DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Market Making Mechanisms and Liquidity Dynamics in Blockchain-Based Prediction Markets

Parshant Singh

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Market Making Mechanisms and Liquidity Dynamics in Blockchain-Based Prediction Markets

Market-Making-Mechanismen und Liquiditätsdynamiken in Blockchain-basierten Prognosemärkte

Author:	Parshant Singh
Supervisor:	Jonas Gebele
Advisor:	Prof. Dr. Florian Matthes
Submission Date:	3rd September, 2025

I confirm that this master's thesis in informatics is my own work and I have documented all sources and material used.

Munich, 3rd September, 2025 Parshant Singh

AI Assistant Usage Disclosure

Introduction

Performing work or conducting research at the Chair of Software Engineering for Business Information Systems (sebis) at TUM often entails dynamic and multi-faceted tasks. At sebis, we promote the responsible use of *AI Assistants* in the effective and efficient completion of such work. However, in the spirit of ethical and transparent research, we require all student researchers working with sebis to disclose their usage of such assistants.

For examples of correct and incorrect AI Assistant usage, please refer to the original, unabridged version of this form, located at this link.

Use of *AI Assistants* for Research Purposes

I have used AI Assistant(s) for the purposes of my research as part of this thesis.

Yes ☑ No ☐

Explanation:
In this research, AI assistants are responsibly utilized to support writing and code debugging, specifically by refining written content to meet scientific standards and by effectively troubleshooting implemented algorithms.

Munich, 03.09.2025 Parshant Singh

Location, Date Author

Acknowledgments

The completion of this thesis has been made possible by the support of many people. I would like to express my deepest gratitude to my advisor, Jonas Gebele, for his invaluable guidance, support, and encouragement throughout the course of this research. I am also sincerely thankful to Prof. Dr. Florian Matthes for providing me with the opportunity to conduct this thesis at the Sebis Chair. Finally, I am grateful to my family for their unwavering support and encouragement throughout my academic journey.

Abstract

Prediction markets have increasingly emerged as alternatives to traditional polling methods. Decentralised designs, in particular, reduce reliance on central operators and enable more transparent forecasting. A persistent challenge, however, lies in liquidity provision. Automated Market Makers (AMMs), while popular in decentralised finance, suffer from impermanent loss, an issue that is especially severe in prediction markets, since one of the outcome tokens becomes worthless once the event resolves. This study focuses on *Polymarket*, a leading decentralised prediction market built on Polygon, and compares two market designs it has employed: AMM-based and Central Limit Order Book (CLOB)-based systems. Our analysis shows that liquidity provision in the CLOB-based design is significantly more profitable than in the AMM-based design, where most potential profits are eroded by realised impermanent loss. We also find that LP strategies differ across the two systems. In AMMs, providers often use Just-in-Time (JiT) liquidity and remove liquidity before market resolution to reduce exposure to impermanent loss. In CLOBs, LPs dynamically adjust bid–ask quotes to manage inventory risk while maximising platform rewards. These results highlight how market design directly impacts LP behaviour and profitability, and offer insights for the development of more sustainable prediction markets.

Contents

Acknowledgments **iv**

Abstract **v**

1. Introduction **1**
1.1. Background and Motivation . . . 1
1.2. Problem Statement . . . 2
1.3. Research Questions . . . 2

2. Background **4**
2.1. Polygon . . . 4
2.2. Blockchain Oracles . . . 4
2.3. Decentralised Finance . . . 5
2.3.1. Automated Market Maker (AMM) . . . 5
2.3.2. Impermanent Loss . . . 6
2.4. Central Limit Order Book (CLOB) . . . 8
2.5. Gnosis Conditional Token Framework (CTF) . . . 9
2.5.1. ERC-1155 Token Standard . . . 10

3. Prediction Market Structure **12**
3.1. Prediction Market Mechanisms . . . 12
3.1.1. Price Discovery . . . 12
3.1.2. Liquidity Provision . . . 12
3.1.3. Market Resolution . . . 13
3.2. Centralised Prediction Markets . . . 13
3.3. Decentralised Prediction Markets . . . 13
3.3.1. Polymarket . . . 14

4. Methodology **19**
4.1. Data Gathering . . . 19
4.2. FPMM Data Analysis . . . 21
4.2.1. Impermanent Loss Calculation . . . 21
4.2.2. P&L Calculation . . . 22
4.3. CLOB Data Analysis . . . 23
4.3.1. Spread Capture P&L Calculations . . . 23
4.3.2. Rewards Distribution . . . 24

5. Results **25**
5.1. AMM . . . 25
5.2. Central Limit Order Book (CLOB) . . . 32
5.3. Research Questions . . . 34
5.3.1. RQ1: Profitability Factors in AMMs . . . 34
5.3.2. RQ2: Strategy Optimization in AMMs . . . 35
5.3.3. RQ3: Profitability in CLOB Market Making . . . 36

6. Discussion **37**

7. Conclusion **40**

8. Future Work **41**

A. Apendix **42**
A.1. Top 20 Most Profitable Makers by spread capture (CLOB) . . . 42
A.2. Top 10 LPs by Net Profit (AMM) . . . 43
A.3. Top 10 LPs by Fees (AMM) . . . 43
A.4. Top Rewards-Earning Addresses . . . 44

List of Figures **45**

List of Tables **46**

Acronyms **47**

Bibliography **48**

1. Introduction

Political betting markets date back as early as the 16th century, with records of civic election and papal selection betting in Italy, and to 18th-century England, where election betting was common in universities and coffeehouses [1]. One of the earliest modern electronic prediction markets was the Iowa Electronic Markets (IEM) [2], established as a research project by the University of Iowa in 1988. It allowed students and faculty to invest real money in contracts whose payoffs were tied to real-world events ranging from political elections to stock market performance.

1.1. Background and Motivation

Prediction markets are essentially information aggregation platforms where different actors, each with their own models, information, and beliefs, come together for price discovery of a future event. The market price of the event or an outcome reflects a collective estimate of the probability that the event will occur, making these markets useful for both trading and forecasting. They have been shown to outperform traditional polling methods [3, 4, 5], as they align incentives with correct information. Informed investors can profit from their knowledge while simultaneously contributing to a more accurate representation of the likelihood of different outcomes.

Unlike traditional stock markets, where traders rely on derivative contracts such as futures and options to speculate on asset prices, prediction markets provide a more straightforward way to trade on the likelihood of real-world outcomes [6]. This accessibility makes them appealing to both professional traders and casual participants who want to express a belief or hedge against a potential event.

In prediction markets, the price of a contract serves as an estimate of probability: the higher the price, the greater the perceived likelihood that the corresponding event will take place. An outcome token or contract represents a conditional claim, entitling the holder to payment if the outcome resolves to true. Similar to financial securities, the value of such a contract fluctuates over time as new information becomes available. Prediction market platforms facilitate this trading by providing counterparties for buy and sell orders, thereby enabling continuous price discovery. A key enabler of this process is liquidity provision. Liquidity providers (LPs) supply the market with capital by continuously posting buy and sell offers, ensuring that traders can always find a counterparty. Without them, markets would be illiquid, spreads would widen, and price discovery would become less efficient. By narrowing spreads and absorbing order flow, LPs make trading possible on a large scale.

1.2. Problem Statement

A central challenge for prediction market platforms lies in attracting sufficient liquidity to support active trading. In traditional financial markets, organised around centralised order books maintained by exchanges, liquidity providers are incentivised through mechanisms such as rebates and reduced transaction fees [7]. In decentralised finance, by contrast, market design can be based on automated market makers (AMM), central limit order books (CLOB), or hybrid structures, each of which creates unique incentives and risks for liquidity providers. This thesis examines Polymarket, one of the largest prediction markets, to assess how its transition from a fully decentralised design to a hybrid model in 2022 influenced the profitability of liquidity providers. Furthermore, it explores whether liquidity providers adopted specific strategies in response to these design features.

1.3. Research Questions

- **RQ1:** What factors determine the profitability of liquidity provision in AMMs on Polymarket prediction markets?

Liquidity provision in AMM pools requires depositing two complementary tokens of equal value. In return, liquidity providers (LPs) receive LP tokens that represent their proportional share of the pool. LPs earn a portion of the trading fees generated by activity in the pool.

In prediction markets, liquidity is typically provided using USDC as collateral, which is split into equal-value complementary tokens. The LP tokens obtained by providers fluctuate in value as arbitrageurs trade against the pool when its prices deviate from those on external markets. As a result, the value of LP tokens is not constant over the market's lifetime. If an LP redeems tokens when the relative proportions of the pool have shifted compared to when liquidity was first added, a loss may occur.

This research question investigates whether the trading fees earned by LPs are sufficient to offset losses caused by price divergence. It further examines how factors such as the duration of liquidity provision, the trading volume of the market, and the amount of collateral supplied influence the overall profitability of liquidity providers.

- **RQ2:** How can liquidity providers optimise their strategies to improve profitability in prediction market AMMs?

Building on the profitability analysis from RQ1, this question explores whether liquidity providers actively deploy strategies to outperform their peers, or whether a more passive approach proves to be more effective. The focus is on identifying patterns of behaviour among profitable LPs and understanding which tactics contribute to better outcomes.

Key considerations include whether actively managing liquidity—such as frequently adjusting positions, entering or exiting at specific times, or withdrawing liquidity close to market resolution reduces losses and improves profitability compared to simply remaining passive. The analysis also examines whether AMM-specific strategies, such as concentrating liquidity

during high-volume periods or timing entry and exit to minimise exposure to adverse price movements, are already being employed by existing LPs. By comparing these approaches, this research question aims to determine which strategies, if any, offer a consistent advantage in prediction market AMMs.

- **RQ3:** What factors determine the profitability of liquidity provision in CLOB-based prediction markets?

In CLOB-based designs, similar to traditional financial markets, an order book facilitates trading by matching bids and asks of comparable size. Liquidity providers in this environment act as counterparties to trades, ensuring sufficient liquidity and depth in the book. Their profitability typically comes from capturing the bid–ask spread, defined as the difference between the highest bid and the lowest ask.

Some platforms, such as Polymarket, further incentivise liquidity provision by offering rewards to those who place orders close to the mid-point of the spread. This research question examines whether liquidity providers in CLOB-based prediction markets are able to generate consistent profits from spreads and platform rewards, and to what extent these rewards contribute to their overall returns. It also considers whether liquidity providers employ specific strategies to maximise profitability—for example, optimising order placement, adjusting positions in response to market activity, or competing for rewards near the mid-price.

2. Background

This section provides an overview of key terms relevant to the thesis. It introduces the Polygon blockchain, which serves as the settlement layer for Polymarket, Oracles, and the Gnosis Conditional Token Framework, which underpins its market design. It also presents an overview of decentralised finance (DeFi), including core components such as automated market makers (AMMs) and impermanent loss. Together, these elements provide the necessary background to understand the analysis in this thesis.

2.1. Polygon

Polygon[8], formerly known as the Matic Network, is a framework designed to address Ethereum's [9] scalability limitations. It operates as a layer-2 scaling solution by processing transactions on separate Ethereum-compatible blockchains and then anchoring the results back to Ethereum. This approach reduces network congestion, lowers transaction costs to a fraction of a cent, and significantly increases throughput compared to Ethereum's base layer.

The network was founded in 2017 and initially launched as the Matic Network and later rebranded as Polygon in 2021. Polygon supports multiple scaling technologies, including plasma chains, proof-of-stake (PoS) sidechains [10], and rollup-based solutions such as zk-rollups and optimistic rollups [11]. This flexibility allows developers to select the most suitable approach for their decentralised applications (DApps). As a result, Polygon has become one of the most widely adopted Ethereum scaling platforms, enabling faster and cheaper interactions with DApps while maintaining compatibility with Ethereum's ecosystem.

Polymarket[1] uses the Polygon blockchain as a settlement layer and for deploying its smart contracts. By leveraging Polygon's low fees and high throughput, Polymarket ensures efficient trading and settlement of prediction market positions.

2.2. Blockchain Oracles

Blockchains, by design, cannot directly access external information such as asset prices, weather data, or IoT sensor outputs. This isolation ensures security and determinism but also creates a limitation known as the *oracle problem* [12]. To unlock most real-world use cases, smart contracts must consume off-chain data in a reliable, trust-minimised way. Oracles serve as this bridge between blockchains and external systems.

A blockchain oracle is middleware that listens for smart contract requests, fetches off-chain data from APIs or other sources, formats it into an on-chain compatible form, validates it

[1] `https://polymarket.com/`

through cryptographic proofs or consensus, and delivers it to the blockchain. In some cases, oracles also relay blockchain outputs back to external systems. Because centralised oracles introduce a single point of failure, decentralised oracle networks (such as Chainlink[13]) have emerged as the standard approach for secure data delivery.

An important recent development is the rise of *optimistic oracles* [14]. These operate under the assumption that most submitted data is correct and only escalate to a dispute resolution layer if challenged. This allows faster and cheaper data feeds compared to constant multi-party consensus. UMA's Optimistic Oracle [15] is a leading example: it accepts asserted data, gives a predefined window for disputes, and relies on UMA's Data Verification Mechanism (DVM) to resolve disagreements if they arise.

2.3. Decentralised Finance

Decentralised finance, or DeFi, represents a shift away from traditional financial systems that rely on banks and other centralised institutions as intermediaries. At its core, DeFi uses blockchain technology and smart contracts to create financial applications that operate on permissionless networks, allowing users to trade, lend, borrow, and invest directly with each other without needing a middleman [16, 17].

The philosophy behind DeFi is to make financial services more accessible, transparent, and cost-effective [18]. Unlike traditional finance, where banks control access and set rules, DeFi operates on public blockchains where anyone with an internet connection can participate [19]. The key components that make DeFi work include automated market makers (AMMs) and decentralised exchanges (DEXs).

2.3.1. AMM

AMMs [20, 21] are algorithmic systems that facilitate trades between cryptocurrencies without relying on traditional order books or direct counterparties. Instead of matching buyers and sellers, AMMs use mathematical formulas to determine prices and execute trades against liquidity pools.

Liquidity providers (LPs) supply equal values of paired tokens to AMM pools, earning a share of fees proportional to their contribution [22]. While this broadens access to market making, LPs face the risk of impermanent loss [23], which occurs when the relative price of deposited tokens diverges compared to simply holding them. Trading fees can partially offset these losses, but profitability is not guaranteed.

Automated Market Makers (AMMs) were formalised by Angeris et al. [24]. A widely adopted AMM design is the Constant Product Market Maker (CPMM), as implemented in Uniswap[2], which maintains the invariant:

$$x \cdot y = k$$

Where:

[2] `https://app.uniswap.org/swap`

- x — the quantity of token A in the pool,
- y — the quantity of token B in the pool,
- k — the constant product preserved during trades.

This mechanism adjusts prices automatically in response to supply and demand [25]. For example, when a user purchases ETH from a USDC/ETH pool, they deposit USDC and receive ETH, with quantities calculated to preserve k. This reflects the economic principle that increased demand leads to higher prices [26].

