Working Paper April 2026

Microstructure, Market Making, and Algorithmic Arbitrage on Decentralised Prediction Markets: A Quantitative Framework for Polymarket

Horacle Capital Research

research@horacle.capital

Abstract

This paper develops a rigorous quantitative framework for algorithmic participation in decentralised binary prediction markets, using Polymarket as the primary empirical setting. We formalise the pricing of conditional tokens as discrete Arrow–Debreu state-contingent claims, demonstrating the structural inadequacy of the Black–Scholes–Merton paradigm for event-driven discontinuous payoffs. We derive a Logit Jump-Diffusion adaptation of the Avellaneda–Stoikov market-making model that explicitly respects the $[0, 1]$ probability bound inherent to binary outcome contracts, and extend the Kelly criterion to a Brier-score-tiered capital allocation framework that scales wager size dynamically to empirically validated model accuracy. On the microstructure side, we characterise Polymarket's Central Limit Order Book (CLOB) architecture—including its ≈ 250 ms adversarial-latency protection mechanism, its Negative Risk (NegRisk) collateral-efficiency protocol, and its Liquidity Rewards programme distributing in excess of \$5 million monthly to passive market makers.

We further document two classes of structural arbitrage—intra-market rebalancing and cross-market combinatorial—that on-chain analyses estimate to have generated approximately \$40 million in cumulative alpha extraction. Finally, we address the integrity of on-chain data for copy-trading applications, noting that academic studies identify up to 18 % of historical Polymarket volume as potentially attributable to wash-trading under strict heuristic thresholds, and propose graph-theoretic filtering pipelines to purge such signals. The synthesis constitutes an operational blueprint for institutional-grade algorithmic deployment on prediction market infrastructure.

Keywords: prediction markets, market microstructure, market making, Avellaneda–Stoikov, Arrow–Debreu, Kelly criterion, Brier score, on-chain analytics, arbitrage, Polymarket, CLOB

JEL Codes: G13, G14, C61, D82

Contents

Introduction

The emergence of blockchain-native prediction markets represents a structurally significant development in the architecture of information aggregation mechanisms. Platforms such as Polymarket have evolved from niche academic prototypes to institutional venues, with monthly trading volumes reported in excess of \$21 billion as of 2026 (5). This growth has attracted a new class of algorithmic participant—quantitative funds, latency-optimised bots, and on-chain arbitrageurs—whose presence accelerates price discovery while simultaneously eroding the naive profit opportunities that once characterised such markets.

From a theoretical standpoint, binary prediction markets occupy a precise niche in the taxonomy of financial instruments: they are isomorphic to Arrow–Debreu pure state-contingent claims, contracts that pay exactly one unit of account if and only if a specific state of the world is realised. This structure differs materially from the continuous-underlying, log-normally-distributed assets that underlie the Black–Scholes–Merton framework, necessitating an entirely distinct set of pricing, hedging, and risk-management tools.

This paper proceeds as follows. Section 2 establishes the mathematical foundations, including the stochastic dynamics of binary contract prices, Bayesian information processing, and the derivation of a Logit Jump-Diffusion extension to the classical Avellaneda–Stoikov model. Section 3 develops the Brier-score-tiered Kelly framework. Section 4 analyses the microstructure of Polymarket's CLOB, covering order taxonomy, the NegRisk protocol, latency constraints, and the Liquidity Rewards programme. Section 5 examines structural arbitrage opportunities documented on-chain. Section 6 addresses algorithmic implementation, including infrastructure requirements, wash-trading detection, and copy-trading signal integrity. Section 7 concludes with implications for market efficiency.

Mathematical Framework

Stochastic Dynamics of Binary Contract Prices

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, \mathbb{P})$ be a filtered probability space. A Polymarket contract on event E with resolution time T is characterised by its payoff function:

$$\Phi(\omega) = \mathbf{1}_E(\omega), \quad \omega \in \Omega$$

where $\mathbf{1}_E$ denotes the indicator function of the event. Under the risk-neutral measure $\mathbb{Q}$, the contract price at time $t < T$ is:

$$P_t = \mathbb{E}^{\mathbb{Q}}\left[\mathbf{1}_E \mid \mathcal{F}_t\right]$$

so that $P_t \in [0,1]$ is the risk-neutral probability of event realisation conditional on the current information set. Under the maintained hypothesis of market efficiency (absence of predictable arbitrage), the sequence $\{P_t\}$ forms a $\mathbb{Q}$-martingale:

$$\mathbb{E}^{\mathbb{Q}}\left[P_{t+\Delta} \mid \mathcal{F}_t\right] = P_t, \quad \forall\,\Delta > 0$$

