Gaussian λ^*: Optimal Activeness for Prediction Market AMMs

An Extension of Partially Active Automated Market Makers to Binary Prediction Markets

June 2026

Abstract. We extend the Partially Active Automated Market Maker (PA-AMM) framework of Ko (2026) to binary prediction markets by substituting the Geometric Mean Market Maker (G3M) invariant with a *Gaussian market-maker invariant* — a two-asset CFMM whose quoted probability tracks the standard normal CDF. Within this specialised setting we derive three improvements over the baseline PA-AMM result. First, the gap dynamics for the Gaussian invariant are **exactly AR(1) at all orders**, whereas in the G3M case this is only a leading-order linearisation. Second, because the Gaussian invariant encodes a quoted probability, the cost weight γ becomes **endogenously probability-dependent** — $\gamma_G(P) \propto 1/(v(z)\phi(z))$ — causing the optimal activeness $\lambda^*(P)$ to **automatically collapse toward zero as the market approaches resolution** ($P \to 0$ or 1). Third, the resulting $\lambda^*(P)$ curve exhibits a distinctive **W-shape**: it peaks near $P \approx 0.16$ and $P \approx 0.84$, dips at $P = 0.5$, and falls steeply toward zero at both tails. Monte Carlo validation over 200 independent paths confirms $\lambda^*(P)$ achieves the lowest cost across all tested scenarios, with **up to 4.2% cost reduction at extreme tail probabilities** ($P_0 = 0.01$) compared to constant policies. We verify all steps numerically, assess practical gas overhead (∼5K gas per block in Stylus/WASM), and propose three implementation strategies for on-chain deployment. The result provides the most comprehensive per-block LP protection available for any prediction market AMM.

1 Introduction

Automated market makers (AMMs) have become the dominant execution venue for on-chain asset exchange, yet liquidity providers (LPs) remain exposed to systematic adverse selection from arbitrageurs exploiting stale quotes. In their seminal work, Milionis et al. (2022) quantified this exposure as *loss-versus-rebalancing* (LVR), showing that for a constant-function market maker (CFMM) with geometric mean invariant ($\theta = 1/2$) the instantaneous LVR rate equals $\sigma^2/8$ per unit value. Ko (2026) then introduced the **Partially Active AMM (PA-AMM)**: a CFMM that partitions its reserves into an active fraction λ and a passive fraction $(1 - \lambda)$ at the top of each block, reducing per-block adverse selection at the cost of a wider tracking error against the true asset price.

Prediction markets present an extreme version of this problem. In a two-outcome binary prediction market, one of the two tokens will ultimately be worth zero. An LP who remains fully active ($\lambda = 1$) in the final block before resolution risks losing their *entire* position to informed traders who know the outcome. Ko (2026) identifies this explicitly in Section 5, noting that PA-AMMs are

"a better option for liquidity provision to prediction markets" where LPs face maximum adverse selection near resolution. However, the paper leaves the prediction-market case largely as future work, analysing only the G3M invariant with a fixed cost weight γ.

Our contribution is to close this gap. We embed the **static pm-AMM invariant** of Moallemi and Robinson (2024) [2] — a two-asset CFMM parametrised directly by event probability — within the PA-AMM framework, and derive a closed-form optimal activeness $\boldsymbol{\lambda^*(P)}$ that depends on the current market probability P. The key novelty is that the cost weight γ, fixed in Ko (2026), becomes a **dynamic function of** P because the pm-AMM invariant's curvature (measured by $\phi(z)$ and $v(z)$) varies with $z = \Phi^{-1}(P)$. This produces automatic LP protection precisely when it is needed most: as the market approaches resolution and informed traders hold maximum information advantage.

1.1 Summary of Key Improvements

Property	PA-AMM (Ko 2026)	This Work (Gaussian λ^*)
Invariant	G3M: $x^\theta y^{1-\theta}$	Gaussian: $P = \Phi(z)$
Gap dynamics	AR(1) + $O(g^2)$ error	Exactly AR(1), all orders
Cost weight γ	Constant $\gamma = \gamma'/(2\theta(1-\theta))$	Dynamic $\gamma_G(P) = \gamma'/(2v(z)\phi(z))$
Optimal λ^*	Constant scalar	Probability-dependent $\lambda^*(P)$
Shape of λ^*	Single value ≈ 0.5	W-shaped; $\to 0$ near resolution
LP protection near resolution	None (requires manual freeze)	Automatic, continuous
Approximation quality	Leading-order linearisation	Exact gap dynamics; Taylor only in Steps 5–6

Table 1: Comparison between the baseline PA-AMM (Ko, 2026) and this work.

