-4}$ days for the liquid and illiquid pool, respectively. The number of observed



(a) **Top**: Out-of-sample instantaneous and oracle rates for the pair ETH/USDC. **Bottom**: Inventory process $\tilde{y}$ for the optimal, TWAP, and single order strategies.

(b) **Top**: Out-of-sample instantaneous and oracle rates for the pair ETH/DAI. **Bottom**: Inventory process $\tilde{y}$ for the optimal, TWAP, and single order strategies.

Figure A.1: Liquidation strategies starting at noon on 16 March 2022.

transactions in the in-sample data is approximately 238,039 ETH and 4,031 ETH in the liquid and illiquid pool, respectively. Thus the LTs' target is to liquidate 14,877 and 1,007 units of ETH within 2 and 12 hours in the ETH/USDC and ETH/DAI pools, respectively. Table A.2 summarises all the parameters used to run our strategy.

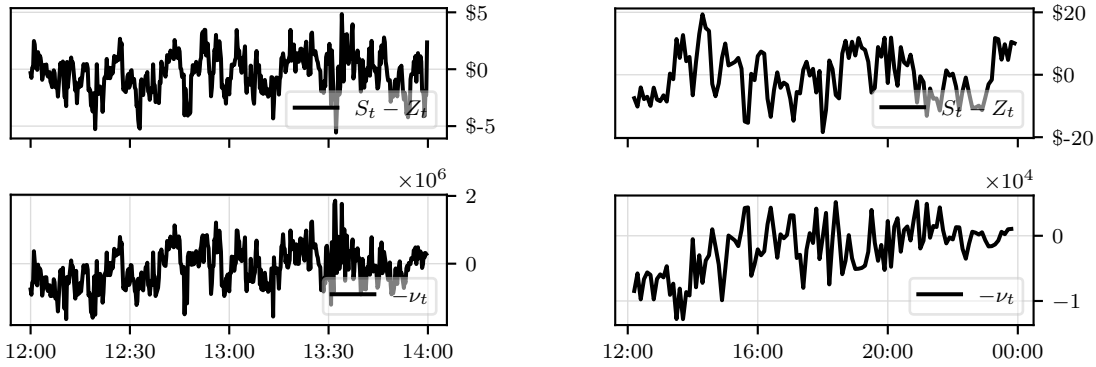
	ETH/USDC	ETH/DAI
κ_0	22,561,783	1,666,175
$\tilde{y}_0$	14,877 ETH	1,007 ETH
$\tilde{S}_0$	2,689.2 USDC	2,686.09 DAI
$\tilde{Z}_0$	2,690.77 USDC	2,694.04 DAI
η	17.3×10^{-5} days	41×10^{-4} days

Table A.2: Values of model parameters.

Figure A.1 shows the instantaneous and oracle rates and the inventories of the strategies during the execution, for both ETH/USDC and ETH/DAI. Figure A.1 clearly showcases the difference between the strategies. In particular, the liquidation strategy is adaptive and trades on the difference between the two rates S and Z during the liquidation programme. Figure A.2 shows how the difference $S_t - Z_t$ drives the trading speed ν_t. The oracle rate is used as a predictive signal for future moves of the instantaneous rate.

A.3 Model I with stochastic convexity cost

We consider the same problem as that of Section 3.3.1 where an LT exchanges a large position in asset Y into asset X in a CPMM or executes a statistical arbitrage over a period of time $[0, T]$, where $T > 0$. She uses the rate Z in (3.12) from the pool and the rate S in (3.8) from another more liquid exchange. The depth κ of the pool is constant and the LT liquidates a position $\tilde{y}_0 \in \mathbb{R}$



(a) **Top**: Difference between the oracle and the instantaneous rate for ETH/USDC. **Bottom**: $-\nu_t$.

(b) **Top**: Difference between the oracle and the instantaneous rate for ETH/DAI. **Bottom**: $-\nu_t$.

Figure A.2: Trading speed.

in asset Y. Her wealth is valued in terms of asset X and has dynamics (3.11), where the execution cost is stochastic and its dynamics are known. During the trading programme, the LT trades at the speed $(\nu_t)_{t\in[0,T]}$, so the inventory $(\tilde{y}_t)_{t\in[0,T]}$ evolves as in (3.10), where we do not restrict the speed to be positive, and for simplicity, trading fees are set to zero.

The LT maximises her expected terminal wealth in units of X while penalising inventory. The set of admissible strategies is defined in (3.13) and the LT's performance criterion is a function $u^\nu\colon [0,T]\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_{++}\times\mathbb{R}_{++}\to\mathbb{R}$ defined in (3.14). The value function $u:[0,T]\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_{++}\times\mathbb{R}_{++}\to\mathbb{R}$ of the LT is given by

$$u(t,\tilde{x},\tilde{y},Z,S)=\sup_{\nu\in\mathcal{A}}\{u^\nu(t,\tilde{x},\tilde{y},Z,S)\}\ . \tag{A.1}$$

The value function solves the Hamilton–Jacobi–Bellman (HJB) equation

$$\begin{aligned}0=&\,\partial_t w-\phi\,\tilde{y}^2+\beta\,(S-Z)\,\partial_Z w+\frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}w+\frac{1}{2}\sigma^2\,S^2\,\partial_{SS}w\\&+\sup_{\nu\in\mathbb{R}}\left(\left(\nu\,Z-\frac{\eta}{\kappa}\,Z^{3/2}\,\nu^2\right)\partial_{\tilde{x}}w-\nu\,\partial_{\tilde{y}}w\right)\ ,\end{aligned} \tag{A.2}$$

with terminal condition

$$w(T,\tilde{x},\tilde{y},Z,S)=\tilde{x}+\tilde{y}\,Z-\alpha\,\tilde{y}^2\ . \tag{A.3}$$

The form of the terminal condition (A.3) suggests the ansatz

$$w(T,\tilde{x},\tilde{y},Z,S)=\tilde{x}+\tilde{y}\,Z+\theta(t,\tilde{y},Z,S)\ . \tag{A.4}$$

Now replace (A.4) into (A.2) to obtain the PDE

$$\begin{aligned}0\ =&\,\partial_t\theta-\phi\,\tilde{y}^2+\beta\,(S-Z)\,(\tilde{y}+\partial_Z\theta)+\frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta+\frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta\\&+\sup_{\nu\in\mathbb{R}}\left(-\frac{\eta}{\kappa}\,Z^{3/2}\,\nu^2-\nu\,\partial_{\tilde{y}}\theta\right)\ ,\end{aligned} \tag{A.5}$$

defined over $[0,T) \times \mathbb{R} \times \mathbb{R}^\star \times \mathbb{R}$, with terminal condition

$$\theta(T,\tilde{y},Z,S) = -\alpha\,\tilde{y}^2 \ .$$

The first two terms on the right-hand side of (A.4) are the mark-to-market value of the LT's holdings and the last term is the additional value that the LT obtains by following the optimal strategy. Next, solve the first order condition in (A.5) to obtain the optimal trading speed in feedback form

$$\nu^\star = -\frac{\kappa}{2\,\eta}\,Z^{-3/2}\,\partial_{\tilde{y}}\theta \ , \tag{A.6}$$

and substitute (A.6) into (A.5) to write

$$\partial_t\theta = -\phi\,\tilde{y}^2 + \beta\,(S-Z)\,(\tilde{y}+\partial_Z\theta) + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta + \frac{\kappa}{4\,\eta}\,Z^{-3/2}\,\partial_{\tilde{y}}\theta^2 \ . \tag{A.7}$$