AMMs provide an alternative to central limit order books (CLOBs), since prices are determined algorithmically through constant-function mechanisms rather than continuous quoting by market makers. This design has been widely adopted in decentralised exchanges such as Uniswap and Sushiswap[3]. Similarly, decentralised prediction markets often rely on AMMs deployed on-chain via smart contracts. For example, Polymarket relied on AMMs until 2022, while Augur continues to use them on Ethereum.

Challenges of AMM-based order matching: In AMMs, liquidity pools serve as the counterparty to trades. This design introduces certain drawbacks. Liquidity providers may experience impermanent loss, i.e., a temporary reduction in the value of their locked tokens relative to holding them outright. AMMs are also subject to slippage, where large trades move prices significantly, particularly in pools with low liquidity [20]. Execution speed is constrained by the underlying blockchain, meaning slow transaction finality can lead to delays in trade settlement. Additionally, AMMs are vulnerable to Miner/Maximal Extractable Value (MEV) [27], where validators or block producers reorder or insert transactions for profit.

Liquidity provision in imbalanced pools: Over time, as market participants buy and sell outcome tokens, the AMM may hold unequal amounts across outcomes. When a liquidity provider adds liquidity under such conditions, the AMM preserves the existing proportions of tokens in the pool. Rather than accepting all of the newly deposited tokens, the AMM refunds the excess back to the liquidity provider so that the relative balance of outcome tokens remains unchanged (see Fig.2.1).

2.3.2. Impermanent Loss

Impermanent loss [28], also referred to as divergence loss, is the temporary reduction in the value of collateral provided by a liquidity provider to an AMM pool. It occurs when the token prices in the pool lag behind those on centralised exchanges or, in the case of prediction markets, the perceived fair value of the outcome tokens. While this loss is considered temporary, it becomes a realised loss if the liquidity provider withdraws their funds when prices diverge from the levels at which the liquidity was supplied initially. In essence, the size

[3] `https://www.sushi.com/ethereum/swap`

Figure 2.1.: Number of shares received on providing liquidity in AMM.

of the loss corresponds to the difference between the value of the tokens withdrawn from the pool and the value the provider would have had if they had held the tokens outside the pool.

It can be expressed mathematically as a function of the price ratio between the two assets. This loss is measured relative to simply holding the two assets (HODLing), and is given by [29]:

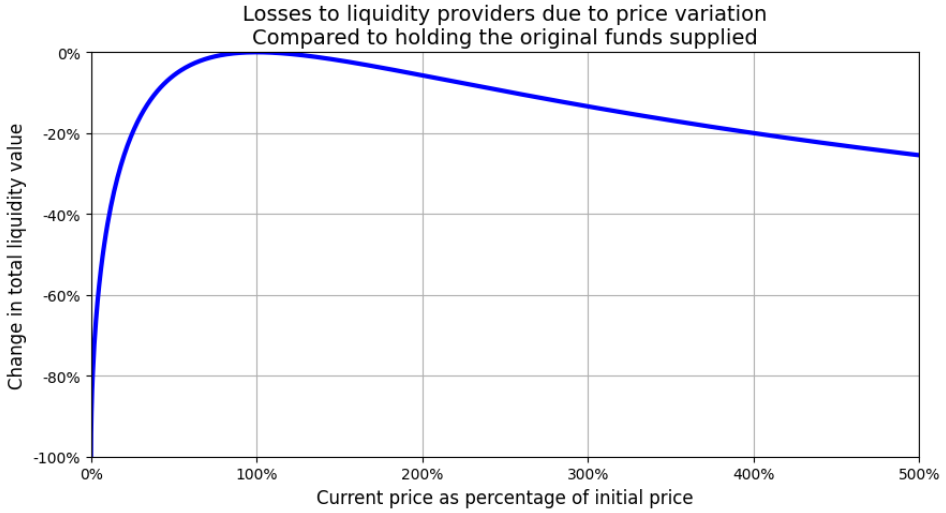

Figure 2.2.: Impermanent loss (divergence loss) experienced by liquidity providers compared to holding assets.

$$\text{Divergence Loss}(r) = \frac{2\sqrt{r}}{1+r} - 1 \tag{2.1}$$

Where:

- r is the price ratio between the new price and the initial price (e.g., $r = 2$ means the price has doubled),
- $\sqrt{r}$ arises from the geometric mean used in CPMMs,
- The entire expression compares the value of the LP position to the value if the user had held the assets.

To express this as a loss (i.e., a positive percentage), we can rewrite it as:

$$\text{Impermanent Loss}(r) = 1 - \frac{2\sqrt{r}}{1+r} \tag{2.2}$$

This form yields a percentage loss that increases as the price diverges more from the original.

2.4. Central Limit Order Book (CLOB)

In a CLOB, trades are executed by matching bids and asks stored in a central order book. The midpoint between the highest bid and the lowest ask often serves as a reference for the market price. To ensure smooth functioning, CLOBs rely on active market making to maintain sufficient liquidity. Market makers provide this liquidity by continuously posting orders on both sides of the book, while traders interact with these orders depending on their trading needs.

In practice, traders submit price–quantity orders to the book: buy (bid) or sell (ask). Makers post resting limit orders and supply liquidity, whereas takers consume liquidity by accepting outstanding quotes. When compatible orders cross, a trade is executed at the agreed price. Orders that are not immediately filled remain in the book until matched or cancelled. In some cases, only part of an order is filled, and the remainder stays in the book as a smaller order. This process creates a constantly updating view of supply and demand, which allows prices to adjust quickly as new information arrives.

CLOBs can be implemented either fully on-chain or with off-chain matching and on-chain settlement. Fully on-chain CLOBs provide maximum transparency but suffer from high latency and gas costs. Off-chain CLOBs, by contrast, offer lower latency and richer matching functionality, while settling the final trades on-chain for auditability and security. This separation of responsibilities enables higher performance without fully sacrificing transparency.

Because order matching is centralised, CLOBs introduce a potential point of centralisation: the operator could censor, delay, or prioritise specific trades. However, compared to AMMs, they are generally faster and more capital-efficient for market makers, as they avoid issues such as impermanent loss. In prediction markets, where participants frequently update beliefs and trading activity can spike around key events, this efficiency can be particularly important. To balance efficiency with transparency, some platforms such as Polymarket

adopt hybrid designs where order matching occurs off-chain via a CLOB, while settlement is performed on-chain. This hybrid model supports standard order types (e.g., market and limit), preserves a public record of executed trades, and maintains a clear separation between off-chain discovery/matching and on-chain transfer/settlement.

Challenges of CLOB-based prediction markets: CLOBs depend on continuous participation from market makers, which may discourage smaller liquidity providers. Thin order books can result in poor execution quality, wide bid–ask spreads, and high volatility in illiquid markets. Moreover, centralised matching engines can become bottlenecks or single points of failure, raising concerns about fairness and censorship resistance. Maintaining low-latency infrastructure is also costly and tends to benefit sophisticated traders who can invest in better technology. This dynamic may reduce accessibility for retail participants and shift profitability towards professional market makers.

2.5. Gnosis Conditional Token Framework (CTF)

The Gnosis Conditional Token Framework (CTF) [30] provides a framework for tokenising outcomes in prediction markets. Each outcome, represented by a `positionId`, is issued as a token conforming to the ERC-1155 standard [31], created by locking some collateral, typically USDC.

The framework enables users to *split* collateral into outcome tokens or *merge* outcome tokens back into the underlying collateral (see Fig. 2.3). This mechanism is extensively employed by platforms such as Polymarket. In the CPMM design, when collateral is supplied by a liquidity provider to the AMM contract, the CTF splits the collateral into an equal number of complementary outcome tokens, which are then added to the pool. When liquidity is withdrawn, these outcome tokens are merged to return the underlying collateral to the provider.

Beyond splitting and merging, the framework also supports two further core functionalities. First, *reporting payouts*, which is performed by the designated oracle. Once a market has resolved, the oracle submits the payout vector for the condition, specifying the share of collateral assigned to each outcome. Second, *redemption*, which allows holders of outcome tokens to exchange them for their share of collateral based on the reported payouts. Redemption can only take place once the condition has been resolved, ensuring that holders of winning outcome tokens receive their proportional claim.[4]

By standardising the creation, management, and settlement of outcome tokens, the CTF simplifies the development of decentralised prediction markets and ensures interoperability across platforms.

[4] `https://github.com/gnosis/conditional-tokens-contracts`

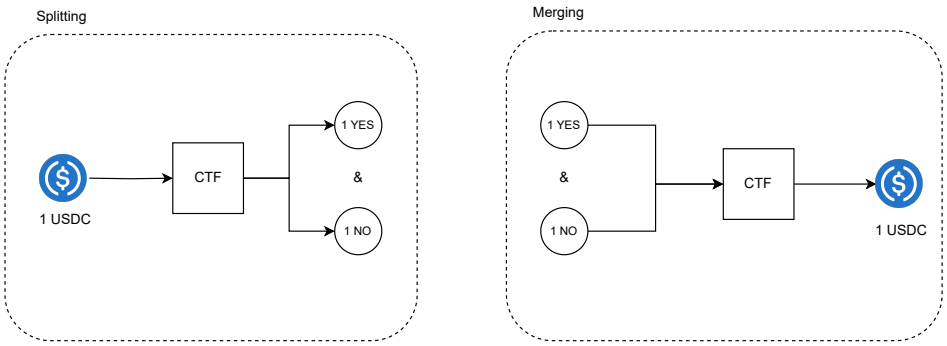

Figure 2.3.: Gnosis Conditional Token Framework.

2.5.1. ERC-1155 Token Standard

A key technical foundation of the Conditional Token Framework is the ERC-1155 token standard [31], which was introduced to address limitations of earlier Ethereum token standards, particularly ERC-20 [32] and ERC-721 [33]. The ERC-20 standard is designed for fungible tokens, where each token unit is identical and interchangeable with any other. Examples of such tokens include cryptocurrencies like USDC. In contrast, the ERC-721 standard is used for non-fungible tokens (NFTs), where each token is unique and can represent distinct assets such as digital art or collectables.

ERC-1155 builds on these ideas by providing a unified interface that can represent both fungible and non-fungible tokens within a single smart contract. This means that a single ERC-1155 contract can manage multiple types of tokens at the same time. For example, it can issue fungible tokens for in-game currency as well as non-fungible tokens for unique items or achievements, all under the same contract address. Each token type is identified by a unique ID, and the contract keeps track of balances for each address and token ID pair. This approach removes the need to deploy separate contracts for each new token type, as is required with ERC-20 and ERC-721, which helps reduce both deployment and transaction costs.

Another important feature of ERC-1155 is its support for batch operations. Functions such as `safeBatchTransferFrom` make it possible to execute multiple token transfers in a single transaction, even if the transfers involve different token types and recipients. This batching capability reduces the total amount of gas required and improves scalability. It is especially useful in applications where users may need to transfer many tokens at once, such as in gaming platforms or digital marketplaces.

To ensure safety and compatibility, ERC-1155 includes mechanisms that require receiving contracts to explicitly confirm their support for the standard. Specifically, the functions `onERC1155Received` and `onERC1155BatchReceived` must be implemented by contracts that are intended to receive ERC-1155 tokens. If a token is sent to a contract that does not implement these functions, the transfer will fail. This prevents tokens from being accidentally lost by being sent to contracts that cannot process them, which was a more common issue with earlier token standards.

ERC-1155 provides a flexible and cost-efficient token model that unifies fungible and non-fungible tokens under one interface. This reduces complexity for both developers and users, while also improving scalability and safety. Its adoption within the Conditional Token Framework enables efficient management of outcome tokens and supports the scalability required for decentralised prediction markets.

3. Prediction Market Structure

Prediction markets can be designed differently depending on how orders are matched and prices are determined. Common approaches include Automated Market Makers (AMMs) and Central Limit Order Books (CLOBs). Each design comes with distinct trade-offs, and many platforms experiment with hybrid models to balance efficiency, decentralisation, and user experience.

3.1. Prediction Market Mechanisms

The design of a prediction market directly influences liquidity, price discovery, and barriers to participation.

3.1.1. Price Discovery

How prices are determined in prediction markets depends on the underlying market mechanism. In AMMs, the price is set algorithmically and updates continuously based on the ratio of outcome tokens in the liquidity pool, moving as traders buy or sell. In contrast, in CLOBs, prices emerge from the interaction of buyers and sellers, with the prevailing market price typically reflected by the midpoint between the highest bid and the lowest ask in the order book.

3.1.2. Liquidity Provision

Liquidity providers, or market makers, are essential to ensure continuous trading. They take on market risk by maintaining a "two-way quote," i.e., simultaneously offering to buy and sell at competitive prices. In traditional financial markets, market makers are compensated by earning the bid–ask spread. In AMM-based prediction markets, liquidity providers are rewarded through trading fees, whereas in CLOB-based systems, they can earn both from the bid–ask spread and from rewards distributed by the platform for supplying liquidity.

In prediction markets, liquidity provision is particularly challenging because providers risk holding tokens that may become worthless once the market resolves. Adequate liquidity is therefore critical not only for smooth trading but also for effective information aggregation. Deep markets reduce transaction costs for arbitrageurs, encouraging their participation and allowing new information to be incorporated more efficiently, which in turn helps correct mispricing.

3.1.3. Market Resolution

A market is considered resolved once the final outcome becomes known. Resolution rules are usually specified before the market opens. Upon resolution, tokens corresponding to losing outcomes become worthless, while winning tokens settle at $1, and trading is halted.

Different platforms employ different resolution mechanisms. For example, Polymarket relies on UMA's decentralised oracle [34]: UMA token holders propose an outcome, which can be disputed if contested by the community [35]. In contrast, Kalshi[1] uses dedicated review teams that interpret market rules and declare the final outcome.

3.2. Centralised Prediction Markets

Historically, most prediction markets have been centralised, with a central operator responsible for matching bids and asks through mechanisms such as the Continuous Double Auction. Well-known examples include Bet365 [36], PredictIt [37], and Kalshi [38]. While effective in some respects, centralisation introduces several drawbacks:

1. **Single Point of Failure**: Centralised systems depend on a single entity or server. If the operator's infrastructure fails, the entire market may become inaccessible.
2. **Lack of Transparency**: The central operator controls both market data and decision-making processes. Participants, therefore, lack full visibility into how data is handled or how disputes are resolved.
3. **Higher Costs**: Operating a centralised market entails significant overhead, which is often passed on to users in the form of fees or commissions.
4. **Limited Access**: Access can be restricted based on geography or regulatory requirements set by the operator. For example, Kalshi is only accessible to participants within the U.S. jurisdiction.
5. **Market Manipulation**: Concentrated control creates the risk of manipulation, whether by insiders or the operator itself. Academic literature further highlights that participants with non-public insider information gain an advantage over others [39, 40]. This issue affects both centralised and decentralised prediction markets, but in centralised systems, such participants' holdings are not publicly visible, reducing accountability.