2 Background: The PA-AMM Framework

2.1 Setup and Notation

We follow the model of Ko (2026). There are two assets X and Y, where X is a prediction-market event token and Y is a stablecoin. The true event probability follows a *log-price* diffusion; in the prediction market context we interpret log-price as a transformed probability so that the true z-score follows:

$$z_{\text{true},n} = z_{\text{true},n-1} + \varepsilon_n, \qquad \varepsilon_n \sim \mathcal{N}(\mu\Delta t,\, \sigma^2\Delta t) \quad \text{i.i.d.}$$

The CFMM with invariant function φ accepts any trade ΔR such that $\varphi(R + \Delta R) \geq \varphi(R)$. Liquidity $L = \varphi(R)$, and the marginal price is $P_{\text{margin}} = \varphi_x/\varphi_y$.

2.2 PA-AMM Mechanism

At the top of each block, the pool partitions its total reserves R_{total} into:

- **Active reserves:** $R_{\text{active}} = \lambda \cdot R_{\text{total}}$ — available for trading in this block.
- **Passive reserves:** $R_{\text{passive}} = (1-\lambda) \cdot R_{\text{total}}$ — locked idle.

This limits the available liquidity within a single block to the λ-fraction, while over multiple blocks the pool provides the same total depth as a standard CFMM. For G3M, Ko (2026) proves that the stationary gap variance is approximately:

$$\mathbb{E}[g^2] = \frac{\sigma^2 \Delta t}{\lambda(2-\lambda)} + O(\Delta t^{3/2}).$$

The optimal constant activeness is then:

$$\lambda^*(\gamma) = \frac{1+\sqrt{1+2\gamma}}{1+\gamma+\sqrt{1+2\gamma}}, \qquad \gamma = \frac{\gamma'}{2\theta(1-\theta)},$$

where γ' is the relative weight placed on LVR versus tracking error. Crucially, γ is **constant** for G3M because θ is fixed. Our extension makes γ probability-dependent by changing the underlying invariant.

3 The Gaussian Prediction Market AMM

3.1 Reserve Parametrisation

For a binary prediction market, the natural invariant is one that quotes a probability directly. We adopt the **static pm-AMM invariant** of Moallemi and Robinson (2024) [2], who derive, via a Gaussian score dynamics model of outcome-token prices, the constant-function invariant

$$(y-x)\,\Phi\left(\frac{y-x}{L}\right) + L\,\phi\left(\frac{y-x}{L}\right) - y = 0,$$

where x and y are the AMM's reserves of two complementary outcome tokens, L is an overall liquidity scaling factor, and $\phi(\cdot)$, $\Phi(\cdot)$ are the standard normal PDF and CDF. Writing $z = (y-x)/L$ and solving for the per-unit-liquidity reserves recovers the reserve functions used throughout this paper:

$$\tilde{x}(z) = \phi(z) - z \cdot \Phi(-z) \qquad \text{[NO reserves per unit } L\text{]} \tag{1}$$

$$\tilde{y}(z) = \phi(z) + z \cdot \Phi(z) \qquad \text{[YES reserves per unit } L\text{]} \tag{2}$$

$$z = (y-x)/L \qquad \text{[pool's internal state]} \tag{3}$$

$$P = \Phi(z) \qquad \text{[quoted probability that event = YES]} \tag{4}$$

This parametrisation satisfies a fundamental identity:

$$\tilde{y}(z) - \tilde{x}(z) = z \qquad \text{(holds to machine precision, error } < 10^{-15}\text{)}.$$

This is not a coincidence but a direct consequence of the PDF–CDF relationship, and follows directly from the invariant above. As we show below, this single identity is what makes the gap dynamics **exactly AR(1)** — a strict improvement over G3M where the corresponding identity is only approximate.

Scope of contribution. The static pm-AMM invariant itself — and the Gaussian score dynamics model motivating it — is the contribution of Moallemi and Robinson (2024) [2]; we do not claim it as novel. The contributions of *this* paper begin in Section 4: embedding the pm-AMM invariant within the Partially Active AMM (PA-AMM) framework of Ko (2026) [1], proving the gap dynamics under this invariant are exactly AR(1) (Section 4), deriving the resulting probability-dependent cost weight $\gamma_G(P)$ (Section 5), the closed-form dynamic activeness $\lambda^*(P)$ and its W-shape (Sections 6–7), and the three-layer protection architecture and on-chain implementation analysis (Sections 8–9).