Finally, we propose the ansatz

$$\theta(t,\tilde{y},Z,S) = \theta_2(t,Z,S)\,\tilde{y}^2 + \theta_1(t,Z,S)\,\tilde{y} + \theta_0(t,Z,S) \ . \tag{A.8}$$

Now substitute (A.8) into (A.7) to obtain the system of PDEs

$$\begin{cases} -\,\partial_t\theta_2 = -\phi + \beta\,(S-Z)\,\partial_Z\theta_2 + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta_2 + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta_2 + \frac{\kappa}{\eta}\,Z^{-3/2}\,\theta_2^2 \ , \\ -\,\partial_t\theta_1 = \beta\,(S-Z)\,(1+\partial_Z\theta_1) + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta_1 + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta_1 + \frac{\kappa}{\eta}\,Z^{-3/2}\,\theta_2\,\theta_1 \ , \\ -\,\partial_t\theta_0 = \beta\,(S-Z)\,\partial_Z\theta_0 + \frac{1}{2}\gamma^2\,Z^2\,\partial_{ZZ}\theta_0 + \frac{1}{2}\sigma^2\,S^2\,\partial_{SS}\theta_0 + \frac{\kappa}{4\,\eta}\,Z^{-3/2}\,\theta_1^2 \ , \end{cases} \tag{A.9}$$

defined over $[0,T) \times \mathbb{R}^\star \times \mathbb{R}$, with terminal condition

$$\theta_2(T,Z,S) = -\alpha \ , \qquad \theta_1(T,Z,S) = 0 \ , \qquad \theta_0(T,Z,S) = 0 \ . \tag{A.10}$$

The optimal strategy in feedback form (A.6) is given by

$$\nu^\star = -\frac{\kappa}{2\,\eta}\,Z^{-3/2}\,\left(2\,\theta_2\,\tilde{y} + \theta_1\right) \ . \tag{A.11}$$

The next result provides a-priori bounds for the solution to the system of PDEs (A.9).

Theorem 5. *Assume there exist $\theta_0 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$, $\theta_1 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$, and $\theta_2 \in C^{1,2,2}([0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++})$ solution to the system of PDEs (A.9) with terminal condition (A.10). Define θ in (A.8) and define w in (A.4). Assume that for all $(t,\tilde{x},\tilde{y},Z,\kappa) \in [0,T]\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}_{++}\times\mathbb{R}_{++}$ we have $u^\nu(t,\tilde{x},\tilde{y},Z,\kappa) \le w\,(t,\tilde{x},\tilde{y},Z,\kappa)$, where u^ν is defined in (3.14), and that equality is obtained for the optimal control $(\nu^\star)_{t\in[0,T]}$ in feedback form in (A.11). Then θ_0, θ_1, and θ_2 have the following bounds for all $(t,Z,S) \in [0,T]\times\mathbb{R}_{++}\times\mathbb{R}_{++}$:*

$$\begin{cases} \theta_0\,(t,Z,S) \ge 0 \\ \theta_0\,(t,Z,S) \le A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 - Z + \mathbb{E}\,[Z_T] \ , \end{cases}$$

$$\begin{cases} \theta_1\,(t,Z,S) \ge -\,(\alpha+\phi\,(T-t)) - A(t)\,S^2 - \frac{1}{2}B(t)\,S\,Z - C(t)\,Z^2 \\ \theta_1\,(t,Z,S) \le A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 + \alpha + \phi\,(T-t) \ , \end{cases}$$

$$\begin{cases} \theta_2\,(t,Z,S) \ge -\,(\alpha+\phi\,(T-t)) \\ \theta_2\,(t,Z,S) \le A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 \ , \end{cases}$$

where $\mathbb{E}\,[Z_T]$ is in (A.13) and A, B, and C are in (A.18).

Proof. We provide a-priori bounds for the solutions θ_2, θ_1, θ_0 of (A.9).

Lower bound for θ By definition of our control problem in (A.1) we know that for any control $\underline{\nu} \in \mathcal{A}$, we have

$$u^{\underline{\nu}}(t, \tilde{x}, \tilde{y}, Z, S) \leq \sup_{\nu \in \mathcal{A}} u^{\nu}(t, \tilde{x}, \tilde{y}, Z, S) \ .$$

Use the definition of θ in (A.4) and the form of the performance criterion to write

$$\mathbb{E}\left[\tilde{x}_T^{\underline{\nu}} + \tilde{y}_T^{\underline{\nu}} Z_T - \alpha \left(\tilde{y}_T^{\underline{\nu}}\right)^2 - \phi \int_t^T \left(\tilde{y}_s^{\underline{\nu}}\right)^2 ds\right] \leq \tilde{x} + \tilde{y}\, Z + \theta(t, \tilde{y}, Z, S) \ .$$

Next, consider a sub-optimal strategy consisting in keeping a constant inventory with no trading, i.e., a strategy defined by $\underline{\nu}_s = 0$ for each $s \geq t$. Then the inventory process is $\left(\tilde{y}_s^{\underline{\nu}}\right)_{s \geq t} = \tilde{y}$. Use the inequality above to write

$$\tilde{x} + \tilde{y}\, Z + \theta(t, \tilde{y}, Z, S) \geq \mathbb{E}\left[\tilde{x} + \tilde{y}\, Z - \tilde{y}\,(Z - Z_T) - \alpha\, \tilde{y}^2 - \phi\, \tilde{y}^2\,(T - t)\right],$$

and conclude that

$$\theta(t, \tilde{y}, Z, S) \geq -\tilde{y}\, Z - \tilde{y}\mathbb{E}\left[Z_T\right] - \alpha\, \tilde{y}^2 - \phi\, \tilde{y}^2\,(T - t) > -\infty\,, \tag{A.12}$$

which provides a lower bound for θ . The last inequality in (A.12) is a consequence of

$$\mathbb{E}\begin{bmatrix} S_T \\ Z_T \end{bmatrix} = \begin{pmatrix} S \\ Z \end{pmatrix} e^{A(T-t)}\,, \quad \text{where} \quad A = \begin{pmatrix} 0 & 0 \\ -\theta & \theta \end{pmatrix}\,,$$

which simplifies to

$$\mathbb{E}\left[Z_T\right] = Z\, e^{-\theta(T-t)} + \theta\, S \int_t^T e^{-\theta(T-s)} ds = Z\, e^{-\theta(T-t)} + S\left(1 - e^{-\theta\,(T-t)}\right) \ . \tag{A.13}$$

Upper bound for θ Use integration by parts to write, for all $(t, V, \tilde{y}, Z, S)$,

$$\begin{aligned}
\tilde{x} + \tilde{y}\, Z + \theta(t, \tilde{y}, Z, S) =& \sup_{\nu \in \mathcal{A}_t} \mathbb{E}\left[\tilde{x}_T + \tilde{y}_T Z - \alpha\, \tilde{y}_T^2 - \phi \int_t^T \tilde{y}_s^2\, ds\right] \\
=& \tilde{x} + \tilde{y}\, Z + \sup_{\nu \in \mathcal{A}_t} \mathbb{E}\left[\int_t^T \tilde{y}_s\, dZ_s - \frac{\eta}{\kappa} \int_t^T Z_s^{3/2}\, v_s^2\, ds - \alpha\, \tilde{y}_T^2 - \phi \int_t^T \tilde{y}_s^2\, ds\right] \\
\leq& \tilde{x} + \tilde{y}\, Z + \sup_{\nu \in \mathcal{A}_t} \mathbb{E}\left[\int_t^T \tilde{y}_s\, dZ_s - \phi \int_t^T \tilde{y}_s^2\, ds\right],
\end{aligned} \tag{A.14}$$

where the first equality is by definition of our control problem and θ, the second equality uses integration by parts, and the last inequality holds because the terms $\frac{\eta}{\kappa} \int_t^T Z_s^{3/2}\, v_s^2\, ds$ and $\alpha\, \tilde{y}_T^2$ are always positive.