3.3. Decentralised Prediction Markets

Decentralised prediction markets are platforms that allow users to trade on the outcome of future events without relying on a central authority. By leveraging blockchain technology and smart contracts, these markets enable transparent, permissionless, and trust-minimised

[1] `https://kalshi.com/`

trading of event-based tokens. Participants can buy and sell shares representing different outcomes, with prices reflecting the collective beliefs of market participants about the likelihood of each event. This approach not only democratizes access to prediction markets but also provides a robust mechanism for aggregating information and forecasting real-world events.

3.3.1. Polymarket

Polymarket is the largest decentralised prediction market, handling over $9 billion in trading volume in 2024 [41]. Launched in 2020 and originally based on the Gnosis CTF framework (see Section 2.5), the platform gradually transitioned from an AMM-based model to a CLOB design by the last quarter of 2022.

Older AMM-based design

Until the last quarter of 2022, Polymarket employed a CPMM architecture implemented through the `FixedProductMarketMaker` smart contract. This design is based on the foundational CPMM principle where the product of token quantities remains constant: $x \cdot y = k$, where x and y represent the quantities of outcome tokens in the liquidity pool, and k is the invariant constant.

The `FixedProductMarketMaker`[2] contract integrates with the CTF to facilitate trading of prediction market outcomes. The contract maintains pools of conditional outcome tokens, where each outcome represents a distinct market scenario. Users could interact with the market through four primary operations: liquidity provision (`addFunding`), liquidity removal (`removeFunding`), token purchases (`buy`), and token sales (`sell`).

Price discovery occurs algorithmically through the CPMM formula. The `calcBuyAmount` and `calcSellAmount` functions implement the mathematical relationships that determine token quantities based on the current pool state.

The contract incorporates a fee mechanism (typically 2%) that is extracted from each trade and distributed proportionally to liquidity providers based on their share of the total liquidity pool. Liquidity providers receive ERC-20 LP tokens representing their proportional ownership of the pool, enabling them to claim accumulated fees and withdraw their share of the underlying collateral.

The design leverages the Conditional Tokens Framework's position splitting and merging capabilities to handle complex multi-outcome scenarios. When liquidity is added, the contract splits collateral tokens into conditional outcome tokens across all possible market outcomes. Conversely, when liquidity is removed or trades are executed, the contract merges conditional tokens back into collateral through the resolution process.

[2] `https://github.com/Polymarket/conditional-tokens-market-makers/blob/a48f865d702bcda35bb13aeb6c1840ea16453d24/contracts/FixedProductMarketMaker.sol`

Current CLOB-based design

Currently, on Polymarket, all trades are peer-to-peer. Users bet on event outcomes by purchasing *Yes* or *No* outcome tokens, whose prices reflect the market-implied probability of that outcome. For example, if a *Yes* token is priced at \$0.20, this implies a 20% chance that the event will resolve to *Yes*. The token price is determined by the balance of buy and sell orders. It is typically set at the midpoint between the highest bid and the lowest ask, or, if the bid–ask spread is too wide, by the last traded price. All markets on Polymarket are binary, meaning that every market can be expressed in terms of two mutually exclusive outcomes: *Yes* or *No*.

Fig.3.1 depicts the current design of the Polymarket order flow and resolution.

1. **Market setup.** Each binary market is instantiated in the Conditional Tokens Framework (CTF). The exchange (`CTFExchange`) is configured with the ERC20 collateral (USDC) and the ERC1155 conditional tokens contract; tokenIds for the two complementary outcomes are registered.

2. **User order creation.** A trader prepares either a market or limit order (buy or sell of an outcome token) as EIP-712 typed data. The order specifies maker/taker assets and amounts (USDC or outcome token), price, expiry, and a base fee rate. The trader signs the order off-chain.

3. **Off-chain matching.** The Polymarket's centralised operator collects signed orders, matches compatible orders off-chain. When a cross is found, the operator submits a transaction to the exchange, calling `matchOrders` with the matched orders and fill amounts.

4. **On-chain settlement (atomic).** The exchange transfers assets according to the match:
   - *Normal path (token ↔ collateral):* ERC1155 outcome tokens and USDC are exchanged between the two parties.
   - *Mint path (collateral → token set):* If both sides provide collateral for complementary outcomes, the exchange mints a set of complementary tokens (CTF split) and delivers the requested side to each counterparty.
   - *Merge path (token set → collateral):* If complementary tokens are supplied, the exchange merges them to collateral and pays out USDC (CTF merge).

5. **Resolution.** Upon event close, the oracle adapter posts the winning outcome to the CTF (e.g., via an optimistic-oracle mediated resolution flow with a dispute window). If no dispute succeeds, the condition is resolved on-chain.

6. **Redemption.** After resolution, winning outcome tokens are redeemable 1:1 for USDC (losing tokens redeem to \$0).

[3] `https://rocknblock.io/blog/polymarket-clob-design`

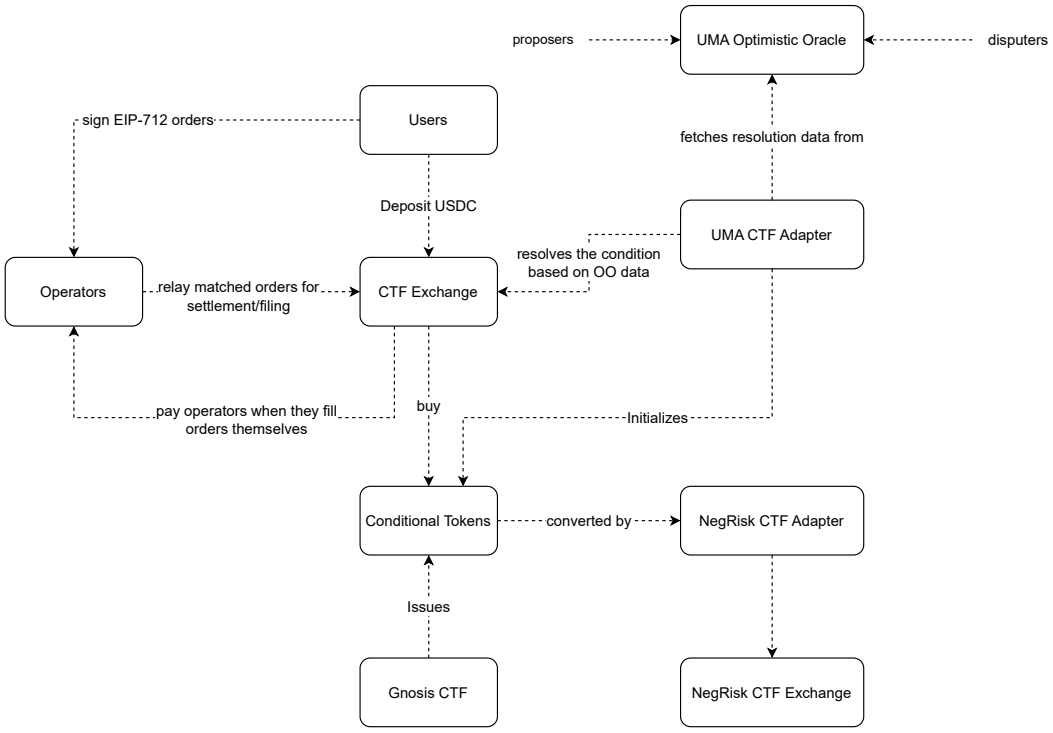

Figure 3.1.: Polymarket CLOB mechanism and design.[3]

Unified Order Books

In Polymarket's binary markets, a single underlying order book governs trading for both outcomes (e.g., *YES* and *NO*). Rather than maintaining separate books for each side, the platform uses a unified structure where the representation of orders depends on the outcome being viewed. This means that placing a bid for one outcome automatically appears as an ask on the complementary outcome's book, and vice versa.

This is achieved through a simple inversion: bids become asks, asks become bids, and order prices are adjusted by taking their complement with respect to 1 (i.e., $1 - p$) as depicted in fig.3.2. Importantly, this is purely a visual change; orders themselves are not duplicated or altered. This design allows for all trading directions (buy/sell for either outcome) to be fulfilled from a shared liquidity pool, improving efficiency and reducing fragmentation in binary markets.

Market Resolution

Prediction markets on Polymarket are resolved using UMA's Optimistic Oracle (OO)[4], which provides a permissionless mechanism to finalise market outcomes [15]. The OO operates as an escalation game: proposed resolutions are assumed to be correct unless challenged,

[4] `https://oracle.uma.xyz/`

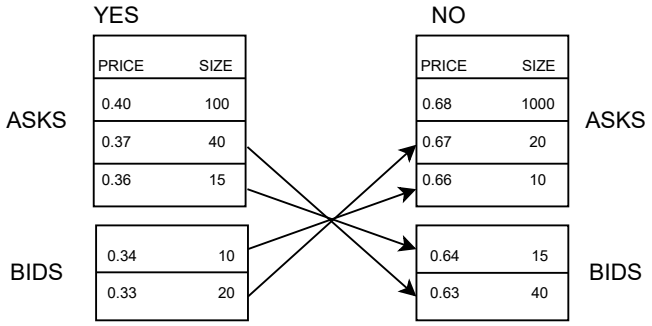

Figure 3.2.: Unified Order-book on Polymarket.

in which case the dispute is referred to UMA's Data Verification Mechanism (DVM), where tokenholders vote on the correct outcome. This structure allows markets to resolve quickly when there is agreement, while still providing a robust dispute process when disagreement arises.

To integrate with conditional tokens, Polymarket employs a custom contract called the `UmaCtfAdapter`[5]. This adapter connects the conditional token framework with the OO and manages the resolution process. When a market is created, the adapter initialises the resolution parameters by submitting ancillary data (the market question and clarifications), the reward amount for proposers, the bond size, and the liveness period during which proposals can be challenged.

The resolution process unfolds in several stages. Once the outcome of the underlying event is known, any participant may propose a resolution by submitting a price to the OO together with the required proposal bond. If the proposal is not challenged within the liveness period, it is accepted, and holders of winning outcome tokens can redeem them for $1 each. In this case, the proposer receives their bond back along with the reward for successful resolution.

If another participant disputes the proposed outcome, they must post an equal bond, and the dispute is escalated. Depending on the implementation, this can trigger either a second round of proposals or referral to the DVM for a vote among UMA tokenholders. The dispute process ensures that only well-supported resolutions succeed, while discouraging incorrect proposals through the risk of bond forfeiture.

The possible outcomes of the dispute process are straightforward. If the proposer is correct, they recover their bond and receive half of the challenger's bond as a bounty. If the challenger prevails, they keep their own bond and receive half of the proposer's bond. Proposals that are made prematurely, before the underlying event has concluded, are treated as invalid, with the challenger rewarded accordingly. In rare cases where the event cannot be clearly resolved, the market may settle at a predefined fallback, such as a 50/50 outcome.

[5] `https://github.com/Polymarket/uma-ctf-adapter`

Market Making Rewards

Polymarket operates a market-making rewards program designed to sustain liquidity across all active markets. The objective is to incentivise resting limit orders that provide two-sided depth close to the midpoint price of each market throughout its lifecycle [42]. By rewarding passive quoting, the system promotes tighter spreads, deeper order books, and balanced liquidity on both the YES and NO sides of each market.

Eligibility for rewards depends on a market maker's order placement. Only limit orders above a minimum size and within a specified distance from the midpoint are considered. The midpoint is calculated as the average of the best bid and best ask, and each market specifies a maximum spread threshold. Orders placed outside this range do not contribute to scoring. An order is eligible for rewards if

$$\text{midpoint} - \text{spread}_{\text{max}} < \text{price} < \text{midpoint} + \text{spread}_{\text{max}}.$$

Once filtered, eligible orders are scored according to three dimensions:

- **Tightness**: Orders closer to the midpoint earn higher scores. A quadratic scoring rule increases the reward for orders that narrow the spread.
- **Size**: Larger orders contribute proportionally more to a participant's score.
- **Two-sided quoting**: Participants providing liquidity on both sides of the book are rewarded more heavily than those quoting on only one side. Single-sided liquidity is down-weighted but not excluded.

Scores are sampled at regular intervals and aggregated over a reward epoch. A participant's final score in a market, Q_{epoch}, reflects their share of total qualified liquidity during that period. Rewards for that market are then allocated proportionally, according to

$$\text{Reward}_{i,m} = \frac{Q_{\text{epoch},i,m}}{\sum_j Q_{\text{epoch},j,m}} \times R_m,$$

where R_m is the total reward pool allocated to market m.
i indexes the individual liquidity provider, and m indexes the market under consideration.
Rewards are distributed directly to market makers' addresses in USDC.

4. Methodology

The first step involved fetching relevant on-chain data for markets deployed on the AMMs, followed by filtering the required events. The processed data was then analysed and used to generate plots.

For CLOB data, Polymarket's Gamma API[1] was used to obtain off-chain order data, since orders are matched off-chain but settled on-chain.

4.1. Data Gathering

To obtain the list of AMM markets, the Polymarket Subgraphs were utilised. A subgraph[43] is a custom API built on blockchain data that extracts, processes, and stores information in a manner that can be efficiently queried using GraphQL. All `Split` and `Merge` events, as well as all Fixed Product Market Maker (FPMM) addresses, were fetched from the Subgraph. The subgraph IDs used for this analysis were `Bx1W4S-n2DiBp`[2] and `81Dm16-f66nyC`[3]. Each FPMM address corresponds to a market, which is a contract deployed on the Polygon blockchain. From these contracts, token transfers and transaction details were retrieved using the Polygonscan API[4]. For each transaction, the senders, receivers, and funding-related events such as `FPMMFundingAdded`, `FPMMFundingRemoved`, `FPMMBuy`, and `FPMMSell` were extracted and stored locally in MongoDB. Additional details, such as block metadata, were queried separately to obtain timestamps and stored in MongoDB.

Listing 4.1: FPMM events

```
event FPMMFundingAdded(
    address indexed funder,
    uint[] amountsAdded,
    uint sharesMinted
);
event FPMMFundingRemoved(
    address indexed funder,
    uint[] amountsRemoved,
```

[1] https://docs.polymarket.com/developers/gamma-markets-api/overview

[2] https://thegraph.com/explorer/subgraphs/Bx1W4S7kDVxs9gC3s2G6DS8kdNBJNVhMviCtin2DiBp?view=Query&chain=arbitrum-one

[3] https://thegraph.com/explorer/subgraphs/81Dm16JjuFSrqz813HysXoUPvzTwE7fsfPk2RTf66nyC?view=Query&chain=arbitrum-one

[4] https://polygonscan.com/

```
        uint collateralRemovedFromFeePool,
        uint sharesBurnt
    );
    event FPMMBuy(
        address indexed buyer,
        uint investmentAmount,
        uint feeAmount,
        uint indexed outcomeIndex,
        uint outcomeTokensBought
    );
    event FPMMSell(
        address indexed seller,
        uint returnAmount,
        uint feeAmount,
        uint indexed outcomeIndex,
        uint outcomeTokensSold
    );
```

For the CLOB Polymarket data, the Gama API was utilised to obtain all available market details. The Subgraph IDs `EZCTgS-Mnjm9D`[5] were queried for all `OrderFilled` events in order to gather the executed orders for each market. However, the `OrderFilled` event does not include the actual makers of the filled order. To identify them, it was necessary to access the input to the `matchOrders` function of the `CTFExchange` contract, which contains the list of maker orders matched with a taker order. The corresponding `matchOrders` input for each filled order was queried using the PolygonScan API using the transaction IDs, and all maker orders were collected.