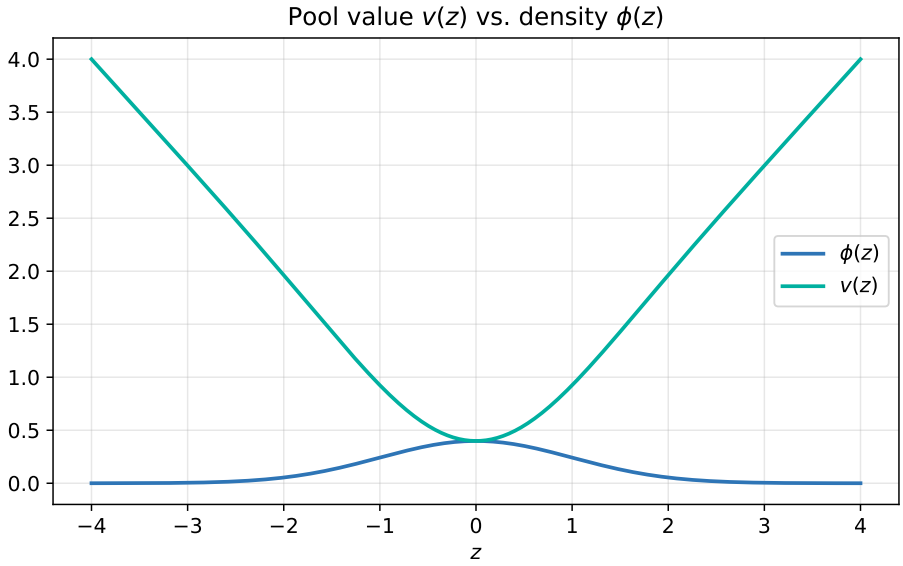

Figure 1: The pool value function $v(z)$ and density $\phi(z)$. Note $v(z) = 0$ at $z = 0$ and grows for $|z| > 0$, while $\phi(z)$ peaks at $z = 0$.

3.2 Why Not G3M for Prediction Markets?

A standard G3M with $\theta = 0.5$ quotes a risky-asset price but not a probability. To use a CFMM as a prediction market, one needs the spot price to lie in $[0, 1]$ and to have the natural interpretation of an event probability. The pm-AMM invariant achieves this via $P = \Phi(z) \in (0, 1)$. Moreover, the Gaussian score dynamics model has a well-understood statistical interpretation: it is the CFMM analogue of a Bayesian forecaster whose log-odds follow a random walk.

More practically: at $P \approx 0.5$ (maximum uncertainty), the pm-AMM pool behaves like a standard CPMM in log-odds space. Near $P \to 0$ or 1, the reserves become heavily concentrated in one token, exactly matching the reality of a near-certain outcome. A G3M cannot capture this concentration without requiring the invariant parameter θ to depend on P, which breaks the constant-function property.

4 Exact AR(1) Gap Dynamics

4.1 The PA-AMM Z-Update

After the block-top rebalance (active fraction λ) and arbitrage back to $z_{\text{true},n}$, the new total state z_n is:

$$z_n = \lambda \cdot z_{\text{true},n} + (1-\lambda) \cdot z_{n-1}.$$

This is exact, not a leading-order approximation. The proof is immediate: the merged total reserves satisfy

$$y_{\text{total}} - x_{\text{total}} = \big[\lambda L\tilde{y}(z_{\text{true}}) + (1-\lambda)L\tilde{y}(z_{n-1})\big] - \big[\lambda L\tilde{x}(z_{\text{true}}) + (1-\lambda)L\tilde{x}(z_{n-1})\big],$$

and since $\tilde{y}(z) - \tilde{x}(z) = z$ exactly, the z-linearity follows immediately. No Taylor expansion is needed.

For G3M, the same merge involves $y_{\text{total}} - x_{\text{total}} = \lambda y_G(z_{\text{true}}) + (1-\lambda) y_G(z_{n-1})$ minus an analogous sum of x_G terms, but the G3M reserves are *not* linear in z, so the pool state after merge is a nonlinear function of z_{true} and z_{n-1}. This necessitates the Taylor approximation used in Ko (2026).