Let $t \in [0, T]$. Define the set

$$\mathcal{A}_t^M = \left\{\left(\tilde{y}_s^M\right)_{s \in [t,T]}\,,\ \mathbb{R}\text{–valued},\ \mathbb{F}\text{–adapted and such that } \int_t^T \left|\tilde{y}_u^M\right|^2 du < +\infty,\ \ \mathbb{P}\text{–a.s.}\right\}. \tag{A.15}$$

Clearly, one has

$$\sup_{\nu \in \mathcal{A}_t} \mathbb{E}\left[\int_t^T \tilde{y}_s\, dZ_s - \phi \int_t^T \tilde{y}_s^2 ds\right] \leq \sup_{\tilde{y}^M \in \mathcal{A}_t^M} \mathbb{E}\left[\int_t^T \tilde{y}_s^M\, dZ_s - \phi \int_t^T \left(\tilde{y}_s^M\right)\, ds\right].$$

The term on the right-hand side of the inequality is a classical Merton problem, where the rate Z follows the dynamics in (3.12). We introduce the wealth process $\left(\tilde{x}_t^M\right)_{t\in[0,T]}$ with dynamics

$$d\tilde{x}_t^M = \tilde{y}_t^M \, dZ_t \,,$$

In the Merton problem, the LT controls her inventory $\tilde{y}^M$, and her strategy solves the optimization problem

$$\overline{u}(t,\tilde{x},Z,S) = \sup_{\tilde{y}\in\mathcal{A}_t^M} \mathbb{E}\left[\tilde{x}_T^M - \phi \int_t^T \left(\tilde{y}_s^M\right)^2 ds\right] = \sup_{\tilde{y}\in A_t^M} \mathbb{E}\left[\int_t^T \tilde{y}_s^M \, dZ_s - \phi \int_t^T \left(\tilde{y}_s^M\right)^2 ds\right] . \tag{A.16}$$

We introduce the value function $\overline{w}$ associated with the optimisation problem (A.16) and which solves the HJB

$$\begin{aligned} 0 &= \partial_t \overline{w} + \beta\,(S-Z)\,\partial_Z \overline{w} + \frac{1}{2}\gamma^2 Z^2 \partial_{ZZ}\overline{w} + \frac{1}{2}\sigma^2 S^2 \partial_{SS}\overline{w} \\ &\quad + \sup_{\tilde{y}^M\in\mathbb{R}} \left(\beta\,(S-Z)\,\tilde{y}^M \partial_{\tilde{x}}\overline{w} + \frac{1}{2}\gamma^2 Z^2 \left(\tilde{y}^M\right)^2 \partial_{\tilde{x}\tilde{x}}\overline{w} + \gamma^2 Z^2 \tilde{y}\,\partial_{\tilde{x}Z}\overline{w} - \phi\left(\tilde{y}^M\right)^2\right), \end{aligned}$$

with terminal condition $\overline{w}\left(T,\tilde{x}^M,Z,S\right) = \tilde{x}^M$. Use the ansatz $\overline{w}(t,\tilde{x}^M,Z,S) = \tilde{x}^M + \overline{\theta}(t,Z,S)$ to obtain the PDE

$$0 = \partial_t\overline{\theta} + \beta\,(S-Z)\,\partial_Z\overline{\theta} + \frac{1}{2}\gamma^2 Z^2 \partial_{ZZ}\overline{\theta} + \frac{1}{2}\sigma^2 S^2 \partial_{SS}\overline{\theta} + \frac{\beta^2 (S-Z)^2}{4\phi} ,$$

with terminal condition $\overline{\theta}(T,Z,S) = 0$. The optimal control is

$$\tilde{y}^{M\star} = \frac{\beta\,(S-Z)}{2\,\phi} . \tag{A.17}$$

Use the ansatz

$$\overline{\theta}(t,Z,S) = A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 + D(t)\,S + E(t)\,Z ,$$

to obtain the following system of ODEs:

$$\begin{cases} -A'(t) = \sigma^2 A(t) + \dfrac{\beta^2}{4\,\phi} + \dfrac{1}{2}\beta\,B(t) , \\ -B'(t) = -\beta\,B(t) - \dfrac{\beta^2}{\phi} + 4\,\beta\,C(t) , \\ -C'(t) = \left(\gamma^2 - 2\,\beta\right)\,C(t) + \dfrac{\beta^2}{4\,\phi} , \\ -D'(t) = \beta\,E(t) , \\ -E'(t) = -\beta\,E(t) , \end{cases}$$

with terminal conditions $A(T) = B(T) = C(T) = D(T) = E(T) = 0$. The solution is

$$\begin{cases} A(t) = \displaystyle\int_t^T e^{\sigma^2\,(u-t)}\,(\frac{1}{2}\beta B(u) + \frac{\beta^2}{4\,\phi})\,du , \\ B(t) = \displaystyle\int_t^T e^{-\beta\,(u-t)}(4\,\beta\,C(u) - \frac{\beta^2}{\phi})\,du , \\ C(t) = \dfrac{-\beta^2}{4\,\phi\,(\gamma^2 - 2\,\beta)}\left(1 - e^{\left(\gamma^2-2\,\beta\right)\,(T-t)}\right) , \\ D(t) = 0 , \\ E(t) = 0 . \end{cases} \tag{A.18}$$

Note that we obtain a classical solution to the Merton problem so standard results apply. In particular, one only needs to verify that the optimal control is admissible, which is clear from the optimal control (A.17) and the admissible set (A.15). Finally, write the inequalities in (A.12) and (A.14) as

$$A(t)\,S^2 + \frac{1}{2}\,B(t)\,S\,Z + C(t)\,Z^2 \geq \theta(t,\tilde{y},Z,S) \geq -\tilde{y}\,\left(-Z + \mathbb{E}\left[Z_T\right]\right) - \left(\alpha + \phi\,\left(T-t\right)\right)\,\tilde{y}^2\,. \quad \text{(A.19)}$$

Lower and upper bounds for θ_0, θ_1, and θ_2 From the à-priori bounds on θ given by the original control problem, we deduce à-priori bounds on the solutions θ_0, θ_1, and θ_2 of the system (A.9). Recall that

$$\theta(t,\tilde{y},Z,S) = \theta_2(t,Z,S)\,\tilde{y}^2 + \theta_1(t,Z,S)\,\tilde{y} + \theta_0(t,Z,S)\,.$$