The scripts used for querying and storing data are available in the GitHub repository[6]. The FPMM[7](AMM markets) and `CTFExchange`[8](CLOB markets) can be accessed on GitHub as well.

Listing 4.2: CTFExchange OrderFilled event and matchOrders function

```
    event OrderFilled(
        bytes32 indexed orderHash,
        address indexed maker,
        address indexed taker,
        uint256 makerAssetId,
```

[5] https://thegraph.com/explorer/subgraphs/EZCTgSzLPuBSqQcuR3ifeiKHKBnpjHSNbYpty8Mnjm9D?view=Query&chain=arbitrum-one

[6] https://github.com/singhparshant/polymarketData

[7] https://github.com/Polymarket/conditional-tokens-market-makers/blob/a48f865d702bcda35bb13aeb6c1840ea16453d24/contracts/FixedProductMarketMaker.sol

[8] https://github.com/Polymarket/ctf-exchange/blob/00945f5c5a475e0bf5764b6d977f2a1ce4e8525d/src/exchange/CTFExchange.sol

```
    uint256 takerAssetId,
    uint256 makerAmountFilled,
    uint256 takerAmountFilled,
    uint256 fee
);

function matchOrders(
    Order memory takerOrder,
    Order[] memory makerOrders,
    uint256 takerFillAmount,
    uint256[] memory makerFillAmounts
)
```

4.2. FPMM Data Analysis

Based on the events shown in code snippets 4.1, the timeline of transactions on each FPMM market was reconstructed. Since the amounts being added to and removed from the pool are available, the state of the pool can be inferred, as the ratio of tokens added to and removed from the pool represents the current ratio of tokens present. The price of each token is calculated by taking the ratio of the quantities of the two tokens. When a liquidity provider transfers the collateral amount (in USDC) to the pool, equal amounts of the two complementary tokens are minted, and the quantities are rebalanced according to the ratio of the quantities in the pool. The higher-valued token, which is present in lower quantities in the pool, is returned to the liquidity provider(Fig.2.1). Therefore, the actual amount in USDC that becomes locked in the pool is not the full collateral amount, but rather the quantities of complementary tokens added to the pool multiplied by their respective prices.

4.2.1. Impermanent Loss Calculation

As discussed in Section 2.3.2, impermanent loss refers to the deviation in the value of a liquidity provider's collateral in an AMM pool compared to simply holding the tokens. From on-chain data, the `sharesMinted` values from funding-added events and `sharesBurnt` values from funding-removed events are observed. These represent the provider's share of the pool over time.

When liquidity is withdrawn, the realised impermanent loss is given by the difference between (i) the actual value of tokens removed from the pool, and (ii) the value of the same proportion of tokens had they simply been held.
The variables are defined as follows:

- q_i: Quantity of token i actually withdrawn from the pool.
- p_i: Current market price of token i.

- Q_i: Quantity of token i that the liquidity provider would hold if they had simply held their original deposit instead of providing liquidity.
- `shares_burnt`: Number of liquidity pool shares burned (i.e., removed) when liquidity is withdrawn.
- `total_shares_held`: Total liquidity pool shares held by the provider prior to withdrawal.
- `token_value`: Value of tokens actually withdrawn from the pool, computed as $\sum_i q_i \cdot p_i$.
- `hodl_total_value_now`: Current total value of the provider's original tokens if they had not been deposited in the pool, computed as $\sum_i Q_i \cdot p_i$.
- `proportion_removed`: Fraction of the provider's pool shares withdrawn, given by $\frac{\text{shares\_burnt}}{\text{shares\_held}}$.
- `hodl_value_removed`: Value of the proportion of the original tokens corresponding to the withdrawn shares.
- `Realised IL`: Realised impermanent loss, i.e., the difference between the actual withdrawn token value and the corresponding holding value.

$$\text{removed\_token\_value} = \sum_i q_i \cdot p_i$$

$$\text{hodl\_total\_value\_now} = \sum_i Q_i \cdot p_i$$

$$\text{proportion\_removed} = \frac{\text{shares\_burnt}}{\text{total\_shares\_held}}$$

$$\text{hodl\_value} = \text{proportion\_removed} \times \text{hodl\_total\_value\_now}$$

$$\text{Realised IL} = \text{hodl\_value} - \text{removed\_token\_value}$$

To calculate the total realised impermanent loss, this loss is summed across all liquidity removal events and across all markets.

4.2.2. P&L Calculation

Liquidity providers (LPs) earn trading fees by supplying liquidity to the AMM pool. The fee rate is determined by the market contract creator and typically ranges between 2% and 9%. However, LPs also face impermanent loss (IL), which occurs when token prices diverge from the original price at which the tokens were deposited into the pool.

To obtain a more realistic measure of profitability, both the fees earned and the realised impermanent loss are accounted for. The effective profit and loss (P&L) of a liquidity provider across all markets is given by:

$$\text{Effective P\&L} = \sum_{m \in \mathcal{M}} \text{Fees}_m - \sum_{m \in \mathcal{M}} \text{IL}_m$$

where $\mathcal{M}$ denotes the set of markets in which the LP has participated, Fees_m is the total trading fee earned in market m, and IL_m is the realised impermanent loss in market m calculated in 4.2.1.

4.3. CLOB Data Analysis

In CLOBs, unlike in AMM-based markets where any participant in a funding add or removal event is certainly a liquidity provider (since the primary goal is to earn trading fees rather than to bet on outcomes), the situation differs. In CLOBs, any user can place limit orders in the order book. This means it is not always possible to distinguish between a user who is genuinely providing liquidity and one who is simply speculating on future prices. To address this, users are considered liquidity providers only if they meet two criteria: (i) at least two buy and sell order matches in each order book in which they participate, and (ii) activity in at least five different markets. Another clarification: when referring to a "market" in the context of CLOBs, this denotes the two complementary order books ('Yes' and 'No' tokens) associated with the same condition ID.

Market makers in CLOB markets earn revenue in two main ways: (i) through spread capture and (ii) through rewards distributed by Polymarket. To capture revenue from bid-ask spreads, the market maker places limit orders on both sides of the order book. When passive orders are matched on both sides, revenue equal to the spread between bids and asks is earned.

However, market makers face certain risks when providing liquidity on order books:

- **Directional risk**: The risk of holding a position that loses value if the market moves strongly in one direction. For example, if the market price drifts away from the market maker's quoted range, an increasingly unprofitable position may be accumulated.
- **Inventory risk**: This arises when a market maker's buy and sell orders are not matched evenly, resulting in an imbalanced inventory of assets. If the price then moves against this imbalance, losses are incurred.

To manage these risks, market makers must constantly rebalance their inventory and adjust their bids and asks to avoid excessive exposure to large price swings or inventory imbalances.

4.3.1. Spread Capture P&L Calculations

Spread capture is calculated as the difference between the total value bought and sold by addresses that act as makers in matched orders. Since every taker order can be matched

against multiple maker orders (as seen in the input of the `matchOrders` method discussed in Section 4.2), all maker orders are iterated over and the net difference between their executed buys and sells is computed.

This difference, however, does not directly represent the maker's P&L, as it ignores the possibility that some tokens may have been acquired earlier as a taker, which would not appear in the above calculation. To obtain a more accurate measure of P&L, an inventory-neutral assumption is enforced: the maker should end the calculation with the same inventory level as at the start (i.e., zero net position).

To achieve this, an adjustment is made for any residual inventory at the end of the calculation:

- If the maker has sold more quantity than bought, it is assumed that the excess inventory was acquired earlier and is sold off at the last execution price until the inventory returns to zero.
- If the maker has bought more quantity than sold, the excess holdings are liquidated at the last execution price in order to neutralize the position.

By enforcing a flat inventory, profits attributable purely to spread capture are isolated, avoiding distortions from directional exposure. This approach enables a more accurate estimation of P&L for liquidity provision in CLOB markets.

4.3.2. Rewards Distribution

Rewards are distributed daily in USDC by the reward distribution contract `0xc288`. All token transfers from this contract between 2023-11-25 and 2025-07-24 were collected.

Unfortunately, Polymarket does not publish data on which rewards correspond to which specific markets. In other words, it is not possible to attribute rewards on a per-market basis. As a result, only the addresses that received the most rewards overall can be identified, but not the exact markets for which these rewards were earned.

5. Results

This chapter presents the empirical findings from the analysis of liquidity provision and market making in both AMM-based and CLOB-based prediction markets on Polymarket. The results are structured to address the research questions outlined previously, with a focus on profitability, strategy optimization, and the factors influencing outcomes for liquidity providers and market makers.

5.1. AMM

A total of 2,643 unique FPMM markets were observed over the period from October 2020 to November 2024. Approximately 3,000 unique liquidity providers participated in these markets, resulting in roughly 21,000 unique market–LP pairs.

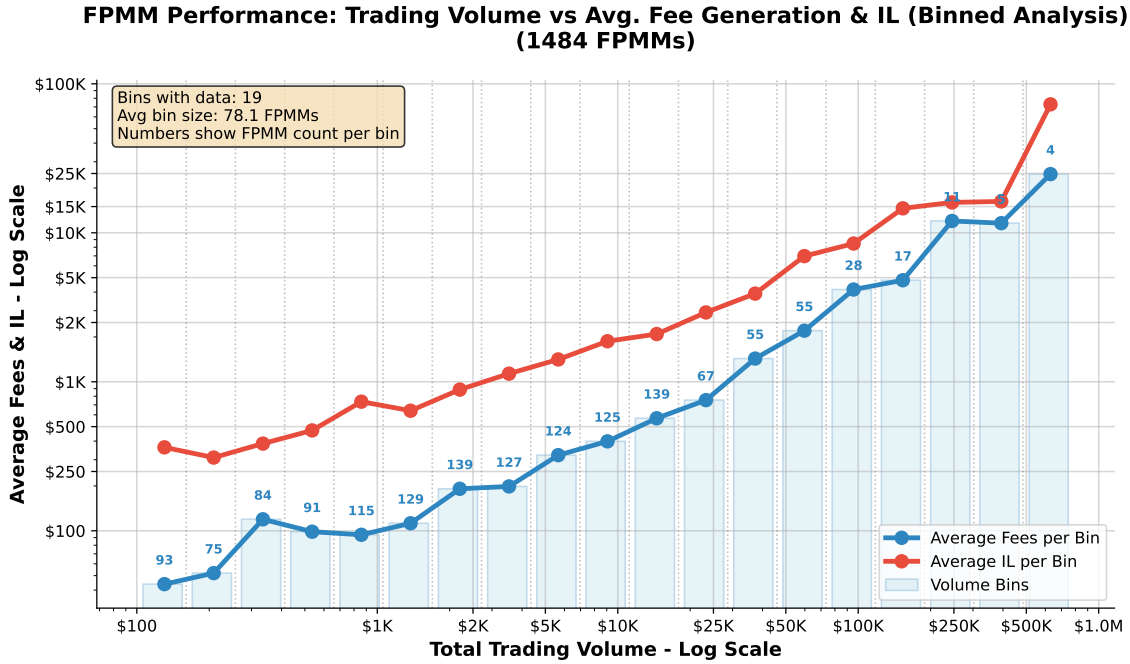

Figure 5.1.: Average fee earned, realised impermanent loss vs Total Trading volume of FPMM (Binned analysis)

To understand the effect of trading volume on LP profitability, a line graph (Fig. 5.1) was plotted, showing the average fees earned and the average impermanent loss on the y-axis

against the total market volume on the x-axis. Markets with similar volumes were grouped into bins, with the numbers on the bars indicating the number of markets in each bin. The results indicate that the average fees earned by LPs increase almost linearly with market volume, with the highest-volume markets generating nearly $25K USDC in fees on average. However, the average impermanent loss also rises with volume, and across all bins, the realised impermanent loss exceeds the fees earned. This suggests that, while higher-volume markets generate greater fee revenue through increased trading activity, the fees are not sufficient to offset the realised impermanent loss incurred by LPs.

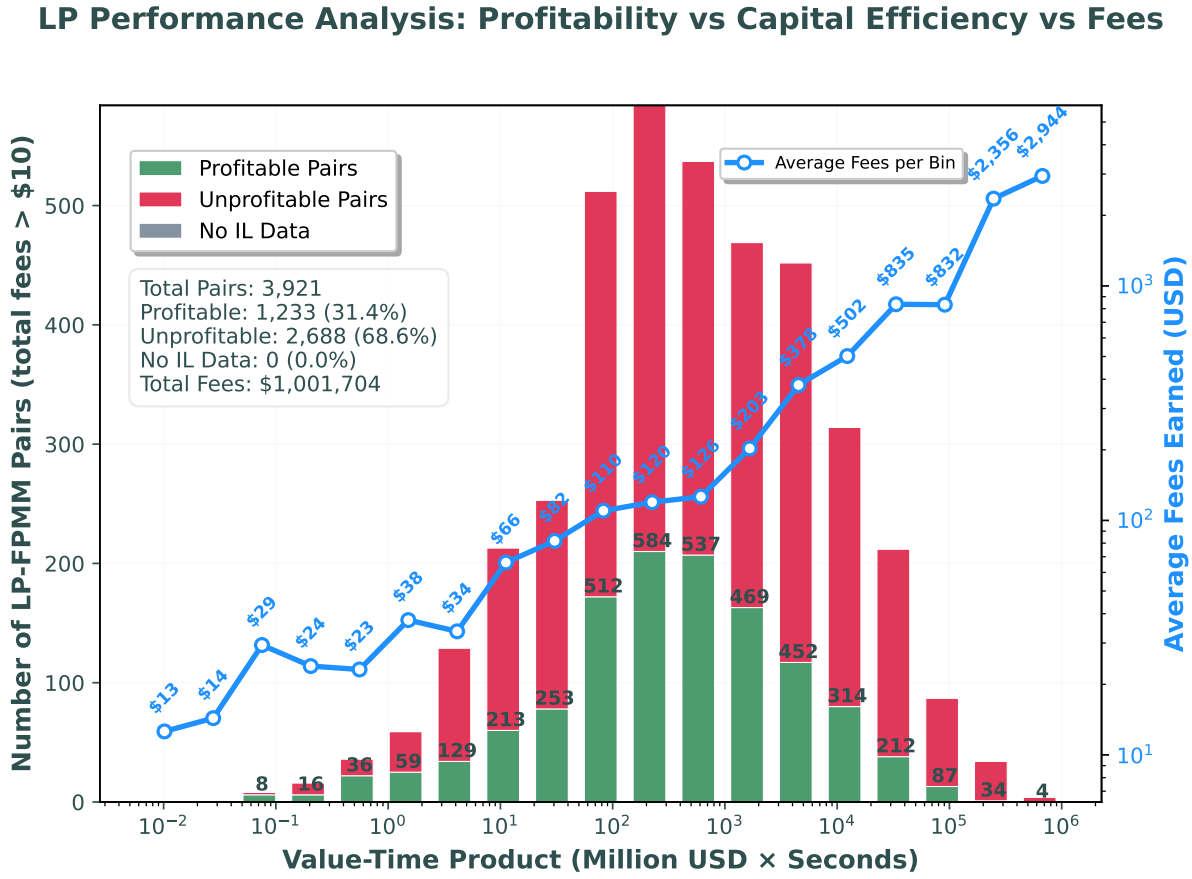

Figure 5.2.: Capital efficiency

Having established that higher-volume markets cannot offset the losses for LPs, the relationship between locking larger amounts of collateral for longer durations and LP profitability was examined. To this end, a dual-axis graph (Fig. 5.2) was plotted with capital efficiency on the x-axis, the number of LP–market pairs on the left y-axis, and average fees on the right y-axis. Capital efficiency is defined as the product of the collateral value supplied by an LP and the time (in seconds) it remained locked in the pool. For example, an LP providing a large amount of collateral for a short period may have a lower capital efficiency than one providing a smaller amount over a longer duration. This measure allows for the evaluation of