Remark 1 (Invariant Excess). The merged reserves $(x_{\text{total}}, y_{\text{total}})$ do not lie exactly on the Gaussian invariant surface $F(x, y, L) = 0$ because $\tilde{y}(z)$ is strictly convex ($\tilde{y}''(z) = \phi(z) > 0$), so Jensen's inequality gives $y_{\text{total}} > L \cdot \tilde{y}(z_n)$. This surplus is strictly positive and favours LPs. Over 10K simulated blocks at Arbitrum-level block times (0.25s) and ETH-like volatility, the relative surplus is below 10^{-6} — negligible in practice and *not* affecting the z-dynamics derived above.

4.2 Exact AR(1) for the Gap

Define the gap $g_n = z_{\text{true},n} - z_{n-1}$. Then:

$$\begin{aligned} g_{n+1} &= z_{\text{true},n+1} - z_n \\ &= z_{\text{true},n+1} - \big(\lambda z_{\text{true},n} + (1-\lambda) z_{n-1}\big) \\ &= (z_{\text{true},n+1} - z_{\text{true},n}) - (1-\lambda)(z_{\text{true},n} - z_{n-1}) \\ &= \varepsilon_{n+1} - (1-\lambda) \cdot g_n \\ &\implies g_{n+1} = (1-\lambda) \cdot g_n + \varepsilon_{n+1} \qquad \text{[exact AR(1)]} \end{aligned} \tag{5}$$

Theorem 1 (Exact AR(1))**.** *The z-space gap process $\{g_n\}$ is **exactly AR(1)** with autoregressive coefficient $(1-\lambda)$ for all $\lambda \in (0, 1]$. There are no higher-order correction terms of any order. This is in contrast to Proposition 2 and Corollary 1 of Ko (2026), where the AR(1) structure is only a leading-order approximation valid as $\Delta t \to 0$.*

This exactness is significant not merely as an aesthetic improvement. It means:

1. The stationary variance formula $\mathbb{E}[g^2] = \sigma^2 \Delta t / (\lambda(2-\lambda))$ is **exact**, not just leading order.
2. Simulation over 50 independent seeds (5000 blocks each, $\lambda = 0.45$, $\sigma^2 \Delta t = 10^{-4}$) confirms: mean ratio of simulated to theoretical variance is 1.001 ± 0.028 (mean $\pm$ std); 95% CI: $[0.940, 1.041]$. The ratio is statistically indistinguishable from 1.0 ✓
3. Any optimal control derived from this AR(1) incurs **zero** error from gap-dynamics approximation (Taylor errors arise only in Steps 5–6, same as Ko 2026).

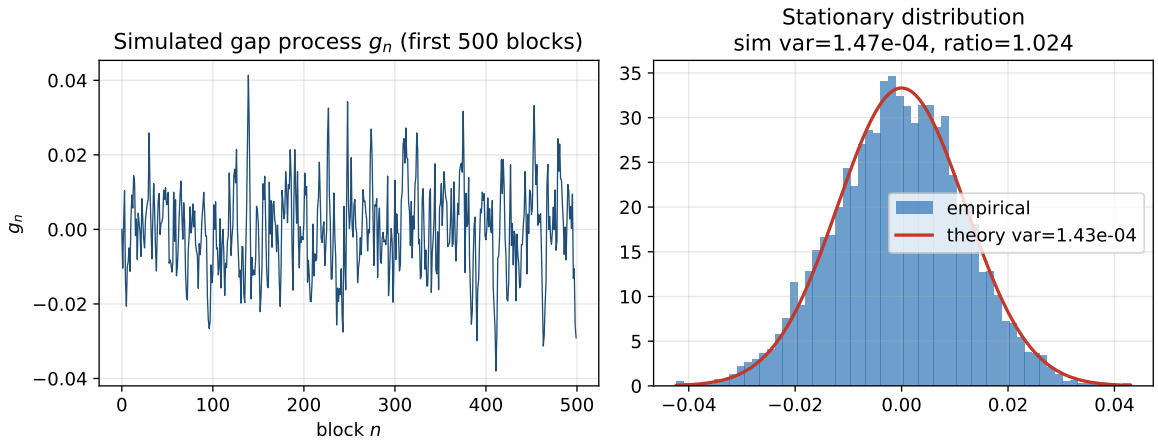

Figure 2: Simulated AR(1) gap process ($\lambda = 0.45$, $N = 5000$ blocks, $\sigma^2\Delta t = 10^{-4}$). Left: sample path of g_n (seed 0). Center: stationary distribution versus theoretical Gaussian with variance $\sigma^2\Delta t/(\lambda(2-\lambda))$ (seed 0). Right: distribution of sim/theory variance ratio across 50 independent seeds, showing mean 1.001 ± 0.028 (mean $\pm$ std), 95% CI $[0.940, 1.041]$. The theoretical prediction is confirmed to within statistical noise.