First, notice that the lower bound is in the form of a quadratic polynomial in $\tilde{y}$, so $\theta_2\left(t,Z,S\right) \geq -\left(\alpha + \phi\left(T-t\right)\right)$. Next, for the upper bound, see that it holds for any value of the inventory $\tilde{y}$. In particular, observe that for $\tilde{y} = 0$ we find that

$$A(t)\,S^2 + \frac{1}{2}B(t)\,SZ + C(t)\,Z^2 \geq \theta_0\left(t,Z,S\right) \geq 0\,. \quad \text{(A.20)}$$

Moreover, if we let $\tilde{y} = 1$ and $\tilde{y} = -1$ in (A.19) one obtains

$$2A(t)\,S^2 + B(t)\,S\,Z + 2C(t)\,Z^2 \geq 2\theta_0\left(t,Z,S\right) + 2\theta_2(t,Z,S) \geq -2\left(\alpha + \phi\,\left(T-t\right)\right)\,.$$

Finally, use (A.20) to write

$$A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 \geq \theta_2\left(t,Z,S\right) \geq -\left(\alpha + \phi\left(T-t\right)\right).$$

Next, substitute $\tilde{y} = 1$ into (A.19) to write

$$\begin{aligned}\theta_1(t,Z,S) \geq& -Z + \mathbb{E}\left[Z_T\right] - \left(\alpha + \phi\left(T-t\right)\right) - \theta_2\left(t,Z,S\right) - \theta_0(t,Z,S)\\ \geq& -Z + \mathbb{E}\left[Z_T\right] - \left(\alpha + \phi\left(T-t\right)\right) - \left(A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2\right)\,,\end{aligned}$$

and

$$\begin{aligned}\theta_1(t,Z,S) \leq& A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 - \theta_2\left(t,Z,S\right) - \theta_0(t,Z,S)\\ \leq& A(t)\,S^2 + \frac{1}{2}B(t)\,S\,Z + C(t)\,Z^2 + \alpha + \phi\left(T-t\right)\,.\end{aligned}$$

Thus, provided θ exists, then θ_0, θ_1, θ_2 have à-priori upper and lower bounds. The bounds are all at most linear in time, and quadratic in $\{S,\ Z\}$.

□

Appendix B

Appendix to Chapter 5

B.1 Proof of Proposition 5

To solve the problem (5.22), we introduce an equivalent control problem. First, define the process $(\tilde{\pi}_t)_{t\in[0,T]} = (\pi_t - \eta_t)_{t\in[0,T]}$ with dynamics

$$d\tilde{\pi}_t = \Gamma\,(\overline{\pi} - \tilde{\pi}_t)\,dt + \psi\,\sqrt{\tilde{\pi}_t}\,dB_t\,,$$

where $\tilde{\pi}_0 = \pi_0 - \eta_0$ and η is in (5.19).

We introduce the performance criterion $\tilde{u}^\delta\colon [0,T]\times\mathbb{R}^4 \to \mathbb{R}$ given by

$$\tilde{u}^\delta(t,v,z,\pi,\mu) = \mathbb{E}_{t,v,z,\tilde{\pi},\mu}\left[\log\left(V_T^\delta\right)\right]\,,$$

and the value function $\tilde{u} : [0,T]\times\mathbb{R}^4 \to \mathbb{R}$ given by

$$\tilde{u}(t,v,z,\tilde{\pi},\mu) = \sup_{\delta\in\mathcal{A}} u^\delta(t,v,z,\pi,\mu)\,. \tag{B.1}$$

Clearly, the problems (B.1) and (5.22) are equivalent, and the value functions satisfy $u(t,v,z,\pi,\mu) = \tilde{u}(t,v,z,\tilde{\pi},\mu)$ for all $(t,v,z,\pi,\mu)\in[0,T]\times\mathbb{R}^4$ and for all $\tilde{\pi} = \pi - \eta \in \mathbb{R}$, where $\eta = \frac{\sigma^2}{8} - \frac{\mu}{4}\left(\mu - \frac{\sigma^2}{2}\right) + \frac{\varepsilon}{4}$.

The value function in (B.1) admits the dynamic programming principle, so it satisfies the HJB equation

$$\begin{aligned} 0 = {} & \partial_t w + \frac{1}{2}\sigma^2\,z^2\,\partial_{zz}w + \mu\,Z\,\partial_z w + \Gamma\,(\overline{\pi} - \tilde{\pi})\,\partial_{\tilde{\pi}}w + \frac{1}{2}\,\psi^2\,\tilde{\pi}\,\partial_{\tilde{\pi}\tilde{\pi}}w + \mathcal{L}^\mu w \\ & + \sup_{\delta\in\mathbb{R}^+}\left(\frac{1}{\delta}\left(4\,\tilde{\pi} + 4\,\eta - \frac{\sigma^2}{2}\right)v\,\partial_v w + \mu\,\rho\,(\delta,\mu)\;v\,\partial_v w + \frac{1}{2}\,\sigma^2\,\rho\,(\delta,\mu)^2\;v^2\,\partial_{vv}w\right. \\ & \left. - \frac{\gamma}{\delta^2}\,v\,\partial_v w + \sigma^2\,\rho\,(\delta,\mu)\;v\,z\,\partial_{vz}w\right)\,, \end{aligned} \tag{B.2}$$

with terminal condition

$$w(T,v,z,\tilde{\pi},\mu) = \log(v)\,,\quad \forall\,(v,z,\tilde{\pi},\mu)\in\mathbb{R}^4\,,$$

where $\mathcal{L}^\mu$ is the infinitesimal generator of μ.

To study the HJB in (B.2), use the ansatz

$$w\left(t,v,z,\tilde{\pi},\mu\right)=\log\left(v\right)+\theta\left(t,z,\tilde{\pi},\mu\right)\,,$$

to obtain the HJB

$$\begin{aligned}0=&\,\partial_t\theta+\frac{1}{2}\sigma^2\,z^2\,\partial_{zz}\theta+\mu\,Z\,\partial_z\theta+\Gamma\left(\overline{\pi}-\tilde{\pi}\right)\partial_{\tilde{\pi}}w+\frac{1}{2}\,\psi^2\,\tilde{\pi}\,\partial_{\tilde{\pi}\tilde{\pi}}w+\frac{\mu}{2}-\frac{1}{8}\,\sigma^2\\&+\mathcal{L}^{\mu}\theta+\sup_{\delta\in\mathbb{R}^+}\left(\frac{1}{\delta}\left(4\,\tilde{\pi}+4\,\eta-\frac{\sigma^2}{2}\right)+\frac{\mu^2}{\delta}-\frac{1}{2}\,\sigma^2\,\left(\frac{\mu^2}{\delta^2}+\frac{\mu}{\delta}\right)-\frac{\gamma}{\delta^2}\right),\end{aligned}\tag{B.3}$$

with terminal condition

$$\theta\left(T,z,\tilde{\pi},\mu\right)=0\,,\quad\forall\left(z,\tilde{\pi},\mu\right)\in\mathbb{R}^3\,.$$