In continuous time, the canonical dynamics for an unconstrained asset follow Geometric Brownian Motion (GBM):

$$dS_t = \mu S_t\,dt + \sigma S_t\,dW_t$$

where W_t is a standard $\mathbb{Q}$-Wiener process. However, GBM is structurally incompatible with the $[0,1]$ boundary constraint on P_t. The probability process must instead be modelled on the logit scale. Define the log-odds transform:

$$L_t = \ln\left(\frac{P_t}{1-P_t}\right), \quad L_t \in \mathbb{R}$$

L_t is unconstrained, admitting standard diffusion dynamics. One recovers P_t via the logistic (sigmoid) function $\sigma(\cdot)$:

$$P_t = \sigma(L_t) = \frac{1}{1+e^{-L_t}}$$

Bayesian Information Processing

Since P_t encodes a probability, any informational event E_{news} should update P_t according to Bayes' theorem. Let H denote the hypothesis that the outcome resolves YES. Upon observing evidence $\mathcal{E}$, the posterior probability is:

$$P(H \mid \mathcal{E}) = \frac{P(\mathcal{E} \mid H)\cdot P(H)}{P(\mathcal{E})}$$

where the marginal likelihood is decomposed via the law of total probability:

$$P(\mathcal{E}) = P(\mathcal{E} \mid H)\cdot P(H) + P(\mathcal{E} \mid \neg H)\cdot P(\neg H)$$

An algorithm exploiting informational asymmetry must compute $P(H \mid \mathcal{E})$ faster and more accurately than competing market participants. When the posterior diverges materially from the order-book mid-price, a positive-expected-value directional position exists. The informational edge Δ is:

$$\Delta = P(H \mid \mathcal{E}) - P_{\text{mid}}$$

and persistence of the opportunity is bounded by competitive dynamics, as detailed in Section 5.

Inadequacy of Black–Scholes for Binary Prediction Contracts

For a European Cash-or-Nothing binary option with strike K on underlying S_t following GBM, the Black–Scholes–Merton price is:

$$V = e^{-rT}\mathcal{N}(d_2), \quad d_2 = \frac{\ln(S_0/K) + (r - \frac{\sigma^2}{2})T}{\sigma\sqrt{T}}$$

where $\mathcal{N}(\cdot)$ is the standard normal CDF. Application of this formula to prediction markets is structurally precluded by three considerations:

1. There exists no continuously tradable underlying S_t through which a delta-neutral hedge can be constructed. The event itself is the only "underlying," and it is not tradeable.
2. The terminal payoff is a Heaviside step function of a discrete event outcome, not a smooth function of a continuous underlying.
3. The option Greeks, particularly $\Gamma = \frac{\partial^2 V}{\partial S^2}$, diverge to infinity as $t \to T$ near the strike, rendering dynamic delta hedging mathematically intractable in the vicinity of resolution.

Remark 1. *The correct pricing framework is one of pure probabilistic modelling, consistent with Arrow–Debreu equilibrium pricing. The Conditional Token Framework (CTF) enforces the linear constraint:*

$$P_{YES} + P_{NO} = 1.00$$

Any deviation $P_{YES} + P_{NO} < 1.00$ *constitutes a riskless arbitrage opportunity via simultaneous purchase of both legs.*

Market-Making Model: Logit Jump-Diffusion Adaptation of Avellaneda–Stoikov

2.4.1 Classical Avellaneda–Stoikov Framework

Avellaneda and Stoikov (1) model the market maker's optimisation problem as a stochastic control problem. The mid-price s_t follows:

$$ds_t = \sigma\, dW_t$$

The market maker posts bid and ask quotes at distances δ^b and δ^a from the mid-price, and fills arrive as Poisson processes with intensity $\lambda(\delta)$ decreasing exponentially in the spread:

$$\lambda(\delta) = Ae^{-\kappa\delta}$$

The solution to the Hamilton–Jacobi–Bellman equation yields a *reservation price* r that adjusts the mid-price for inventory risk:

$$r = s - q \cdot \gamma \cdot \sigma^2 \cdot (T - t)$$

and an optimal symmetric spread:

$$\delta^* = \gamma\sigma^2(T - t) + \frac{2}{\gamma}\ln\left(1 + \frac{\gamma}{\kappa}\right)$$

where q is the signed inventory in units of the asset, $\gamma > 0$ is the risk-aversion parameter, and κ is the order-book depth parameter.