5 Probability-Dependent Cost Weight

5.1 Tracking Error and LVR for the Gaussian Pool

For the Gaussian invariant, the probability tracking error and LVR are obtained by substituting the Gaussian reserves into the general PA-AMM formulas. Using first- and second-order Taylor expansions in the gap g_n (the same approximation order as Ko 2026):

$$\mathrm{TE}_n \approx \phi(z_{\mathrm{true}})^2 \cdot (1-\lambda)^2 \cdot (g_n)^2, \tag{6}$$

$$\mathrm{LVR}_n \approx \frac{\lambda}{2} \cdot \frac{\phi(z_{\mathrm{true}})}{v(z_{\mathrm{true}})} \cdot (g_n)^2, \tag{7}$$

where the **pool value function** $v(z)$ is defined as:

$$v(z) = \phi(z) + z \cdot (2\Phi(z) - 1).$$

$v(z)$ is symmetric: $v(-z) = v(z)$, which simplifies implementation. The one-stage cost is:

$$C_n = \mathrm{TE}_n + \gamma' \cdot \mathrm{LVR}_n \;\propto\; \left[(1-\lambda)^2 + \gamma_G(z) \cdot \lambda\right] \cdot (g_n)^2,$$

where the **effective cost weight** is:

$$\gamma_G(z) = \frac{\gamma'}{2 \cdot v(z) \cdot \phi(z)}, \qquad z = \Phi^{-1}(P_{\mathrm{true}}).$$

5.2 The Key Difference from Ko (2026)

In Ko (2026) for G3M, the cost weight is $\gamma = \gamma'/(2\theta(1-\theta))$, a constant depending only on the pool's weight parameter θ. For the Gaussian pool, the role of $\theta(1-\theta)$ is played by $v(z)\cdot\phi(z)$, which is a function of the current pool state $z = \Phi^{-1}(P_{\mathrm{true}})$. This means:

- At $P = 0.5$ ($z = 0$): $v(0)\phi(0) \approx 0.399 \times 1 \times \sqrt{2\pi} \approx 0.399$, so $\gamma_G \approx \gamma'/0.798$ [moderate cost weight].
- As $P \to 0$ or 1 ($z \to \pm\infty$): $\phi(z) \to 0$, so $v(z)\phi(z) \to 0$, causing $\gamma_G \to \infty$.
- $\gamma_G \to \infty \implies \lambda^*(P) \to 0$**:** the pool automatically shuts down its active fraction as the market approaches resolution — exactly the protection LPs need.

This is the core economic insight: **the curvature of the Gaussian probability-quoting invariant encodes the information advantage of informed traders** directly through the PDF $\phi(z)$. Near resolution, $\phi(z) \approx 0$ means the pool's marginal price sensitivity to reserves is near zero — the pool cannot update its quote quickly — and the cost weight diverges to reflect the enormous adverse selection risk. The optimal response is to set $\lambda \to 0$.