The supremum in the HJB (B.3) is attained at

$$\delta^{\star}=\frac{2\,\gamma+\mu^2\,\sigma^2}{4\,\tilde{\pi}+4\,\eta-\frac{\sigma^2}{2}+\mu\left(\mu-\frac{\sigma^2}{2}\right)}=\frac{2\,\gamma+\mu^2\,\sigma^2}{4\,\tilde{\pi}+\varepsilon}\,.$$

Thus, (B.3) becomes

$$0=\partial_t\theta+\frac{1}{2}\sigma^2\,z^2\,\partial_{zz}\theta+\mu\,Z\,\partial_z\theta+\Gamma\left(\overline{\pi}-\tilde{\pi}\right)\partial_{\tilde{\pi}}\theta+\frac{1}{2}\,\psi^2\,\tilde{\pi}\,\partial_{\tilde{\pi}\tilde{\pi}}\theta+\frac{\mu}{2}-\frac{\sigma^2}{8}+\mathcal{L}^{\mu}\theta+\frac{1}{2}\frac{\left(4\,\tilde{\pi}+\varepsilon\right)^2}{2\,\gamma+\mu^2\,\sigma^2}\,.$$

Next, substitute the ansatz

$$\begin{aligned}\theta\left(t,z,\tilde{\pi},\mu\right)=&\,A\left(t,\mu\right)z^2+B\left(t,\mu\right)\tilde{\pi}\,z+C\left(t,\mu\right)\tilde{\pi}^2\\&+D\left(t,\mu\right)z+E\left(t,\mu\right)\tilde{\pi}+F\left(t,\mu\right)\,,\end{aligned}$$

in (B.3), collect the terms in Z and $\tilde{\pi}$, and write the following system of PDEs:

$$\begin{cases}\left(\partial_t+\mathcal{L}^{\mu}\right)A\left(t,\mu\right)=-\,\sigma^2\,A\left(t,\mu\right)-2\,\mu\,A\left(t,\mu\right)\,,\\\left(\partial_t+\mathcal{L}^{\mu}\right)B\left(t,\mu\right)=-\,\mu\,B\left(t,\mu\right)+\Gamma B\left(t,\mu\right)\,,\\\left(\partial_t+\mathcal{L}^{\mu}\right)C\left(t,\mu\right)=2\,C\left(t,\mu\right)\Gamma-\dfrac{8}{2\,\gamma+\mu^2\,\sigma^2}\,,\\\left(\partial_t+\mathcal{L}^{\mu}\right)D\left(t,\mu\right)=-\,\mu\,D\left(t,\mu\right)-\Gamma\,\overline{\pi}\,B\left(t,\mu\right)\,,\\\left(\partial_t+\mathcal{L}^{\mu}\right)E\left(t,\mu\right)=-\,2\,\Gamma\,\overline{\pi}\,C\left(t,\mu\right)-\psi^2\,C\left(t,\mu\right)+\Gamma\,E\left(t,\mu\right)-\dfrac{4\,\varepsilon}{2\,\gamma+\sigma^2\,\mu^2}\,,\\\left(\partial_t+\mathcal{L}^{\mu}\right)F\left(t,\mu\right)=-\,\Gamma\,\overline{\pi}\,E\left(t,\mu\right)+\psi^2\,\eta\,C\left(t,\mu\right)-\dfrac{1}{2}\dfrac{\varepsilon^2}{2\,\gamma+\sigma^2\,\mu^2}-\dfrac{\mu}{2}+\dfrac{\sigma^2}{8}\,,\end{cases}$$

with terminal conditions $A(T,\mu)=B(T,\mu)=C(T,\mu)=D(T,\mu)=E(T,\mu)=F(T,\mu)=0$ for all $\mu\in\mathbb{R}$.

First, note that the PDEs in A, B, and D admit the unique solutions $A=B=D=0$. Next, we solve the PDE in C. Use Itô's lemma to write

$$C\left(T,\mu_T\right)=C\left(t,\mu_t\right)+\int_t^T\left(\partial_t+\mathcal{L}^{\mu}\right)C\left(s,\mu_s\right)ds\,.$$

Next, replace $(\partial_t + \mathcal{L}^\mu)\, C\,(s, \mu_s)$ with $2\, C\,(s, \mu_s)\, \Gamma - \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2}$ to obtain

$$C\,(T, \mu_T) = C\,(t, \mu_t) + 2\,\Gamma \int_t^T C\,(s, \mu_s)\, ds - \int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2}\, ds\,.$$

Take expectations to get the equation

$$C\,(t, \mu_t) = \mathbb{E}_{t,\mu}\left[-2\,\Gamma \int_t^T C\,(s, \mu_s)\, ds + \int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2}\, ds\right].$$

Now consider the candidate solution function

$$\hat{C}\,(t, \mu_t) = \mathbb{E}_{t,\mu}\left[\int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2} \exp\left(-2\,\Gamma\,(s - t)\right)\, ds\right]$$

and write

$$\begin{aligned}
&\mathbb{E}_{t,\mu}\left[-2\,\Gamma \int_t^T \hat{C}\,(s, \mu_s)\, ds + \int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2}\, ds\right] \\
=&\mathbb{E}_{t,\mu}\left[-2\,\Gamma \int_t^T \mathbb{E}_{s,\mu}\left[\int_s^T \frac{8}{2\,\gamma + \mu_u^2\, \sigma^2} \exp\left(-2\,\Gamma\,(u - s)\right)\, du\right] ds + \int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2}\, ds\right] \\
=&\mathbb{E}_{t,\mu}\left[\int_t^T \frac{8}{2\,\gamma + \mu_s^2\, \sigma^2} \exp\left(-2\,\Gamma\,(s - t)\right)\, ds\right].
\end{aligned}$$

Thus, $\hat{C}$ is a solution to the equation in C and by uniqueness of solutions, we conclude that $C = \hat{C}$.

Follow the same steps as above to obtain the solution

$$E\,(t, \mu) = \mathbb{E}_{t,\mu}\left[\int_t^T \left(\left(2\,\Gamma\,\overline{\pi} + \psi^2\right) C\,(s, \mu) + \frac{4\,\varepsilon}{2\,\gamma + \sigma^2\, \mu_s^2}\right) \exp\left(-\Gamma\,(s - t)\right)\, ds\right]$$

to the PDE in E, and the solution

$$F\,(t, \mu) = \mathbb{E}_{t,\mu}\left[\int_t^T \left(\Gamma\,\overline{\pi}\, E\,(s, \mu_s) + \psi^2\, \eta_s\, C\,(s, \mu_s) - \frac{1}{2}\frac{\varepsilon^2}{2\,\gamma + \sigma^2\, \mu_s^2} - \frac{\mu_s}{2} + \frac{\sigma^2}{8}\right) ds\right]$$

to the PDE in F, where $\eta_s = \frac{\sigma^2}{8} - \frac{\mu_s}{4}\left(\mu_s - \frac{\sigma^2}{2}\right) + \frac{\varepsilon}{4}$, which proves the result. □

B.2 Proof of Theorem 4

Proposition 5 provides a classical solution to (B.2). Therefore, standard results apply and showing that (3.20) is an admissible control is enough to prove that (5.23) is the value function (5.22). Specifically, use the form of the optimal control $\delta^\star$ in (3.20) to obtain

$$0 < \frac{1}{\delta_s^\star} = \frac{\tilde{\pi}_s + \varepsilon}{\sigma^2 \mu_s^2 + 2\,\gamma} \leq \frac{\tilde{\pi}_s + \varepsilon}{2\,\gamma}\,, \qquad \forall s \in [t, T]\,,$$

where $\tilde{\pi}_s = \pi_s - \eta_s$, thus $\delta^\star$ is an admissible control. □

Appendix C

Appendix for Chapter 6

C.1 Moving average convergence/divergence

In this section, we formalise the *moving average convergence/divergence* (MACD) signal, which is one of the features that we mention in Section 6.6.2 and use to train the LSTM network in Section 6.6.3. The MACD signal is a trading indicator commonly used in technical analysis; see Baz et al. (2015); Cartea et al. (2023c).