The bid and ask quotes are then:

$$p_b = r - \frac{\delta^*}{2}, \qquad p_a = r + \frac{\delta^*}{2}$$

2.4.2 Logit Jump-Diffusion Adaptation for Binary Markets

The classical Avellaneda–Stoikov model permits $s_t \in \mathbb{R}$, which is incompatible with a probability price constrained to $[0, 1]$. We propose the following adaptation, termed the *Logit Avellaneda–Stoikov* (LAS) model.

Definition 1 (LAS Model). *Let $L_t = \sigma(P_t)^{-1} = \ln(P_t/(1 - P_t))$ be the logit-transformed mid-price. The market maker operates in logit space, computing the reservation logit:*

$$\ell = L - q \cdot \gamma \cdot \sigma_L^2 \cdot (T - t)$$

where σ_L^2 is the variance of dL_t per unit time, estimated from recent order-book *dynamics. The reservation price is then recovered:*

$$r = \sigma(\ell) = \frac{1}{1 + e^{-\ell}}$$

The optimal spread in probability space is asymmetric around r due to the non-linear logistic mapping:

$$\delta_P = \sigma\left(\ell + \frac{\delta_L^*}{2}\right) - \sigma\left(\ell - \frac{\delta_L^*}{2}\right)$$

where $\delta_L^* = \gamma\sigma_L^2(T - t) + \frac{2}{\gamma}\ln(1 + \gamma/\kappa)$ is the spread computed in logit space. This formulation guarantees $p_b, p_a \in (0, 1)$ and naturally widens the spread near the boundaries $P \approx 0$ and $P \approx 1$, where inventory risk is concentrated.

Proposition 1 (Boundary Saturation)**.** *As $P_t \to 0$ or $P_t \to 1$, the logistic function's derivative approaches zero, implying that for a fixed logit-space spread δ_L^*, the probability-space spread $\delta_P \to 0$. Concurrently, the reservation logit ℓ diverges, and the mapped reservation price saturates at the boundary. The algorithm must therefore impose a floor $p_b \geq \epsilon$ and ceiling $p_a \leq 1 - \epsilon$ for small $\epsilon > 0$ (practically, $\epsilon = 0.01$) to maintain valid order submissions.*

BRIER-SCORE-TIERED CAPITAL ALLOCATION

Standard Kelly Criterion

The Kelly criterion (2) maximises the expected logarithmic growth rate of a bankroll W by determining the optimal fractional wager f^*. For a binary bet at implied odds b (net profit per unit staked) with model probability p of success:

$$f^* = \frac{bp - q}{b}$$

where $q = 1 - p$ is the model probability of failure. If a YES token is purchased at market price P_m, the net odds are $b = (1 - P_m)/P_m$, and the full Kelly fraction is:

$$f^* = \frac{\left(\frac{1-P_m}{P_m}\right)p - (1 - p)}{\frac{1-P_m}{P_m}} = p - P_m$$

This simplification reveals that the full Kelly fraction is exactly equal to the model's edge over the market price.

Limitations of Fractional Kelly and the Brier-Tiered Extension

Standard institutional practice applies a fixed scalar multiplier to f^* (commonly 0.25 or 0.5) to account for model uncertainty. This approach is epistemically inadequate:

it does not distinguish between a model with a strong empirical track record and a model that has been calibrated on limited data.

We propose a principled alternative based on the **Brier Score** $\mathcal{B}$, defined as the mean squared error between model probability forecasts $\hat{p}_i$ and binary outcomes $o_i \in \{0, 1\}$ over a historical sample of N predictions:

$$\mathcal{B} = \frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - o_i)^2$$

A perfect model achieves $\mathcal{B} = 0$; a naive constant model predicting 0.5 achieves $\mathcal{B} = 0.25$. Lower scores indicate superior calibration.