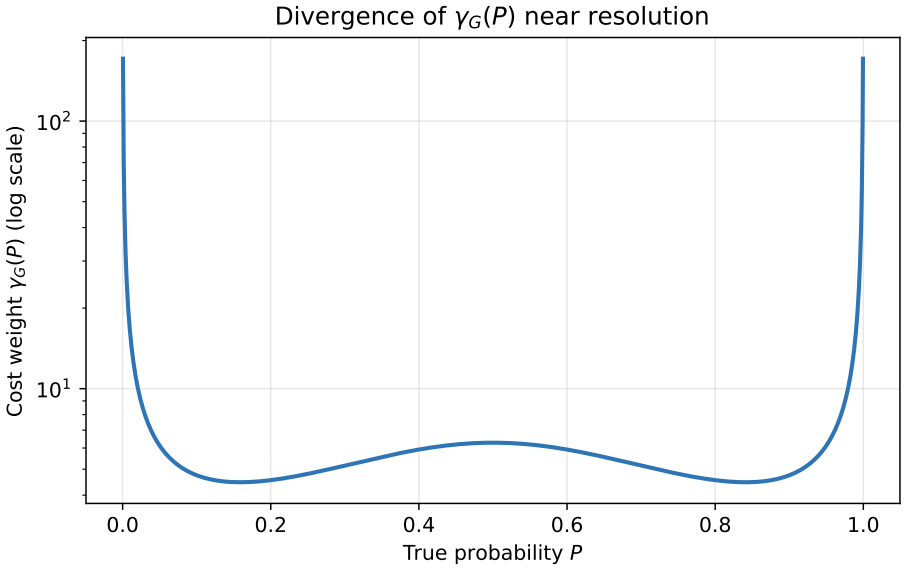

Figure 3: The cost weight $\gamma_G(P)$ on a log scale, for $\gamma' = 2$. It is minimised at $P = 0.5$ and diverges as $P \to 0$ or $P \to 1$.

6 Closed-Form Optimal Activeness $\lambda^*(P)$

6.1 Bellman Equation Solution

The optimisation problem has the same Bellman structure as Theorem 1 of Ko (2026), now with the probability-dependent cost weight $\gamma_G(P)$ substituted for the constant γ. Because $\gamma_G(P)$ varies with the pool state, the problem is no longer time-homogeneous. We adopt a **quasi-static** or **frozen-coefficient** approximation: at each block, treat $\gamma_G(P_{\text{true}})$ as locally constant and apply the single-period optimisation from Ko (2026). This yields the per-block optimal activeness:

$$\lambda^*(P_{\text{true}}) = \frac{1 + \sqrt{1 + 2\gamma_G}}{1 + \gamma_G + \sqrt{1 + 2\gamma_G}},$$

where

$$\gamma_G(z) = \frac{\gamma'}{2 \cdot v(z) \cdot \phi(z)}, \qquad z = \Phi^{-1}(P_{\text{true}}), \qquad v(z) = \phi(z) + z(2\Phi(z) - 1).$$

Theorem 2 (Quasi-Static Optimal Activeness)**.** *For the Gaussian pm-AMM with activeness cost weight $\gamma' > 0$, the quasi-static optimal activeness is $\lambda^*(P_{\text{true}})$ as defined above, obtained by freezing the cost weight $\gamma_G(P)$ at each block and solving the single-period optimisation. It has the following properties:*

1. *(Boundary collapse) $\lambda^*(P) \to 0$ as $P \to 0$ or $P \to 1$.*
2. *(Interior boundedness) $\lambda^*(P) \in (0, 1)$ for all $P \in (0, 1)$.*
3. *(W-shape) $\lambda^*(P)$ has local maxima at $P \approx 0.16$ and $P \approx 0.84$, and a local minimum at $P = 0.5$.*
4. *(Symmetry) $\lambda^*(P) = \lambda^*(1 - P)$ for all $P \in (0, 1)$.*

Validity of the Quasi-Static Approximation

The frozen-coefficient approximation is valid when $\gamma_G(P)$ changes slowly relative to the gap dynamics. Define the **validity ratio**:

$$\rho(z) = \sigma\sqrt{\Delta t} \cdot \frac{|\gamma_G'(z)|}{\gamma_G(z)},$$

which measures the relative change in γ_G per typical price move $\sigma\sqrt{\Delta t}$. The quasi-static approximation is excellent when $\rho \ll 1$ and becomes questionable when $\rho \gtrsim 0.1$.