First, we recall the definition of exponential moving average. Let $\{x_t\}_{t\in\mathbb{N}}$ be a real valued time series and let s be a positive natural number. The *exponential moving average* $\left\{E^{(s)}(x_t)\right\}_{t\in\mathbb{N}}$ of $\{x_t\}_{t\in\mathbb{N}}$ with span s is given by

$$\begin{cases} E^{(s)}(x_0) := x_0\,, \\ E^{(s)}(x_t) := \dfrac{s-1}{s+1}\,E^{(s)}(x_{t-1}) + \dfrac{2}{s+1}\,x_t\,. \end{cases}$$

Then, we define the MACD signal $\{I(x_t)\}_{t\in\mathbb{N}}$ as

$$I(x_t) := E^{(12)}(x_t) - E^{(26)}(x_t)\ .$$

Finally, for each $n \in \mathbb{N}$, the MACD $I^{(n)}(x_t)$ with period n is

$$I^{(n)}(x_t) := E^{(n)}(I(x_t))\ .$$

Bibliography

Adams, H., Zinsmeister, N., Salem, M., Keefer, R., Robinson, D., 2021. Uniswap v3 core. Technical Report.

Alfonsi, A., Fruth, A., Schied, A., 2010. Optimal execution strategies in limit order books with general shape functions. Quantitative Finance 10, 143–157.

Alfonsi, A., Schied, A., 2010. Optimal trade execution and absence of price manipulations in limit order book models. SIAM Journal on Financial Mathematics 1, 490–522.

Almgren, R., 2012. Optimal trading with stochastic liquidity and volatility. SIAM Journal on Financial Mathematics 3, 163–181.

Almgren, R., Chriss, N., 2000. Optimal execution of portfolio transactions. Journal of Risk 3, 5–39.

Angeris, G., Agrawal, A., Evans, A., Chitra, T., Boyd, S., 2022a. Constant function market makers: Multi-asset trades via convex optimization, in: Handbook on Blockchain. Springer, pp. 415–444.

Angeris, G., Chitra, T., 2020. Improved price oracles: Constant function market makers, in: Proceedings of the 2nd ACM Conference on Advances in Financial Technologies, Association for Computing Machinery, New York, NY, USA. p. 80–91.

Angeris, G., Chitra, T., Evans, A., Boyd, S., 2022b. Optimal routing for constant function market makers.

Angeris, G., Kao, H.T., Chiang, R., Noyes, C., Chitra, T., 2021. An analysis of Uniswap markets. Cryptoeconomic Systems 0.

Avellaneda, M., Stoikov, S., 2008. High frequency trading in a limit order book. Quantitative Finance 8, 217–224.

Balch, T.H., Mahfouz, M., Lockhart, J., Hybinette, M., Byrd, D., 2019. How to evaluate trading strategies: Single agent market replay or multiple agent interactive simulation? .

Bank, P., Voß, M., 2019. Optimal investment with transient price impact. SIAM Journal on Financial Mathematics 10, 723–768.

Barger, W., Lorig, M., 2019. Optimal liquidation under stochastic price impact. International Journal of Theoretical and Applied Finance 22, 1850059.

Barzykin, A., Bergault, P., Gueant, O., 2021. Market making by an fx dealer: tiers, pricing ladders and hedging rates for optimal risk control.

Barzykin, A., Bergault, P., Gueant, O., 2022. Dealing with multi-currency inventory risk in fx cash markets.

Barzykin, A., Bergault, P., Guéant, O., 2023. Algorithmic market making in dealer markets with hedging and market impact. Mathematical Finance 33, 41–79.

Barzykin, A., Bergault, P., Guéant, O., 2024. Algorithmic market making in spot precious metals.

Baz, J., Granger, N., Harvey, C.R., Le Roux, N., Rattray, S., 2015. Dissecting investment strategies in the cross section and time series.

Bechler, K., Ludkovski, M., 2015. Optimal execution with dynamic order flow imbalance. SIAM Journal on Financial Mathematics 6, 1123–1151.

Becker, S., Cheridito, P., Jentzen, A., 2020. Pricing and hedging american-style options with deep learning. Journal of Risk and Financial Management 13.

Bengio, Y., Simard, P., Frasconi, P., 1994. Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council 5, 157–66.

Berg, J.A., Fritsch, R., Heimbach, L., Wattenhofer, R., 2022. An empirical study of market inefficiencies in Uniswap and Sushiswap, in: Matsuo, S., Gudgeon, L., Klages-Mundt, A., Perez Hernandez, D., Werner, S., Haines, T., Essex, A., Bracciali, A., Sala, M. (Eds.), Financial Cryptography and Data Security. FC 2022 International Workshops, pp. 238–249.

Bergault, P., Bertucci, L., Bouba, D., Guéant, O., 2023. Automated market makers: Mean-variance analysis of lps payoffs and design of pricing functions. Digital Finance .

Bergault, P., Drissi, F., Guéant, O., 2022. Multi-asset optimal execution and statistical arbitrage strategies under ornstein–uhlenbeck dynamics. SIAM Journal on Financial Mathematics 13, 353–390.

Bergault, P., Evangelista, D., Guéant, O., Vieira, D., 2021. Closed-form approximations in multi-asset market making. Applied Mathematical Finance 28, 101–142.

Bergault, P., Guéant, O., 2021. Size matters for otc market makers: General results and dimensionality reduction techniques. Mathematical Finance 31, 279–322.

Bergault, P., Sánchez-Betancourt, L., 2024. A mean field game between informed traders and a broker. Available at SSRN .

Bertsimas, D., Lo, A., 1998. Optimal control of execution costs. Journal of Financial Markets 1, 1–50.

Biais, B., 1993. Price formation and equilibrium liquidity in fragmented and centralized markets. The Journal of Finance 48, 157–185.

Buehler, H., Gonon, L., Teichmann, J., Wood, B., 2019. Deep hedging. Quantitative Finance 19, 1271–1291.

Buterin, V., 2014. Ethereum: A next-generation smart contract and decentralized application platform.

Campi, L., Zabaljauregui, D., 2020. Optimal market making under partial information with general intensities. Applied Mathematical Finance 27, 1–45.

Capponi, A., Jia, R., 2021. The adoption of blockchain-based decentralized exchanges. arXiv preprint arXiv:2103.08842 .

Cartea, A., Arribas, I.P., Sánchez-Betancourt, L., 2022a. Double-execution strategies using path signatures. SIAM Journal on Financial Mathematics 13, 1379–1417.