Definition 2 (Brier-Tiered Kelly Multiplier)**.** *Define the dynamic Kelly multiplier $\alpha(\mathcal{B})$ as a piecewise linear function of the model's rolling Brier score:*

$$\alpha(\mathcal{B}) = \begin{cases} 0.40 & \textit{if } \mathcal{B} < 0.18 \quad \textit{(high-precision regime)} \\ 0.25 & \textit{if } 0.18 \leq \mathcal{B} < 0.22 \quad \textit{(moderate-precision regime)} \\ 0.10 & \textit{if } \mathcal{B} \geq 0.26 \quad \textit{(low-precision regime)} \end{cases}$$

The adjusted capital fraction is:

$$f_{adj} = \alpha(\mathcal{B}) \cdot f^*$$

Remark 2. *The choice of $\alpha = 0.40$ as the upper bound, rather than 1.0 (Full Kelly), reflects the irreducible model risk in prediction markets. Even a well-calibrated model does not account for tail risks such as resolution disputes, oracle manipulation, or correlated information arrival. A conservative ceiling on α is therefore justified on grounds of ruin-prevention. The multiplier $\alpha(\mathcal{B})$ is re-estimated on a rolling window of 50 completed positions to ensure responsiveness to regime changes in model accuracy.*

MARKET MICROSTRUCTURE AND ARBITRAGE

CLOB Architecture and Order Taxonomy

Polymarket operates a Central Limit Order Book (CLOB) with a hybrid on-chain/off-chain settlement architecture (6). Order matching is performed off-chain by a centralised engine, while settlement of matched trades is executed on-chain via Gnosis Conditional Token Framework (CTF) smart contracts deployed on the Polygon PoS network (Chain ID 137). This architecture achieves sub-second matching latency while benefiting from the non-custodial settlement guarantees of smart contracts.

Orders are authenticated via EIP-712 typed structured data signatures and classified by execution semantics:

Type	Execution Semantics	Algorithmic Use Case
GTC (Good-Till-Cancelled)	Rests passively on the book until filled or cancelled.	Continuous passive liquidity provision (market making).
GTD (Good-Till-Date)	Expires at a specified UTC timestamp; residual cancelled.	Pre-scheduled risk management ahead of catalyst events.
FOK (Fill-Or-Kill)	Entire order filled immediately or fully cancelled.	Arbitrage execution requiring atomic completion guarantees.
FAK (Fill-And-Kill)	Fills maximum available immediately; residual cancelled.	Directional sweep accumulation under high conviction.

Adversarial Latency Protection Mechanism

A critical structural feature distinguishing Polymarket's CLOB from traditional financial exchanges is the imposition of a **matching delay** of approximately 250 milliseconds applied to aggressive (taker) orders on volatile markets such as live sports (7). This mechanism is designed to protect passive liquidity providers (makers) from adverse selection by participants with superior physical-proximity information advantages.

The practical implication for high-frequency trading strategies is significant: the sub-100 ms execution windows that characterise profitable simple arbitrage on traditional venues are structurally precluded for taker orders. Instead, algorithmic alpha extraction is optimally concentrated in:

1. **Passive market making** (maker orders), which bypass the matching delay and earn Maker Rebates.
2. **Cross-market combinatorial arbitrage**, where execution is spread across multiple book interactions and the 250 ms delay per leg can be pipeline-parallelised.

Algorithms that rely on fill-or-kill sweeps for simple YES+NO sum arbitrage face a further structural headwind: in 2026, the mean lifespan of a sub-1 cent pricing inefficiency has compressed to approximately 2.7 seconds, with 73 % of available profit captured by co-located HFT engines executing in under 100 ms (15).

Liquidity Rewards and Maker Rebates

Polymarket operates a substantial incentive programme distributing over \$5 million monthly to passive liquidity providers, modelled on the dYdX liquidity mining protocol (8). Individual position scores are computed as a quadratic function of proximity to the mid-price:

$$S(v, s) = \left(\frac{v - s}{v}\right)^2 \cdot b$$

where v is the protocol's maximum qualifying spread (max spread), s is the order's distance from the adjusted mid-price, and b is the order size in USDC.e. The epoch minimum score is:

$$Q_{\text{min}} = \min(Q_{\text{bid}}, Q_{\text{ask}})$$

requiring simultaneous two-sided presence; unilateral liquidity receives zero reward.