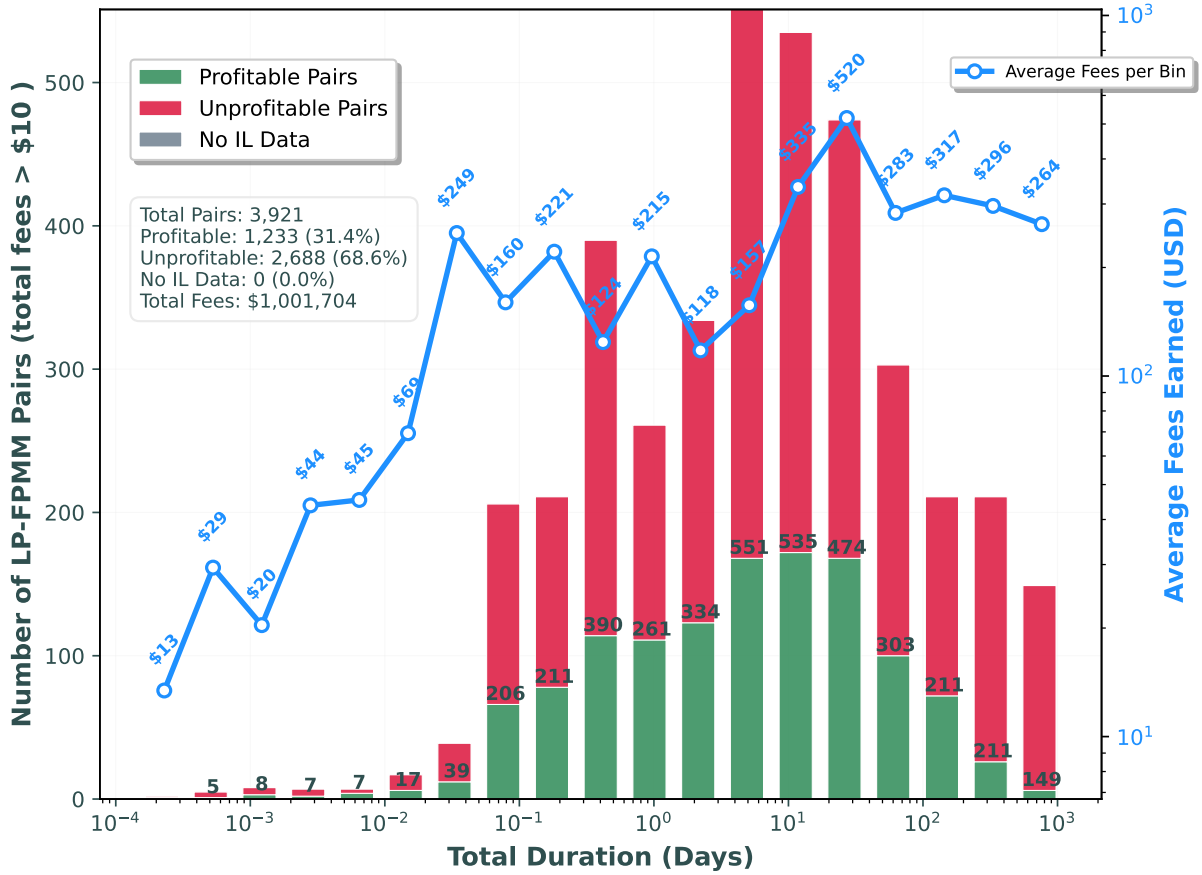

Figure 5.3.: Fee earned vs the duration collateral is locked in the pool

whether locking funds for longer periods can help offset impermanent losses.

The plot (Fig. 5.2) illustrates this relationship: the blue line shows the average fees per capital efficiency bin, while the bars represent the number of LP–market pairs in each bin, with green portions indicating profitable pairs and red portions indicating unprofitable ones. The results show that higher capital efficiency does not lead to improved LP profitability, as the proportion of profitable LP–market pairs remains roughly constant across all levels of capital efficiency. Although the average fees earned increase with capital efficiency, there is no corresponding rise in the share of profitable pairs. This finding reinforces the earlier observation from the fees-versus-volume analysis; even as fees increase, they are not sufficient to offset the losses incurred by LPs.

Fees were also plotted against the duration for which collateral remained locked in the pool (Fig. 5.3). The results show that longer lock-up periods do not improve the profitability of LPs. In fact, the share of profitable LPs remains roughly constant or even declines as duration increases. This indicates that extending the lock-up period neither enhances profitability nor guarantees higher fee income. Instead, fee earnings depend primarily on trading volume, as well as the interaction between the amount of collateral locked and the duration, but not on

duration alone.

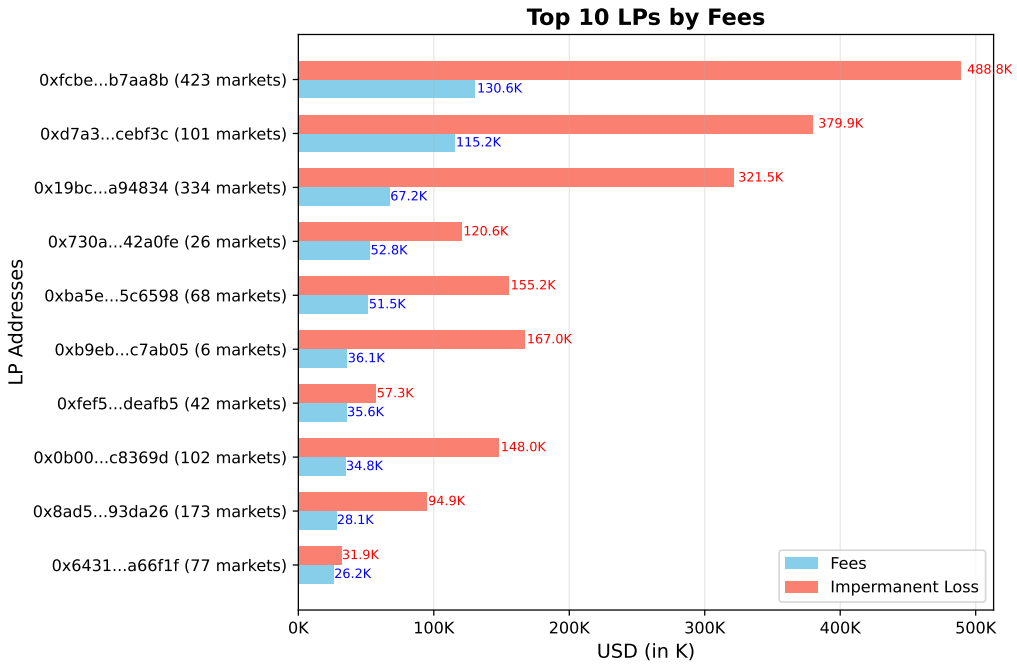

Figure 5.4.: Top 10 highest fee-earning LPs with total fees and realised impermanent losses across all markets

In Fig. 5.4, the highest fee-earning LPs and the corresponding realised impermanent loss across all the markets in which they acted as makers are displayed. Although the address `0xfcbe2` earned more than $130K USDC in fees, this amount is dwarfed by the realised impermanent loss suffered across all markets (>$400K USDC).

In Fig. 5.5, the most profitable LPs across all markets (by numbers) are shown. The methodology to calculate the P&L was discussed in Section 4. The most profitable LP `0x39ee` earned $8.6K just from a single market. This market was "*Will EIP-1559 be implemented on the Ethereum mainnet by August 5, 2021*?"[1].

As shown in the pie chart in Fig. 5.6, only 26% of individual LPs and about 24% of LP–market pairs are profitable, highlighting the overall low profitability of liquidity provision in AMM-based prediction markets. A large share of neutral outcomes among both groups corresponds to addresses that engaged in Just-in-Time (JiT) liquidity provision but accrued either zero or negligible fees.

When neutral LPs are excluded and only more involved providers earning at least $10 in fees are considered (Fig. 5.7), the share of profitable addresses increases to 37% among LPs and 31% among LP–market pairs. Nevertheless, these figures still indicate that liquidity provision in AMM-based prediction markets is largely unprofitable.

[1] `https://polygonscan.com/address/0x24a8b5b25eb8b2512a94982ede3319ca635936a6`

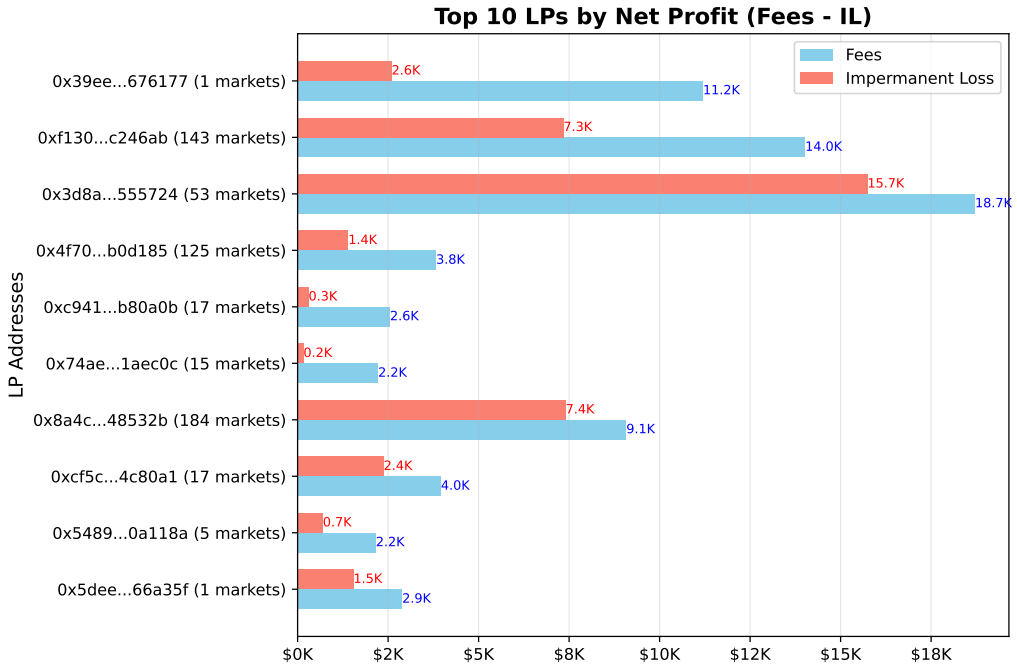

Figure 5.5.: Top 10 most profitable LPs across all markets.

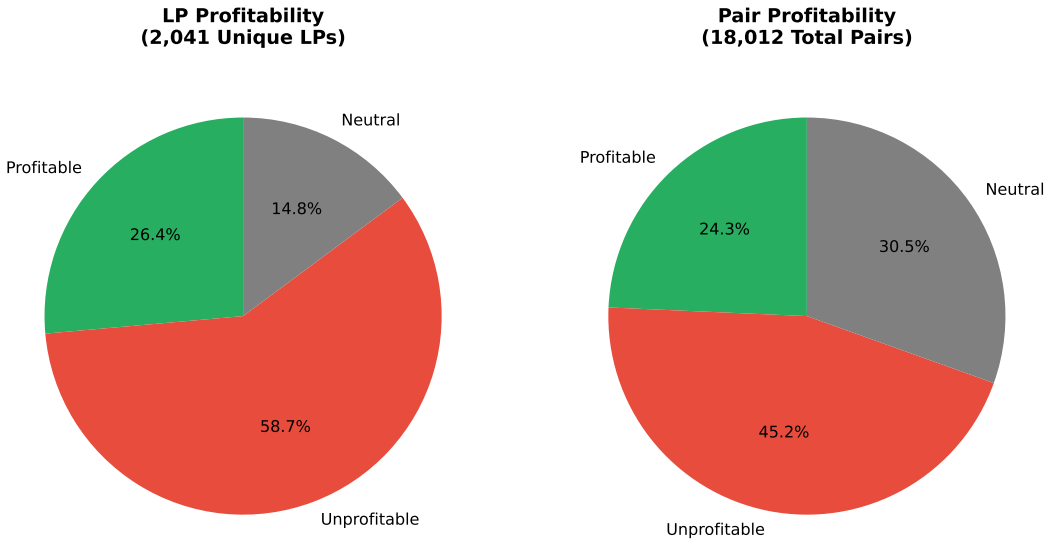

Figure 5.6.: Profitability percentage of all individual LPs and LP–market pairs

LP strategies used in AMM-based markets

In Fig. 5.8, a recurring pattern is observed in which an LP repeatedly adds and removes liquidity within the lifetime of a market. The circles, representing funding removed events,

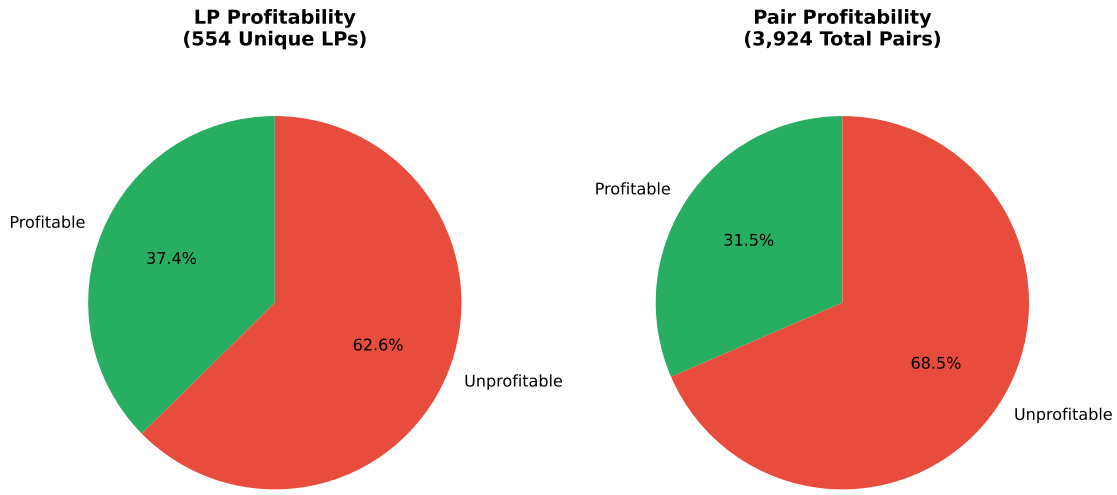

Figure 5.7.: Profitability percentage of individual LPs and pairs of LP and market earning at least $10 fee.

and the '+' signs, representing funding added events, often coincide. This behaviour is common across several LP–market pairs, where liquidity is added and then withdrawn within a very short period, often within just one to two minutes. Such behaviour is characteristic of a mechanism known as **Just-In-Time (JIT) liquidity** [44], also referred to as an LP Sandwich.