Cartea, Á., Donnelly, R., Jaimungal, S., 2017. Algorithmic trading with model uncertainty. SIAM Journal on Financial Mathematics 8, 635–671.

Cartea, Á., Donnelly, R., Jaimungal, S., 2018a. Enhancing trading strategies with order book signals. Applied Mathematical Finance 25, 1–35.

Cartea, Á., Drissi, F., Monga, M., 2022b. Decentralised finance and automated market making: Execution and speculation.

Cartea, Á., Drissi, F., Monga, M., 2022c. Decentralised finance and automated market making: Predictable loss and optimal liquidity provision. Available at SSRN 4273989 .

Cartea, Á., Drissi, F., Monga, M., 2023a. Execution and statistical arbitrage with signals in multiple automated market makers. Available at SSRN .

Cartea, Á., Drissi, F., Monga, M., 2023b. Predictable losses of liquidity provision in constant function markets and concentrated liquidity markets. Available at SSRN 4541034 .

Cartea, Á., Drissi, F., Osselin, P., 2023c. Bandits for algorithmic trading with signals.

Cartea, Á., Drissi, F., Sánchez-Betancourt, L., Siska, D., Szpruch, L., 2023d. Automated market makers designs beyond constant functions. Available at SSRN 4459177 .

Cartea, Á., Gan, L., Jaimungal, S., 2018b. Trading co-integrated assets with price impact. Mathematical Finance 29.

Cartea, Á., Jaimungal, S., 2014. Optimal execution with limit and market orders. Quantitative Finance 15, 1279 – 1291.

Cartea, Á., Jaimungal, S., 2016. Incorporating order-flow into optimal execution. Mathematics and Financial Economics 10, 339–364.

Cartea, Á., Jaimungal, S., Penalva, J., 2015. Algorithmic and High-Frequency Trading. Cambridge University Press.

Cartea, Á., Jaimungal, S., Ricci, J., 2014. Buy low, sell high: A high frequency trading perspective. SIAM Journal on Financial Mathematics 5, 415–444.

Cartea, Á., Jaimungal, S., Sánchez-Betancourt, L., 2021. Latency and liquidity risk. International Journal of Theoretical and Applied Finance 24, 2150035.

Cartea, Á., Jaimungal, S., Sánchez-Betancourt, L., 2023e. Reinforcement learning for algorithmic trading, in: Lehalle, C.A., Capponi, A. (Eds.), Machine Learning and Data Sciences for Financial Markets: A Guide to Contemporary Practices. Cambridge University Press.

Cartea, Á., Sánchez-Betancourt, L., 2021. The shadow price of latency: Improving intraday fill ratios in foreign exchange markets. SIAM Journal on Financial Mathematics 12, 254–294.

Cartea, Á., Sánchez-Betancourt, L., 2022a. Brokers and informed traders: dealing with toxic flow and extracting trading signals. Available at SSRN .

Cartea, Á., Sánchez-Betancourt, L., 2022b. Optimal execution with stochastic delay. Finance and Stochastics .

Cartea, Á., Wang, Y., 2020. Market making with alpha signals. International Journal of Theoretical and Applied Finance 23, 2050016.

Chan, N.T., Shelton, C., 2001. An Adaptive Electronic Market-Maker. Computing in Economics and Finance 2001 146. Society for Computational Economics.

Cheridito, P., Sepin, T., 2014. Optimal trade execution under stochastic volatility and liquidity. Applied Mathematical Finance 21, 342–362.

Chiu, J., Koeppl, T.V., 2019. Blockchain-based settlement for asset trading. The Review of Financial Studies 32, 1716–1753.

Daian, P., Goldfeder, S., Kell, T., Li, Y., Zhao, X., Bentov, I., Breidenbach, L., Juels, A., 2020. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability., in: IEEE Symposium on Security and Privacy, IEEE. pp. 910–927.

Dickey, D.A., Fuller, W.A., 1979. Distribution of the estimators for autoregressive time series with a unit root. Journal of the American Statistical Association 74, 427–431.

Diebold, F.X., Yilmaz, K., 2012. Better to give than to receive: Predictive directional measurement of volatility spillovers. International Journal of Forecasting 28, 57–66.

Donnelly, R., 2022. Optimal execution: A review. Applied Mathematical Finance 29, 181–212.

Donnelly, R., Lorig, M., 2020. Optimal trading with differing trade signals. Applied Mathematical Finance 27, 317–344.

Drissi, F., 2022. Solvability of differential riccati equations and applications to algorithmic trading with signals. Applied Mathematical Finance 29, 457–493.

Drissi, F., 2023. Models of market liquidity: Applications to traditional markets and automated market makers. Available at SSRN 4424010 .

Egorov, M., 2019. Stableswap - efficient mechanism for stablecoin liquidity.

Fan, Z., Marmolejo-Cossio, F., Moroz, D.J., Neuder, M., Rao, R., Parkes, D.C., 2021. Strategic liquidity provision in Uniswap v3. arXiv preprint arXiv:2106.12033 .

Fan, Z., Marmolejo-Cossío, F., Altschuler, B., Sun, H., Wang, X., Parkes, D.C., 2022. Differential liquidity provision in Uniswap v3 and implications for contract design.

Fischer, T., Krauss, C., 2018. Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research 270, 654–669.

Forde, M., Sánchez-Betancourt, L., Smith, B., 2022. Optimal trade execution for Gaussian signals with power-law resilience. Quantitative Finance 22, 585–596.

Forsyth, P., Kennedy, J., Tse, S., Windcliff, H., 2012. Optimal trade execution: A mean quadratic variation approach. Journal of Economic Dynamics and Control 36, 1971–1991.

Fouque, J.P., Jaimungal, S., Saporito, Y.F., 2022. Optimal trading with signals and stochastic price impact. SIAM J. Fin. Math. 13, 944–968.

Fukasawa, M., Maire, B., Wunsch, M., 2023. Model-free hedging of impermanent loss in geometric mean market makers. arXiv preprint arXiv:2303.11118 .

Gatheral, J., Schied, A., 2013. Dynamical models of market impact and algorithms for order execution. Handbook on Systemic Risk, Jean-Pierre Fouque, Joseph A. Langsam, eds , 579–599.

Gatheral, J., Schied, A., Slynko, A., 2010. Transient linear price impact and fredholm integral equations. Math. Finance 22.

Gašperov, B., Begušić, S., Posedel Šimović, P., Kostanjčar, Z., 2021. Reinforcement learning approaches to optimal market making. Mathematics 9.

Gašperov, B., Kostanjčar, Z., 2021. Market making with signals through deep reinforcement learning. IEEE Access 9, 61611–61622.

Giudici, P., Pagnottoni, P., 2020. Vector error correction models to measure connectedness of bitcoin exchange markets. Applied Stochastic Models in Business and Industry 36, 95–109.

Glosten, L.R., Milgrom, P., 1985. Bid, ask and transaction prices in a specialist market with heterogeneously informed traders. Journal of Financial Economics 14, 71–100.

Goodfellow, I.J., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, Cambridge, MA, USA.

Goodman, L.M., 2014. Tezos : A self-amending crypto-ledger position paper.

Goyal, M., Ramseyer, G., Goel, A., Mazières, D., 2023. Finding the right curve: Optimal design of constant function market makers, in: Proceedings of the 24th ACM Conference on Economics and Computation, pp. 783–812.