In addition to epoch rewards, the Maker Rebates programme returns a fraction of transaction fees to market makers: 20 % on cryptocurrency markets and 25 % on sports, political, and financial markets (9).

NegRisk Protocol and Capital Efficiency

For multi-outcome mutually exclusive markets (e.g., tournament winner markets), Polymarket deploys the **Negative Risk** mechanism (10). Classical multi-outcome betting requires separate collateral per outcome; NegRisk exploits the combinatorial constraint that exactly one outcome must prevail. The NegRisk Adapter contract (deployed at `0xd91E80cF2E7be2e162c6513ceD06f1dD0dA35296`) enables the atomic conversion:

$$1 \times \text{NO}_j \leftrightarrow \sum_{i \neq j} 1 \times \text{YES}_i$$

This equivalence permits the following capital-efficient arbitrage logic. Consider a market with n outcomes. If:

$$\text{Cost}(\text{NO}_j) < \sum_{i \neq j} P_{\text{YES}_i}$$

a riskless profit is available by purchasing NO_j, converting via the adapter, and selling the resulting YES tokens at prevailing market prices. Algorithms must set the API flag `negRisk: true` and interact with the dedicated NegRisk CTF exchange contract (`0xC5d563A36AE78145C45a50134d48A1215220f80a`) for eligible operations.

Inventory Management: Split, Merge, Redeem

Two-sided market making requires conditional collateral rather than cash alone. The CTF contract (`0x4D97DCd97eC945f40cF65F87097ACe5EA0476045`) supports three gasless operations (11):

Split. Deposit x USDC.e; receive x YES tokens and x NO tokens. Enables simultaneous ask-side quotation on both outcomes.

Merge. Deposit x YES and x NO tokens; recover x USDC.e. Releases capital from offsetting positions.

Redeem. Post-resolution, deposit winning tokens; recover USDC.e at a 1:1 rate.

STRUCTURAL ARBITRAGE: ON-CHAIN EVIDENCE

Taxonomy of Residual Arbitrage Opportunities

The naive strategy of exploiting deviations from the $P_{\text{YES}} + P_{\text{NO}} = 1$ no-arbitrage constraint has been substantially eliminated by competition (15). Empirical analysis of order-book history documents the compression of mean inefficiency duration from 12.3 seconds in 2024 to 2.7 seconds in 2026 at the simple arbitrage level. Two structurally durable alpha sources have been identified in the academic literature, with cumulative documented profits of approximately \$40 million (3):

5.1.1 Intra-Market Rebalancing Arbitrage

Liquidity imbalances within a single mutually-exclusive market arise when uninformed capital flows asymmetrically into one outcome leg, temporarily distorting the probability sum. The algorithm continuously monitors the sum $\sum_i P_{\text{YES}_i}$ across all outcomes in a NegRisk market. When:

$$\sum_i P_{\text{YES}_i} < 1 - \epsilon_{\text{threshold}}$$

for a calibrated threshold $\epsilon_{\text{threshold}}$, the algorithm executes a proportional buy across all underpriced YES legs simultaneously, relying on the binding constraint that exactly one outcome resolves to YES.

5.1.2 Cross-Market Combinatorial Arbitrage

Probabilistic dependencies between logically related events on separate markets give rise to cross-market mispricings. Let A and B be two events with a logical implication $A \Rightarrow B$. Under no-arbitrage, the conditional probability structure implies:

$$P(A) \leq P(B)$$

Any persistent inversion of this inequality is exploitable. The search problem across n markets and m event types has worst-case complexity $O(2^{n+m})$; in practice, a directed acyclic graph (DAG) representation of event dependencies, combined with BFS-enumeration from known anchor nodes, reduces this to tractable polynomial-time search for markets with sparse dependency structures.

Example: If the market for *"Temperature above 30°C on Monday"* trades at $P = 0.80$ and the market for *"Temperature above 30°C for the entire week"* trades at $P = 0.90$, the latter implies the former, yet $0.90 > 0.80$ violates the necessary constraint $P(\text{week}) \leq P(\text{Monday})$, generating a riskless position.