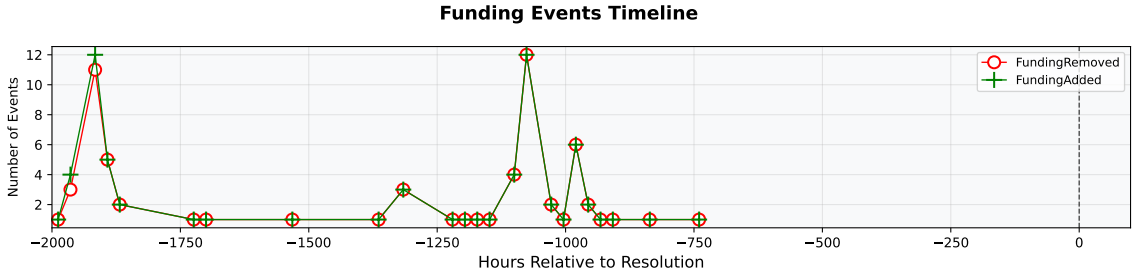

Figure 5.8.: Funding events timeline

In an AMM, trading fees are distributed proportionally to LPs based on their share of the pool. If a user can anticipate or detect an incoming trade to the market, liquidity can be strategically added right before the trade and withdrawn immediately after, effectively "sandwiching" the trade. While this is not harmful to the trader whose transaction is sandwiched—in fact, they may even benefit from reduced slippage due to the temporary increase in liquidity—it negatively impacts passive LPs. The reason is that the fees generated from the trade are now shared among all LPs, including the opportunistic JIT provider. By quickly removing liquidity, the attacker avoids exposure to impermanent loss while still capturing a share of the fees. In practice, such an attacker can monitor the mempool to detect

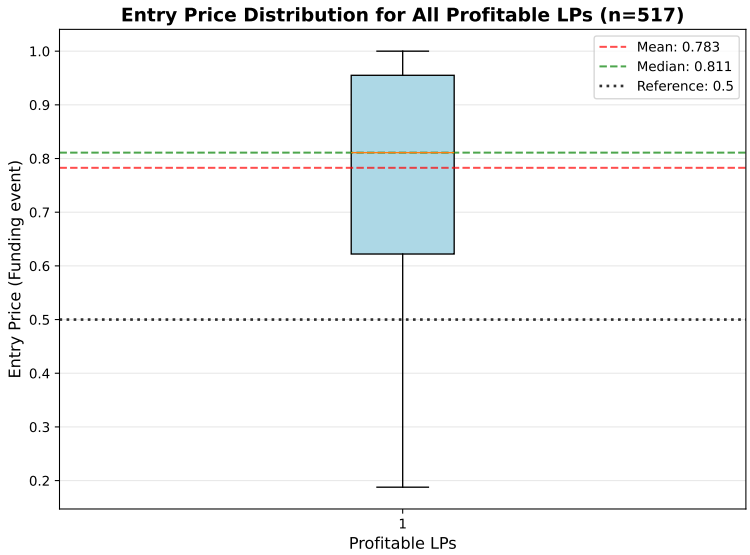

Figure 5.9.: Entry Price Distribution for All Profitable LPs

incoming trades and exploit this mechanism.

Approximately 30,000 transactions were observed in which an LP added liquidity and subsequently removed it entirely (i.e., withdrew all shares) within a span of five minutes, capturing a total of roughly $15,000 in fees. While this figure should be interpreted as a lower bound, since the analysis does not capture every possible JIT liquidity event, it nevertheless provides strong evidence that such practices were fairly common in AMM-based markets.

Another notable pattern among profitable LPs concerns the state of the market at the time of entry. As shown in Fig. 5.9, most profitable LPs added liquidity when the market was already leaning strongly toward resolution, with prices hovering around 75% for one outcome. This behaviour is consistent with the mechanics of impermanent loss: the potential loss is maximised when liquidity is provided near 50%, since one of the outcome tokens is guaranteed to become worthless at resolution. By contrast, entering the pool later, when prices are skewed toward one outcome, reduces the exposure to this loss.

In other words, by timing entry to coincide with a market that was already partially resolved, these LPs could limit downside from impermanent loss and rely on fee accrual to offset the smaller losses incurred.

In Fig. 5.10, the aggregate LP shares (pool tokens) held by all profitable LPs over the life of each market are plotted. The x-axis tracks market time from creation to resolution. At each FundingAdded event, newly minted shares are added; at each FundingRemoved event, shares burnt are subtracted. The series shows a pronounced decline in outstanding LP shares as resolution approaches, collapsing to (approximately) zero at resolution. This is consistent with risk management by LPs, where AMM LP shares represent a bundle of outcome tokens, one of which becomes worthless at resolution. Remaining in the pool through resolution concentrates risk and can crystallise large losses. Profitable LPs therefore exit shortly before

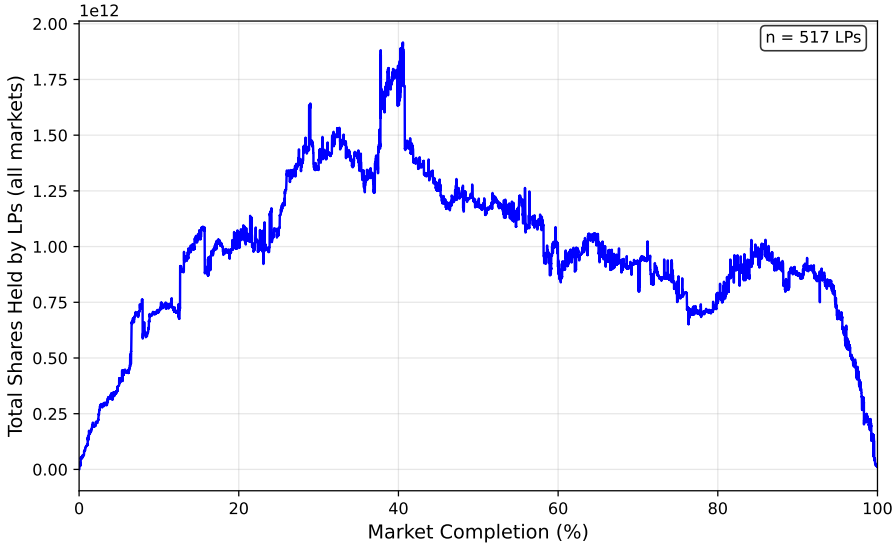

Figure 5.10.: Total Share Holdings Across All Profitable LPs vs market completion

resolution and redeem their position back to collateral, avoiding the binary jump after having earned fees earlier in the market.

5.2. CLOB

A total of 3,100 unique markets were analysed, along with more than 1.2 million matched order transactions. Nearly 800 unique liquidity providers were identified and analysed. Fig. 5.11 shows the cumulative and weekly rewards distributed by the rewards contract over the last 1.5 years. In total, about $9.3 million in rewards were distributed, with the highest weekly distribution reaching $242.4K in the final quarter of 2024. This can be attributed to the high trading volumes during this period, particularly due to the US presidential election.

As shown in Fig. 5.12, the top five addresses account for about 22% of all rewards. The single highest-earning address (`0x9d..1344`) alone received over $1 million in rewards. However, the remaining 78% of rewards were distributed across many other addresses (46,653 in total), suggesting that rewards were not overly concentrated among a few large players.

Fig. 5.13 and Fig. 5.14 illustrate how market makers earn revenue through spreads. For example, the maker (`0xa1..36d9`) places limit orders for the tokens (`6241..6338`) and (`2939..8334`), which are then executed. The basic strategy is to buy low and sell high, adjusting bids and asks as token prices move.

The spread depends on both competition in the market and the reward rules. In order to be eligible to earn rewards, spreads must remain within a certain range specified for each market. Since Polymarket does not make the order book data (bids and asks) publicly available, only execution prices are observed. To approximate spreads, opposing trades by the same maker within a one-hour window are paired, and the average across all such pairs is calculated.

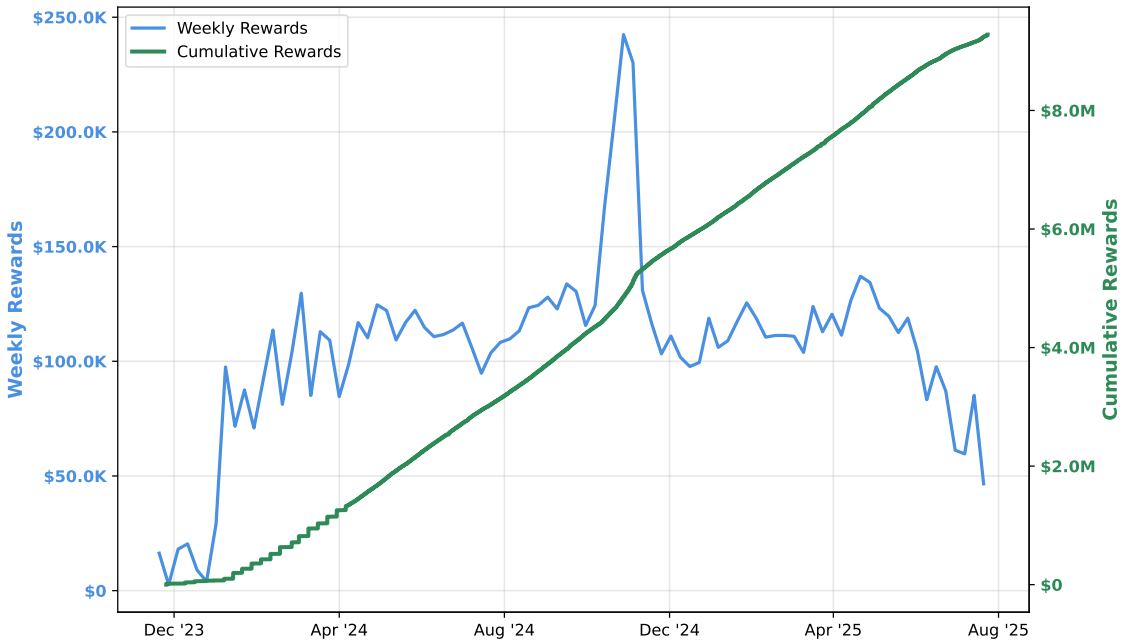

Figure 5.11.: Weekly and cumulative rewards over time.

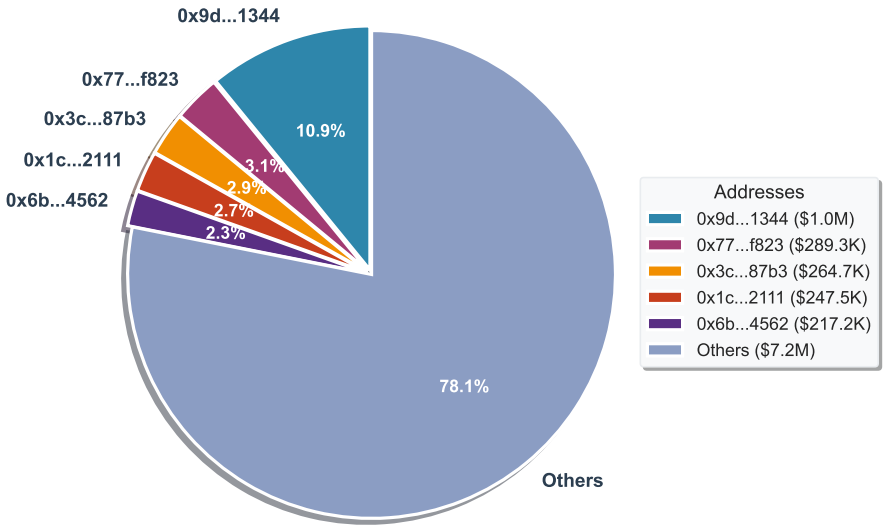

Figure 5.12.: Top 5 reward-earning addresses relative to all others.

From roughly 59,000 observations, an average spread of $0.023 (2.3 cents) is found, which falls within the typical reward-eligible range of 3 cents.

The P&L of makers was calculated following the methodology outlined in Section 4.3.1. The results show that the share of profitable individual makers and maker–market pairs is substantially higher than in AMM-based markets, at 69.7% and 76.1%, respectively.

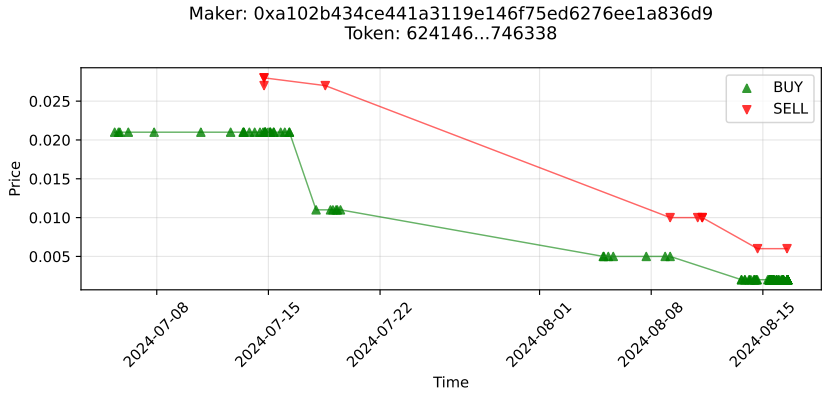

Figure 5.13.: Spread 1

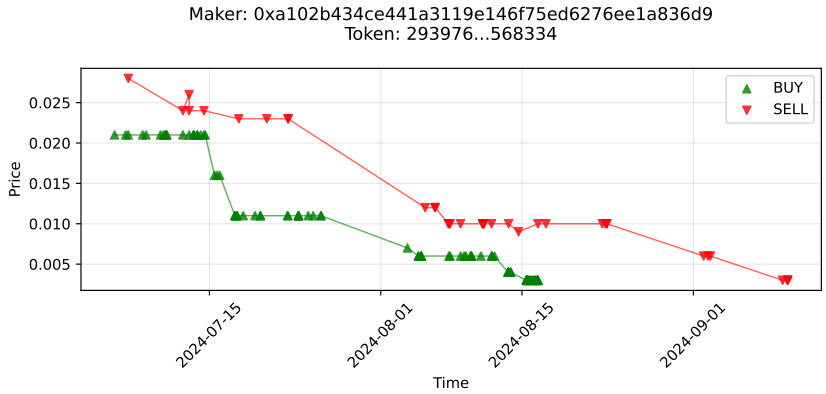

Figure 5.14.: Spread 2

It is important to note that these profitability figures are based solely on spread capture and do not include rewards distributed by Polymarket, as per-market reward data is not publicly available. Fig. 5.15 illustrates the profits earned through spreads by the top 20 makers.