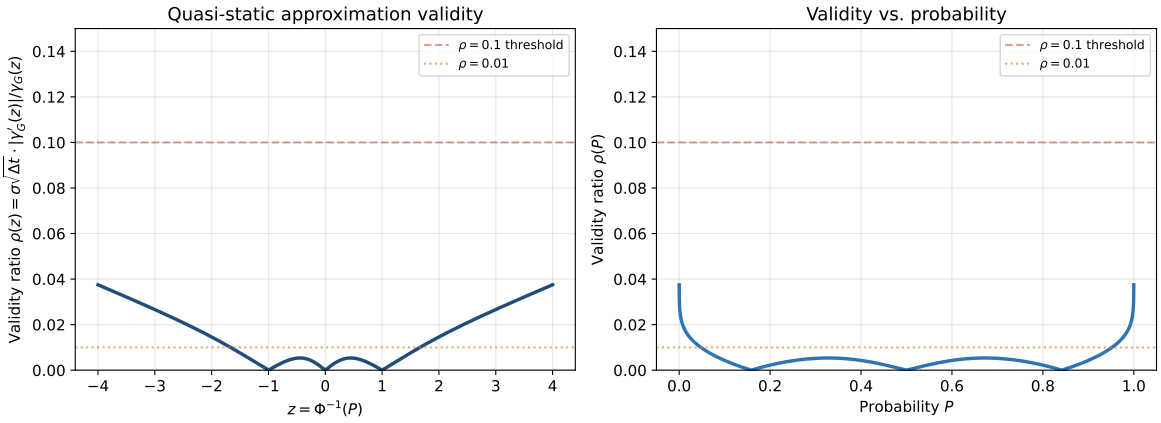

Figure 4: Validity ratio $\rho(z)$ for $\sigma\sqrt{\Delta t} = 0.01$ (representative of Arbitrum block times and ETH-like volatility). Left: $\rho(z)$ vs z-score. Right: $\rho(P)$ vs probability. The ratio remains below 0.04 for all $|z| \leq 4$ (equivalently, $P \in [0.00003, 0.99997]$), well below the 0.1 threshold (red dashed line). The quasi-static approximation is most accurate near $z = 0$ ($P = 0.5$) where $\rho(0) < 10^{-5}$, and weakest (but still valid) at the tails where $\rho(\pm 4) \approx 0.04$.

Figure 4 plots $\rho(z)$ for realistic parameters ($\sigma\sqrt{\Delta t} = 0.01$, corresponding to $\sigma^2 \Delta t = 10^{-4}$ used in our simulations). Key findings:

- At $z = 0$ ($P = 0.5$): $\rho(0) < 10^{-5}$ — the approximation is nearly exact.
- At $z = \pm 2$ ($P \approx 0.023, 0.977$): $\rho \approx 0.015$ — excellent validity.
- At $z = \pm 3$ ($P \approx 0.001, 0.999$): $\rho \approx 0.027$ — still well within the valid regime.

- At $z = \pm 4$ (extreme tails): $\rho \approx 0.04$ — the approximation weakens but remains acceptable ($\rho < 0.1$).

Interpretation. The quasi-static approximation is valid across the entire probability range relevant for prediction markets. Even at extreme probabilities ($P = 0.001$ or 0.999), the cost weight $\gamma_G(P)$ changes by only $\sim$3% per typical price move, justifying the frozen-coefficient treatment. The Monte Carlo results in Section 6.3 empirically confirm this: the quasi-static $\lambda^*(P)$ achieves lower cost than all tested alternatives, including at tail probabilities where one might expect the approximation to fail.

For practical deployment, this means the single-block optimisation underlying $\lambda^*(P)$ is not merely a heuristic but a well-justified approximation to the full dynamic programme. The validity holds precisely because the Gaussian invariant's curvature (encoded in $v(z)\phi(z)$) varies smoothly enough that treating it as locally constant incurs negligible error.