Algorithmic Implementation

Infrastructure Requirements

The deployment of production-grade algorithmic strategies imposes strict infrastructure constraints. The server hosting the bot's execution engine must be located in a Virtual Private Server (VPS) environment physically co-located with Polymarket's API infrastructure to minimise round-trip latency (14). The target architecture is:

- **Blockchain RPC:** Dedicated private Polygon RPC nodes (e.g., Alchemy, Infura) to bypass rate-limited public endpoints. The Polymarket API enforces strict rate limits of 15,000 standard requests per 10 seconds and 1,000 data requests per 10 seconds per IP; violations trigger immediate IP bans that block emergency inventory liquidation (13).
- **Order Flow:** Persistent bidirectional WebSocket connections for real-time order-book feed subscription, replacing polling REST calls which incur per-request latency penalties.
- **Authentication:** Level 2 API credentials derived from a Level 1 blockchain private key signature, handled via the official `py-clob-client` library.

Python Client Implementation

The following illustrates the initialisation of a market-making bot placing a batch of bid and ask orders via the Polymarket CLOB client:

Listing 1: Market-making order submission via py-clob-client

```python
import os
from py_clob_client.client import ClobClient
from py_clob_client.clob_types import OrderArgs, OrderType,
    Side

HOST = "https://clob.polymarket.com"
CHAIN_ID = 137  # Polygon PoS

client = ClobClient(HOST, key=os.getenv("POLY_PRIVATE_KEY"),
    chain_id=CHAIN_ID)
client.set_api_creds(client.create_or_derive_api_creds())

TOKEN_ID = "
    71321045679252212594626385532706912750332728571942532289631379312455583399
    "

# Compute reservation price and spread from LAS model (
    external)
p_b, p_a = compute_las_quotes(mid_price, inventory, gamma,
    sigma_L, T_minus_t, kappa)

bid_order = client.create_order(OrderArgs(
     token_id=TOKEN_ID, price=round(p_b, 3),
     size=500.0, side=Side.BUY
))
ask_order = client.create_order(OrderArgs(
     token_id=TOKEN_ID, price=round(p_a, 3),
     size=500.0, side=Side.SELL
))

# Batch submission: up to 15 orders per cycle for throughput
    maximisation
response = client.post_orders([bid_order, ask_order],
    OrderType.GTC)
```

In production, the `post_orders` batch endpoint (accepting up to 15 orders per cycle) is preferred over sequential single-order submissions to maximise throughput and minimise the window during which the bid-ask structure is partially posted (12).

On-Chain Signal Integrity: Wash Trading Detection

A fundamental challenge in constructing copy-trading strategies from on-chain data is the contamination of volume metrics by wash trading. An academic study by Columbia University estimates that up to 18 % of Polymarket's historical aggregate volume, under strict heuristic thresholds, is attributable to circular trading operations (4). These are executed primarily to extract Liquidity Rewards, token distribution subsidies, or leaderboard rankings, and produce spurious performance metrics in any wallet-ranking system.

A rigorous wash-trading detection pipeline applies **graph-theoretic analysis** to the transaction graph:

1. Construct a directed graph $G = (V, E)$ where nodes $v \in V$ are wallet addresses and each edge $e = (u, v, w)$ represents a token transfer of value w from u to v.
2. Identify strongly connected components (SCCs) with high internal edge volume relative to net profit. If two nodes chronically exchange large positions while their combined net P&L approaches zero ($\lim_{t\to\infty} \mathrm{PnL}_u + \mathrm{PnL}_v \approx 0$), both are flagged as wash-trading entities.
3. Exclude flagged addresses from the copy-trading candidate set.

For a copy-trading signal to pass the institutional validation filter, the target wallet must satisfy all of the following criteria simultaneously:

Metric	Threshold	Rationale
Net historical P&L	> \$200,000	Validates capital deployment at scale.
Win rate	60–70 %	Exceeds Poisson random baseline.
Trading volume	> \$100,000	Confirms liquidity-adequate market participat
Position history	> 50 positions over 6 months	Eliminates survivorship bias from single large

Conclusion

This paper has developed a comprehensive quantitative framework for algorithmic participation in decentralised binary prediction markets. Several conclusions emerge from the analysis.