With the results from Section 5, the research questions introduced in Section 1.3 can now be addressed.

5.3. Research Questions

Based on the results from Sections 5.1 and 5.2, the research questions detailed in Section 1.3 are addressed as follows.

5.3.1. RQ1: Profitability Factors in AMMs

Research Question: What factors determine the profitability of liquidity provision in AMMs on Polymarket prediction markets?

The most decisive factor for AMM profitability is the realised impermanent loss. The results

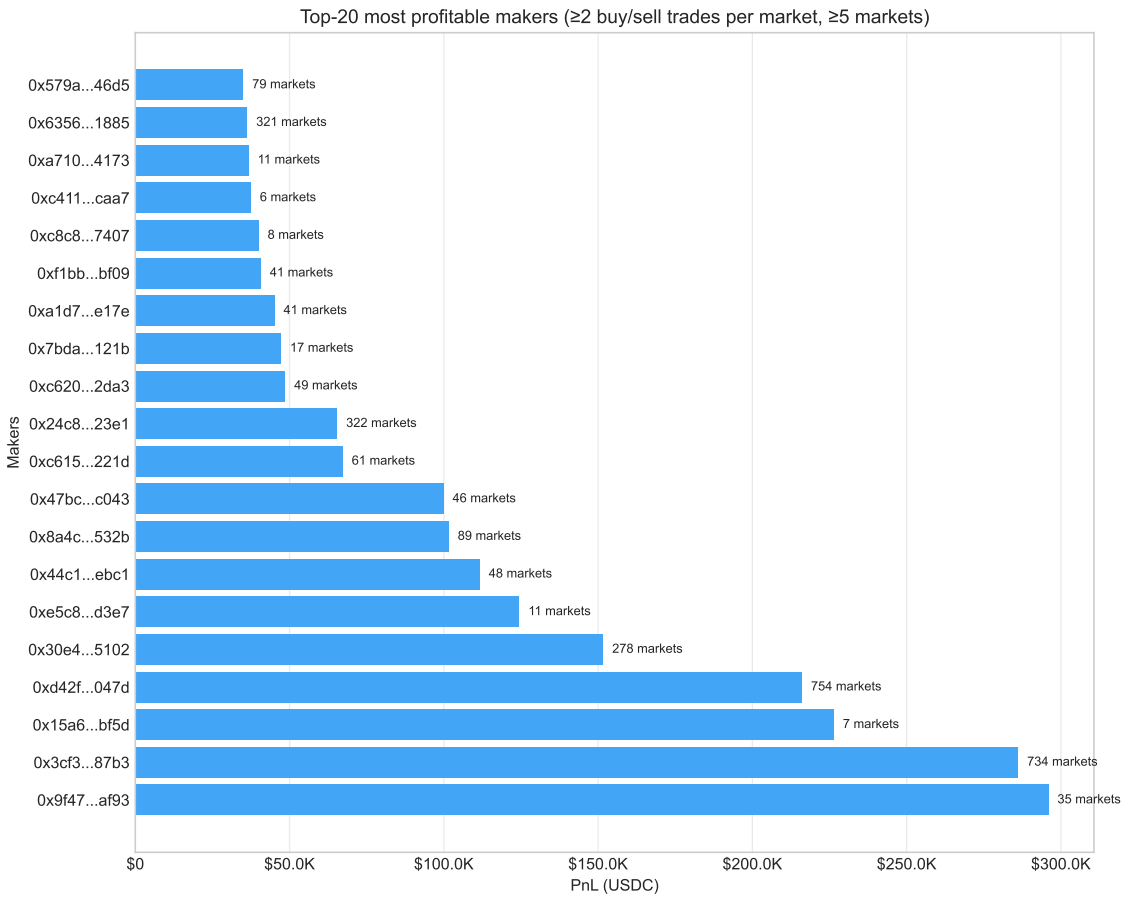

Figure 5.15.: Top-20 most profitable market makers and the number of markets they were involved in

indicate that most liquidity providers suffer impermanent losses that significantly exceed the trading fees earned. This effect is particularly pronounced in prediction markets, where extreme price divergence amplifies impermanent loss.

Passive LPs are therefore at a disadvantage: locking capital in pools for long durations does not offset impermanent loss, even when higher cumulative fees are earned. In fact, the highest fee-earning LPs often realise the largest net losses, as their higher capital locking magnifies exposure to impermanent loss.

5.3.2. RQ2: Strategy Optimization in AMMs

Research Question: How can liquidity providers optimise their strategies to improve profitability in prediction market AMMs?

One way LPs attempt to mitigate impermanent loss is by adding liquidity when the market is already leaning toward resolution. Entering at such a stage reduces the potential price divergence compared to providing liquidity at a balanced 50% state, where one of the tokens is

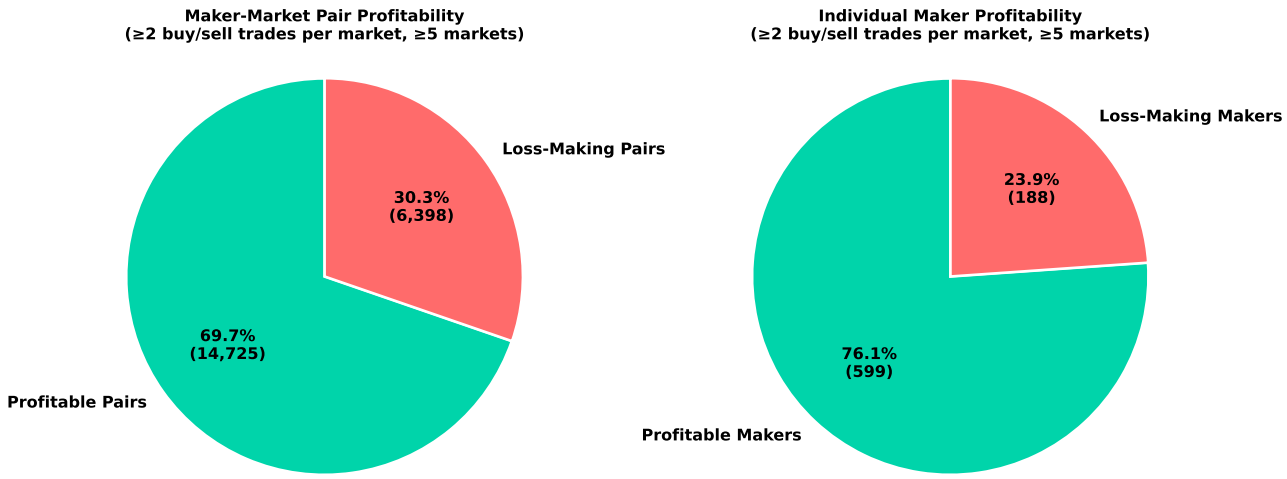

Figure 5.16.: Makers profitability comparison

certain to become worthless at settlement. Another effective strategy identified is *Just-In-Time (JIT) liquidity*, as discussed in Section 5.1. By providing liquidity only during periods of high trading activity and withdrawing it quickly thereafter, LPs can capture fees while minimising exposure to impermanent loss. Shorter provisioning windows, timed to coincide with bursts of trading volume, prove to be far more profitable than long-term passive liquidity provision.

5.3.3. RQ3: Profitability in CLOB Market Making

Research Question: What determines the profitability of market-making strategies in Central Limit Order Books for prediction markets?

In CLOBs, market makers earn primarily through spread capture and through the liquidity rewards distributed by Polymarket. Successful makers continuously adjust their bids and asks to avoid excessive inventory accumulation and to maintain a balanced flow of buys and sells, which maximises revenue from spreads.

From the reward formula, it is established that rewards increase with the square of the distance from the mid-price, while makers placing orders on both sides of the book earn proportionally more than those quoting on a single side. Thus, active order management and balanced quoting are key drivers of profitability.

6. Discussion

This thesis examines how the design of decentralised prediction markets affects the profitability of liquidity providers (LPs). The analysis is based on historical data from Polymarket, the largest decentralised prediction market, covering its transition from an AMM-based design to a CLOB-based design. The focus was on comparing the outcomes for LPs under both models.

The two main designs implemented so far are Automated Market Makers (AMMs) and Central Limit Order Books (CLOBs). A key drawback of AMM-based designs is impermanent loss. The results show that, on average, the impermanent losses suffered by LPs consistently outweigh the fees they earn, across all market sizes and volumes. Simply providing liquidity for longer durations or in larger amounts does not reduce this issue. Impermanent loss grows when the prices of the tokens in the pool diverge from their initial ratio. This effect is particularly severe in prediction markets because one of the two tokens eventually becomes worthless when the market resolves. As a result, LPs are often left holding only the worthless token due to the mechanics of the AMM pool.

Some strategies were observed among LPs to limit or avoid impermanent loss. One such approach is "Just-in-Time" (JIT) liquidity provision, where LPs add and remove liquidity over short periods to capture fees from large trades. Another common strategy is withdrawing liquidity as the market approaches resolution. The most profitable LPs were typically those who exited their positions before market resolution, thereby avoiding being left with the worthless token.

In contrast, under the CLOB design, LPs earn through two channels: spreads and rewards distributed by Polymarket. LPs adjust their bid and ask quotes as market prices shift, with the aim of avoiding large imbalances in their token inventory. The observed average spread was around three cents, suggesting that LPs are primarily focused on both capturing rewards and earning from spreads.

Overall, LP profitability in the CLOB design is significantly higher than in the AMM design. In AMMs, trading fees alone were not sufficient to offset the impermanent losses incurred by LPs. By comparison, the CLOB model offers greater profitability, but it requires active participation. LPs must continuously update their bids and asks to remain competitive, whereas in AMMs liquidity is provided passively and prices are set algorithmically.

There are other AMM designs, such as Uniswap's concentrated liquidity model, which may offer better outcomes for LPs and could be explored further in the context of prediction markets. The precise algorithms used by LPs in the CLOB setting could not be identified due to the lack of detailed quote-level data for each LP. Access to such data would provide a clearer understanding of how LPs place resting orders and manage their strategies within the CLOB design.

Alternative AMM Pool Designs and Their Impact on LP Profitability

Polymarket's AMM design was based on a pool containing "Yes" and "No" tokens, whose prices diverge to extreme levels as the market approaches resolution. Specifically, as the outcome becomes certain, one token approaches a value of zero while the other approaches $1. This dynamic results in severe impermanent loss for LPs, as they are left holding only the token that becomes worthless at settlement.

An alternative design that could mitigate this issue involves using pools composed of collateral and a single complementary outcome token, such as USDC/Yes and USDC/No pools. In this structure, each pool contains the stable collateral (e.g., USDC) and only one outcome token. For example, in a USDC/Yes pool, the price of the "Yes" token would move between zero and one USDC as the market evolves, but the USDC side of the pool would remain stable. This reduces the extent of price divergence compared to the Yes/No pool, where one token's value collapses entirely.

To illustrate, consider an LP that provides liquidity to a USDC/Yes pool. If the market moves in favour of "Yes," the price of the "Yes" token increases, and the LP's share of USDC decreases while their share of "Yes" tokens increases. However, since USDC retains its value, the LP is not exposed to the risk of holding a completely worthless asset. The maximum impermanent loss is therefore less severe than in a Yes/No pool, where the LP could end up with only the losing token, which is worth nothing. This design could provide a more balanced risk profile for LPs and potentially improve their profitability.

Another promising direction for improving LP outcomes in AMMs is the adoption of dynamic fee mechanisms. Unlike traditional AMMs that charge a fixed fee on each trade, dynamic fee models adjust trading fees in real time based on market conditions such as price volatility or deviations from external reference prices. By increasing fees during periods of high volatility or when arbitrage opportunities are more likely, these models make arbitrage less profitable and help recapture more value for LPs. Protocols like Arrakis[1] and HOT AMM[2] have implemented such mechanisms, with fees rising as the risk of loss-versus-rebalancing (LVR)[45] increases.

Concentrated Liquidity AMMs

Another AMM model that could enhance LP profitability is the concentrated liquidity model, as implemented in Uniswap v3 [46]. In this design, LPs can specify a price range within which their liquidity is active, rather than providing liquidity across the entire price spectrum. This allows LPs to concentrate their capital in the price ranges where most trading occurs, thereby increasing their share of trading fees and reducing exposure to impermanent loss outside their chosen range.

For example, in a prediction market, an LP might choose to provide liquidity only in the price range where the market is most likely to trade, based on current information or their own beliefs. If the market price remains within this range, the LP earns a higher proportion of fees relative to the amount of capital deployed. If the price moves outside the specified

[1] `https://arrakis.finance/blog/the-amm-renaissance-how-mev-auctions-and-dynamic-fees-prevent-lvr`
[2] `https://arrakis.finance/blog/hot-the-mev-aware-amm-built-to-empower-lps-is-live`

range, the LP's liquidity is no longer active, which limits further exposure to impermanent loss. This targeted approach can be more capital-efficient and may result in better outcomes for LPs, especially in markets with volatile or skewed price movements.

7. Conclusion

The mechanisms and designs of prediction markets play a central role in shaping liquidity provision, participant incentives, and profitability. An analysis of Polymarket before and after its transition in the last quarter of 2022 from an AMM-based design to a CLOB-based design provides important insights into these dynamics.

The AMM-based design of Polymarket was found to be largely unprofitable for liquidity providers. In most cases, the trading fees earned were insufficient to offset impermanent loss, which became realised once liquidity was withdrawn. Certain strategies were employed by liquidity providers in an attempt to mitigate these losses, such as late entry into markets to reduce exposure, removing liquidity near resolution or the use of Just-in-Time (JIT) liquidity, where liquidity was temporarily added immediately before trades and removed afterwards to capture fees. While these strategies occasionally yielded positive outcomes, they did not fundamentally alter the overall unprofitability of AMM-based prediction markets.

In contrast, the transition to a CLOB-based design substantially improved outcomes for liquidity providers. Profitability increased as liquidity providers were able to consistently earn from both the bid–ask spread and platform rewards. Over the past 1.5 years, the top 5 addresses receiving Polymarket rewards accounted for approximately 22% of the total rewards distributed, with the highest-earning address capturing around 11%. The remaining 78% of rewards were distributed among other addresses, indicating that reward distribution has not yet become highly centralised. This suggests that smaller participants are still able to obtain rewards by providing liquidity to the order book. The dual source of income rendered the CLOB model significantly more favourable compared to the AMM model.

These findings underscore the critical importance of market design in determining the sustainability of liquidity provision and the efficiency of information aggregation in prediction markets.

8. Future Work

Several directions remain open for future research. One is to conduct a more detailed study of Just-in-Time (JIT) liquidity strategies on Polymarket, quantifying how much profit liquidity providers were able to generate through such practices. This would help establish the extent to which JIT liquidity was prevalent in Polymarket transactions.