Graewe, P., Horst, U., Séré, E., 2018. Smooth solutions to portfolio liquidation problems under price-sensitive market impact. Stochastic Processes and their Applications 128, 979–1006.

Granger, C.W.J., 1969. Investigating causal relations by econometric models and cross-spectral methods. Econometrica 37, 424–438.

Guéant, O., 2017. Optimal market making. Applied Mathematical Finance 24, 112–154.

Guéant, O., 2023. Reinforcement Learning Methods in Algorithmic Trading. Cambridge University Press. chapter 11. pp. 182 – 204.

Gueant, O., Lehalle, C.A., Fernandez-Tapia, J., 2012. Dealing with the inventory risk: A solution to the market making problem. Mathematics and Financial Economics 7.

Guéant, O., 2016. The Financial Mathematics of Market Liquidity: From Optimal Execution to Market Making. Chapman & Hall.

Guéant, O., Manziuk, I., 2019. Deep reinforcement learning for market making in corporate bonds: Beating the curse of dimensionality. Applied Mathematical Finance 26, 387–452.

Heimbach, L., Schertenleib, E., Wattenhofer, R., 2022. Risks and returns of Uniswap v3 liquidity providers.

Heimbach, L., Wang, Y., Wattenhofer, R., 2021. Behavior of liquidity providers in decentralized exchanges.

Hendricks, D., Wilcox, D., 2014. A reinforcement learning extension to the almgren-chriss framework for optimal trade execution, in: 2014 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr), pp. 457–464.

Herdegen, M., Muhle-Karbe, J., Stebegg, F., 2023. Liquidity provision with adverse selection and inventory costs. Mathematics of Operations Research 48, 1286–1315.

Hertzog, E., Benartzi, G., Benartzi, G., 2017. Bancor protocol.

Hey, N., Bouchaud, J.P., Mastromatteo, I., Muhle-Karbe, J., Webster, K., 2023. The cost of misspecifying price impact.

Ho, T., Stoll, H., 1981. Optimal dealer pricing under transactions and return uncertainty. Journal of Financial Economics 9, 47–73.

Ho, T.S., Stoll, H.R., 1983. The dynamics of dealer markets under competition. The Journal of Finance 38, 1053–1074.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Computation 9, 1735–1780.

Jerome, J., Sánchez-Betancourt, L., Savani, R., Herdegen, M., 2023. Mbt-gym: Reinforcement learning for model-based limit order book trading, in: Proceedings of the Fourth ACM International Conference on AI in Finance, Association for Computing Machinery, New York, NY, USA. p. 619–627.

Johansen, S., 1991. Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models. Econometrica: journal of the Econometric Society , 1551–1580.

Jusselin, P., 2021. Optimal market making with persistent order flow. SIAM Journal on Financial Mathematics 12, 1150–1200.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization, in: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Koop, G., Pesaran, M.H., Potter, S.M., 1996. Impulse response analysis in nonlinear multivariate models. Journal of Econometrics 74, 119–147.

Kyle, A., 1985. Continuous auctions and insider trading. Econometrica 53, 1315–35.

Lehalle, C.A., Neuman, E., 2019. Incorporating signals into optimal trading. Finance and Stochastics 23.

Lı, T., Naik, S., Papanicolaou, A., Schönleber, L., 2023. Yield farming for liquidity provision .

Limmer, Y., Horvath, B., 2023. Robust hedging gans.

Lipton, A., Treccani, A., 2021. Blockchain and Distributed Ledgers: Mathematics, Technology, and Economics. World Scientific.

Lommers, K., Kim, J., Skidan, B., 2023. The case for stochastically dynamic AMMs. Preprint.

Lorenz, C., Schied, A., 2013. Drift dependence of optimal trade execution strategies under transient price impact. Finance and Stochastics 17.

Lütkepohl, H., 1991. Introduction to Multiple Time Series Analysis. Springer Berlin.

Martinelli, F., Mushegian, N., 2019. Whitepaper.

Milionis, J., Moallemi, C.C., Roughgarden, T., 2023. Automated market making and arbitrage profits in the presence of fees. arXiv preprint arXiv:2305.14604 .

Milionis, J., Moallemi, C.C., Roughgarden, T., Zhang, A.L., 2022. Automated market making and loss-versus-rebalancing. arXiv preprint arXiv:2208.06046 .

Moghar, A., Hamiche, M., 2020. Stock market prediction using lstm recurrent neural network. Procedia Computer Science 170, 1168–1173. The 11th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops.

Muhle-Karbe, J., Wang, Z., Webster, K., 2024. Stochastic liquidity as a proxy for nonlinear price impact. Operations Research 72, 444–458.

Nelson, D.M.Q., Pereira, A.C.M., de Oliveira, R.A., 2017. Stock market's price movement prediction with lstm neural networks, in: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 1419–1426.

Neuman, E., Voß, M., 2020. Optimal signal-adaptive trading with temporary and transient price impact.

Nevmyvaka, Y., Feng, Y., Kearns, M., 2006. Reinforcement learning for optimized trade execution, in: ICML 2006 - Proceedings of the 23rd International Conference on Machine Learning, pp. 673–680.

Ning, B., Lin, F.H.T., Jaimungal, S., 2021. Double deep q-learning for optimal execution. Applied Mathematical Finance 28, 361–380.

Obizhaeva, A., Wang, J., 2013. Optimal trading strategy and supply/demand dynamics. Journal of Financial Markets 16, 1–32.

Park, A., 2021. The conceptual flaws of constant product automated market making. SSRN Electronic Journal .

Pesaran, H., Shin, Y., 1998. Generalized impulse response analysis in linear multivariate models. Economics Letters 58, 17–29.

Schied, A., Schoneborny, T., 2009. Risk aversion and the dynamics of optimal liquidation strategies in illiquid. Finance and Stochastics 13.

Spooner, T., Fearnley, J., Savani, R., Koukorinis, A., 2018. Market making via reinforcement learning, in: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC. p. 434–442.

Spooner, T., Savani, R., 2021. Robust market making via adversarial reinforcement learning, in: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence.

Waldon, H., Drissi, F., Limmer, Y., Berdica, U., Foerster, J.N., Cartea, A., 2024. DARE: The deep adaptive regulator for control of uncertain continuous-time systems, in: ICML 2024 Workshop: Foundations of Reinforcement Learning and Control – Connections and Perspectives. URL: https://openreview.net/forum?id=vTtldsicU9.

Xu, J., Paruch, K., Cousaert, S., Feng, Y., 2021. SoK: Decentralized Exchanges (DEX) with Automated Market Maker (AMM) Protocols. Papers. arXiv.org.

Yakovenko, A., 2014. Solana: A new architecture for a high performance blockchain.

Zhang, X., Liang, X., Zhiyuli, A., Zhang, S., Xu, R., Wu, B., 2019. At-lstm: An attention-based lstm model for financial time series prediction. IOP Conference Series: Materials Science and Engineering 569, 052037.

Zhou, L., Qin, K., Torres, C., Le, D., Gervais, A., 2021. High-frequency trading on decentralized on-chain exchanges, in: IEEE Symposium on Security and Privacy, IEEE. pp. 428–445.