First, the mispricing of binary prediction contracts through the lens of continuous-underlying derivatives models (Black–Scholes–Merton) is a category error. The correct framework is Arrow–Debreu equilibrium pricing constrained by the linear no-arbitrage identity $P_{\mathrm{YES}} + P_{\mathrm{NO}} = 1$, with all risk management conducted through probability-native tools: Bayesian inference for information processing and the Kelly criterion for capital sizing.

Second, the proposed Logit Avellaneda–Stoikov (LAS) model addresses a fundamental deficiency in applying classical market-making theory to bounded probability spaces. By conducting the reservation-price and spread computations in logit space before mapping back through the logistic function, the LAS model guarantees boundary-consistent quotes and naturally accommodates the fat-tailed dynamics near $P = 0$ and $P = 1$.

Third, the Brier-score-tiered Kelly framework represents a principled improvement over fixed fractional Kelly, linking wager sizing directly to empirical model accuracy. This is particularly relevant in prediction markets where the quality of the underlying probabilistic model varies significantly across event categories and information environments.

Fourth, on the microstructure side, the 250 ms matching delay imposed on taker orders, combined with the Liquidity Rewards programme and Maker Rebates structure, creates a favourable environment for passive market-making strategies relative to aggressive latency-racing. Algorithms that optimise for maker status and two-sided presence extract alpha from multiple simultaneous revenue streams: the bid-ask spread, epoch liquidity rewards, and maker fee rebates.

Finally, the documented structural persistence of cross-market combinatorial mispricings—generating an estimated \$40 million in cumulative alpha—suggests that prediction market efficiency at the inter-market level lags significantly behind intra-market efficiency. Algorithmic strategies that model event dependency structures through DAG representations remain the most promising source of durable, scalable alpha on these platforms.

The maturation of prediction market microstructure, the codification of on-chain data analytics, and the development of probability-native stochastic control models constitute a productive research agenda at the intersection of market design, mechanism theory, and quantitative finance.

References

[1] Avellaneda, M., & Stoikov, S. (2008). High-frequency trading in a limit order book. *Quantitative Finance*, 8(3), 217–224.

[2] Kelly, J. L. (1956). A new interpretation of information rate. *Bell System Technical Journal*, 35(4), 917–926.

[3] Anonymous. (2025). Unravelling the probabilistic forest: Arbitrage in prediction markets. *arXiv preprint arXiv:2508.03474*.

[4] Columbia University. (2026). Network-based detection of wash trading. *Working Paper, Columbia Business School*.

[5] TRM Labs. (2026). How prediction markets scaled to USD 21B in monthly volume in 2026. *TRM Labs Research Report*.

[6] Polymarket. (2026). Prices & Orderbook. *Polymarket Documentation*. https://docs.polymarket.com/concepts/prices-orderbook

[7] Polymarket. (2026). Order lifecycle. *Polymarket Documentation*. https://docs.polymarket.com/concepts/order-lifecycle

[8] Polymarket. (2026). Liquidity rewards. *Polymarket Documentation*. https://docs.polymarket.com/market-makers/liquidity-rewards

[9] Polymarket. (2026). Maker rebates program. *Polymarket Documentation*. https://docs.polymarket.com/market-makers/maker-rebates

[10] Polymarket. (2026). Negative risk markets. *Polymarket Documentation*. https://docs.polymarket.com/advanced/neg-risk

[11] Polymarket. (2026). Trading. *Polymarket Documentation*. https://docs.polymarket.com/market-makers/trading

[12] Polymarket. (2026). Create order. *Polymarket Documentation*. https://docs.polymarket.com/trading/orders/create

[13] Polymarket. (2026). Rate limits. *Polymarket Documentation*. https://docs.polymarket.com/api-reference/rate-limits

[14] QuantVPS. (2026). Running Polymarket bots on a VPS: Why low latency matters. *QuantVPS Blog*. https://www.quantvps.com/blog/running-polymarket-bots-on-vps

[15] Beyond simple arbitrage: 4 Polymarket strategies bots actually profit from in 2026. *Medium / Illumination*, 2026.