For the CLOB-based design, our analysis was limited to executed trades, as we did not have access to the full order book, including bids and asks placed by market makers. Access to such data, along with detailed information on market-specific rewards distributed by Polymarket, would enable a clearer understanding of the algorithms and quoting strategies employed by sophisticated market makers when adjusting their bid–ask spreads.

Beyond Polymarket, future work could also explore and compare other prediction market designs, such as Augur or Limitless, examining how their mechanisms differ and how these differences affect the profitability and behaviour of liquidity providers.

A. Apendix

A.1. Top 20 Most Profitable Makers by spread capture (CLOB)

Table A.1.: Top 20 Most Profitable Makers by spread capture (CLOB)

Rank	Maker Address	PnL (USDC)	Markets	Total Trades	Buy Trades	Sell Trades
1	0x9f4..af93	295,900	35	15,859	15,179	680
2	0x3cf..87b3	286,100	734	44,890	30,181	14,709
3	0x15a..bf5d	226,400	7	617	429	188
4	0xd42..047d	216,000	754	39,684	27,823	11,861
5	0x30e..5102	151,600	278	8,982	4,833	4,149
6	0xe5c..d3e7	124,400	11	2,968	2,074	894
7	0x44c..ebc1	111,600	48	1,535	868	667
8	0x8a4c..532b	101,500	89	4,263	2,514	1,749
9	0x47bc..c043	99,900	46	1,460	1,017	443
10	0xc615..221d	67,200	61	1,609	871	738
11	0x24c8..e1	65,400	322	16,132	11,142	4,990
12	0xc620..da3	48,500	49	2,598	1,435	1,163
13	0x7bda..21b	47,200	17	2,557	2,320	237
14	0xa1d7..17e	45,100	41	7,361	3,969	3,392
15	0xf1bb..f09	40,600	41	1,009	561	448
16	0xc8c8..407	39,900	8	365	256	109
17	0xc411..aa7	37,400	6	312	292	20
18	0xa710..173	36,700	11	512	382	130
19	0x6356..885	36,200	321	15,597	8,776	6,821
20	0x579a..6d5	34,900	79	1,517	999	518

Note: Makers with $\geq$2 buy and $\geq$2 sell trades per market, $\geq$5 markets total.

A.2. Top 10 LPs by Net Profit (AMM)

Table A.2.: Top 10 LPs by Net Profit (AMM)

Rank	LP Address	Fees (USDC)	IL (USDC)	Net Profit (USDC)	Markets
1	0x39ee..177	11,200	-2,600	8,600	1
2	0xf130..6ab	14,000	-7,300	6,700	143
3	0x3d8a..724	18,700	-15,700	3,000	53
4	0x4f70..185	3,800	-1,400	2,400	125
5	0xc941..0b	2,600	-300	2,300	17
6	0x74ae..0c	2,200	-200	2,100	15
7	0x8a4c..32b	9,100	-7,400	1,700	184
8	0xcf5c..a1	4,000	-2,400	1,600	17
9	0x5489..18a	2,200	-700	1,500	5
10	0x5dee..35f	2,900	-1,500	1,300	1

Note: Net Profit = Fees - |IL|. IL values are negative indicating losses.

A.3. Top 10 LPs by Fees (AMM)

Table A.3.: Top 10 LPs by Fees (AMM)

Rank	LP Address	Fees (USDC)	IL (USDC)	Markets
1	0xfcbe..a8b	130,600	-488,800	423
2	0xd7a3..f3c	115,200	-379,900	101
3	0x19bc..834	67,200	-321,500	334
4	0x730a..0fe	52,800	-120,600	26
5	0xba5e..598	51,500	-155,200	68
6	0xb9eb..b05	36,100	-167,000	6
7	0xfef5..fb5	35,600	-57,300	42
8	0x0b00..69d	34,800	-148,000	102
9	0x8ad5..a26	28,100	-94,900	173
10	0x6431..f1f	26,200	-31,900	77

Note: Net Profit = Fees - |IL|. IL values are negative indicating losses.

A.4. Top Rewards-Earning Addresses

Table A.4.: Top Rewards-Earning Addresses (Dec '23–Aug '25)

Rank	Address	Rewards (USDC)	Share
1	0x9d84..1344	1,011,500	9.9%
2	0x7789..f823	289,343	2.8%
3	0x3cf3..87b3	264,727	2.6%
4	0x1cfc..2111	247,471	2.4%
5	0x6b7e..4562	217,238	2.1%
6	0x96b5..f5e7	208,028	2.0%
7	0xc8ab..6418	191,250	1.9%
8	0x4cc3..7552	183,953	1.8%
9	0xd42f..047d	171,122	1.7%
10	0x6356..1885	145,887	1.4%
	Others	7.2M	71.2%
	Total	9.1M	

List of Figures

2.1. Number of shares received on providing liquidity in AMM. 7
2.2. Impermanent loss (divergence loss) experienced by liquidity providers compared to holding assets. 7
2.3. Gnosis Conditional Token Framework. 10

3.1. Polymarket CLOB mechanism and design.[1] 16
3.2. Unified Order-book on Polymarket. 17

5.1. Average fee earned, realised impermanent loss vs Total Trading volume of FPMM (Binned analysis) 25
5.2. Capital efficiency 26
5.3. Fee earned vs the duration collateral is locked in the pool 27
5.4. Top 10 highest fee-earning LPs with total fees and realised impermanent losses across all markets 28
5.5. Top 10 most profitable LPs across all markets. 29
5.6. Profitability percentage of all individual LPs and LP–market pairs 29
5.7. Profitability percentage of individual LPs and pairs of LP and market earning at least $10 fee. 30
5.8. Funding events timeline 30
5.9. Entry Price Distribution for All Profitable LPs 31
5.10. Total Share Holdings Across All Profitable LPs vs market completion 32
5.11. Weekly and cumulative rewards over time. 33
5.12. Top 5 reward-earning addresses relative to all others. 33
5.13. Spread 1 34
5.14. Spread 2 34
5.15. Top-20 most profitable market makers and the number of markets they were involved in 35
5.16. Makers profitability comparison 36

List of Tables

A.1. Top 20 Most Profitable Makers by spread capture (CLOB) 42
A.2. Top 10 LPs by Net Profit (AMM) 43
A.3. Top 10 LPs by Fees (AMM) 43
A.4. Top Rewards-Earning Addresses (Dec ’23–Aug ’25) 44

Acronyms

AMM Automated Market Maker. vi, vii, 2, 5, 8, 12, 19, 22, 25

CLOB Central Limit Order Book. vii, 2, 6, 8, 9, 12, 14, 19, 20, 23, 32

CPMM Constant Product Market Maker. 5, 8, 9, 14

CTF Conditional Token Framework. 9, 14

FPMM Fixed Product Market Maker. 19–21, 25

Bibliography

[1] P. W. Rhode and K. Strumpf. "The long history of political betting markets: An international perspective". In: (2013).

[2] Iowa Electronic Markets. *What is the IEM?* https://iemweb.biz.uiowa.edu/about-iem/what-is-the-iem/. Accessed: 2025-07-03. n.d.

[3] J. Wolfers and E. Zitzewitz. "Prediction markets". In: *Journal of economic perspectives* 18.2 (2004), pp. 107–126.

[4] K. J. Arrow, R. Forsythe, M. Gorham, R. Hahn, R. Hanson, J. O. Ledyard, S. Levmore, R. Litan, P. Milgrom, F. D. Nelson, et al. *The promise of prediction markets*. 2008.

[5] D. B. AG. *Xetra Liquidity Provider Programme*. https://www.xetra.com/xetra-en/trading/trading-fees-and-charges/liquidity-provider-programme. Accessed: 2025-08-26. 2025.

[6] Sofia Cossar. *Mechanisms for Prediction Markets*. https://paragraph.com/@mechanism.institute/prediction-markets. Accessed: 2025-07-03.

[7] I. O. of Securities Commissions. *Liquidity Provision in the Secondary Markets for Equity Securities: Final Report*. Final Report FR07/2020. Accessed: 2025-09-03. International Organization of Securities Commissions, Aug. 2020.

[8] P. Labs. *Polygon Blog*. https://polygon.technology/blog. Accessed: 2025-09-03. 2025.

[9] V. Buterin et al. "Ethereum white paper". In: *GitHub repository* 1.22-23 (2013), pp. 5–7.

[10] F. Saleh. "Blockchain without waste: Proof-of-stake". In: *The Review of financial studies* 34.3 (2021), pp. 1156–1190.

[11] D. Čapko, S. Vukmirović, and N. Nedić. "State of the art of zero-knowledge proofs in blockchain". In: *2022 30th Telecommunications Forum (TELFOR)*. IEEE. 2022, pp. 1–4.

[12] C. Labs. *The Blockchain Oracle Problem*. https://chain.link/education-hub/oracle-problem. Last updated August 22, 2025; Accessed: 2025-09-03. 2025.

[13] L. Breidenbach, C. Cachin, B. Chan, A. Coventry, S. Ellis, A. Juels, F. Koushanfar, A. Miller, B. Magauran, D. Moroz, et al. "Chainlink 2.0: Next steps in the evolution of decentralized oracle networks". In: *Chainlink Labs* 1 (2021), pp. 1–136.

[14] U. Protocol. *How does UMA's Oracle work?* https://docs.uma.xyz/protocol-overview/how-does-umas-oracle-work. Accessed: 2025-09-03. 2025.

[15] *UMA Protocol: a Decentralized Optimistic Oracle*. https://uma.xyz/. Accessed: 2025-08-03. UMA, 2025.

[16] F. Schär. "Decentralized finance: On blockchain-and smart contract-based financial markets". In: *Federal Reserve Bank of St. Louis Review* 103.2 (2021), pp. 153–174.

[17] I. Makarov and A. Schoar. "Cryptocurrencies and decentralized finance (DeFi)". In: *BIS Working Paper No 1061* (2022).

[18] P. K. Ozili. "Decentralized finance research and developments around the world". In: *Journal of Banking and Financial Technology* 6.2 (2022), pp. 117–133.

[19] D. Yaga, P. Mell, N. Roby, and K. Scarfone. *Blockchain technology overview*. Tech. rep. 8202. National Institute of Standards and Technology, 2018.

[20] M. Pourpouneh, K. Nielsen, and O. Ross. *Automated market makers*. Tech. rep. 2020/08. University of Copenhagen, 2020.

[21] E. A. Meyer, I. M. Welpe, and P. Sandner. "Decentralized finance–a systematic literature review and research directions". In: *ECIS 2022 Research Papers*. Vol. 25. 2022.

[22] J. Choong. "Are automated market makers a sustainable exchange model?" PhD thesis. University of Technology Sydney, 2021.

[23] D. J. Im, A. Kondratskiy, V. Harvey, and H. W. Fu. "UAMM: Price-oracle based automated market maker". In: *arXiv preprint arXiv:2308.06375* (2024).

[24] G. Angeris and T. Chitra. "Improved price oracles: Constant function market makers". In: *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*. 2020, pp. 80–91.

[25] Y. Chen and D. M. Pennock. "A utility framework for bounded-loss market makers". In: *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*. 2007, pp. 49–56.

[26] M. J. Curry, Z. Fan, and D. C. Parkes. "Optimal automated market makers: Differentiable economics and strong duality". In: *arXiv preprint arXiv:2402.09129* (2024).

[27] S. Yang, F. Zhang, K. Huang, X. Chen, Y. Yang, and F. Zhu. "Sok: Mev countermeasures: Theory and practice". In: *arXiv preprint arXiv:2212.05111* (2022).

[28] L. Heimbach, E. Schertenleib, and R. Wattenhofer. "Risks and returns of uniswap v3 liquidity providers". In: *Proceedings of the 4th ACM Conference on Advances in Financial Technologies*. 2022, pp. 89–101.

[29] A. A. Aigner and G. Dhaliwal. "Uniswap: Impermanent loss and risk profile of a liquidity provider". In: *arXiv preprint arXiv:2106.14404* (2021).

[30] *Gnosis Conditional Tokens Contracts*. `https://conditional-tokens.readthedocs.io/en/latest/`. Accessed: 2025-08-03. Gnosis, 2025.

[31] *ERC1155: Multi-Token Standard (OpenZeppelin Contracts 3.x)*. `https://docs.openzeppelin.com/contracts/3.x/erc1155`. Accessed: 2025-08-03. OpenZeppelin, 2025.

[32] OpenZeppelin. *ERC-20 : Basic Token Standard (OpenZeppelin Contracts 5.x)*. `https://docs.openzeppelin.com/contracts/5.x/erc20`. Accessed: 2025-09-03. 2025.

[33] OpenZeppelin. *ERC-721 : Non-Fungible Token Standard (OpenZeppelin Contracts 5.x)*. `https://docs.openzeppelin.com/contracts/5.x/erc721`. Accessed: 2025-09-03. 2025.

[34] H. Lambur, A. Lu, and R. Cai. "Uma data verification mechanism: Adding economic guarantees to blockchain oracles". In: *Risk Labs, Inc., Tech. Rep., Jul* (2019).

[35] Polymarket Docs. *How Are Markets Resolved?* Accessed: 2025-07-20. 2024. URL: https://docs.polymarket.com/polymarket-learn/markets/how-are-markets-resolved.

[36] Bet365. *Bet365*. https://www.bet365.de. Accessed: 2025-07-03.

[37] PredictIt. *PredictIt*. https://www.predictit.org. Accessed: 2025-07-03.

[38] Kalshi. *Kalshi*. https://www.kalshi.com. Accessed: 2025-07-03.

[39] P.-J. Engelen and L. Van Liedekerke. "The ethics of insider trading revisited". In: *Journal of Business Ethics* 74 (2007), pp. 497–507.

[40] R. W. McGee. "Applying ethics to insider trading". In: *Journal of business ethics* 77 (2008), pp. 205–217.

[41] Y. Khatri. *Polymarket's huge year: $9 billion in volume and 314,000 active traders redefine prediction markets*. Accessed: 2025-07-20. The Block. 2024. URL: https://www.theblock.co/post/333050/polymarkets-huge-year-9-billion-in-volume-and-314000-active-traders-redefine-prediction-markets.

[42] *Polymarket Liquidity Rewards Program Overview*. https://docs.polymarket.com/developers/rewards/overview. Accessed: 2025-08-03. Polymarket, 2025.

[43] Alchemy. *What is a Subgraph?* Accessed: 2025-07-21. 2023. URL: https://www.alchemy.com/overviews/what-is-a-subgraph.

[44] A. Capponi, R. Jia, and B. Zhu. "The paradox of just-in-time liquidity in decentralized exchanges: More providers can sometimes mean less liquidity". In: *arXiv preprint arXiv:2311.18164* (2023).

[45] J. Milionis, C. C. Moallemi, T. Roughgarden, and A. L. Zhang. "Automated market making and loss-versus-rebalancing". In: *arXiv preprint arXiv:2208.06046* (2022).

[46] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson. *Uniswap v3 core*. Technical Whitepaper. Uniswap Labs, 2021.