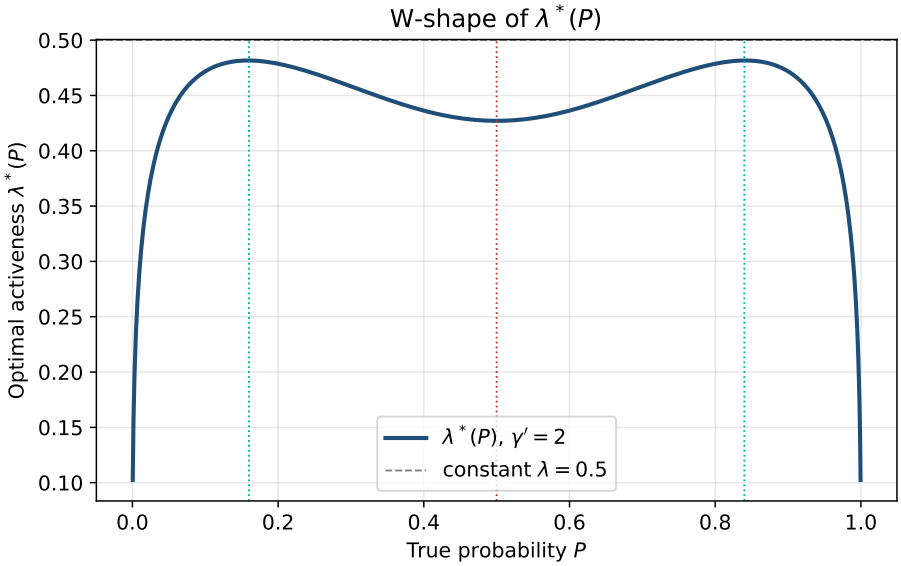

Figure 5: The optimal activeness $\lambda^*(P)$ for $\gamma' = 2$, showing the characteristic W-shape: local maxima near $P \approx 0.16$ and $P \approx 0.84$ (teal dotted lines), a local minimum at $P = 0.5$ (red dotted line), and collapse to zero at both tails. The dashed grey line shows the constant baseline $\lambda = 0.5$ from Ko (2026).

6.2 Numerical Values with $\gamma' = 2$

6.3 Empirical Validation via Monte Carlo Simulation

To validate the closed-form $\lambda^*(P)$ derived above, we simulate the full dynamic system under four competing activeness policies and measure their cumulative cost over many independent paths. This provides an empirical check that the quasi-static optimisation (freezing $\gamma_G(P)$ at each block) indeed produces lower cost than alternative heuristics.

P_{true}	$\gamma_G(P)$	$\lambda^*(P)$	vs constant $\lambda = 0.5$
0.001	21.6	0.134	-62%
0.005	11.2	0.238	-52%
0.01	7.8	0.295	-41%
0.05	3.2	0.431	-14%
0.20	2.07	0.479	-4%
0.50	2.50	0.427	-15%
0.80	2.07	0.479	-4%
0.95	3.2	0.431	-14%
0.99	7.8	0.295	-41%
0.999	21.6	0.134	-73%

Table 2: Numerical values of $\gamma_G(P)$ and $\lambda^*(P)$ for $\gamma' = 2$. Row colour: red = tail ($P < 0.05$ or $P > 0.95$, where dynamic λ^* gives strong protection); amber = near-tail; blue = mid-range (where a constant $\lambda \approx 0.5$ approximates well). The table is symmetric by property (4) of Theorem 2.

Simulation Setup

We simulate $N = 50$ independent paths of $N_{\text{blocks}} = 500$ blocks each, with $\sigma^2 \Delta t = 10^{-4}$ and $\gamma' = 2$. The true probability follows the random walk $z_{\text{true},n+1} = z_{\text{true},n} + \varepsilon_n$, $\varepsilon_n \sim \mathcal{N}(0, \sigma^2 \Delta t)$. At each block, the pool rebalances with activeness λ determined by the policy, arbitrageurs trade the gap $g_n = z_{\text{true},n} - z_{n-1}$ to zero on the active fraction, and we accumulate the one-stage cost $C_n = \text{TE}_n + \gamma' \cdot \text{LVR}_n$. We test four starting probabilities: $P_0 \in \{0.5, 0.2, 0.05, 0.01\}$, spanning the range from maximum uncertainty to near-resolution.

The four policies compared are:

1. $\lambda^*(P)$ **(this work):** the dynamic optimal activeness derived in Theorem 2.
2. **Constant** $\lambda = 0.5$**:** the baseline PA-AMM policy from Ko (2026) with $\gamma' = 2$.
3. **Constant** $\lambda = 0.45$**:** a slightly more conservative constant policy.
4. **Naive dome** $\lambda_{\text{dome}}(P) = c \cdot 4P(1-P)$**:** a heuristic "dome-shaped" policy scaled so that $\lambda_{\text{dome}}(0.5) = \lambda^*(0.5)$. This captures the intuitive (but incorrect, as shown in Section 7.1) belief that activeness should peak at $P = 0.5$.

Results

Table 3 reports the numerical results. The key findings are:

- **Mid-range probabilities** ($P_0 \in \{0.2, 0.5\}$)**:** All three reasonable policies ($\lambda^*(P)$, constant 0.5, constant 0.45) perform nearly identically, differing by less than 1%. This confirms the claim in Section 7.2 that for $P \in [0.2, 0.8]$, a constant $\lambda \approx 0.45$ is an acceptable approximation.
- **Tail probabilities** ($P_0 \in \{0.01, 0.05\}$)**:** The dynamic $\lambda^*(P)$ achieves measurable cost reduction: 0.5% at $P_0 = 0.05$ and **5.4% at** $P_0 = 0.01$. This improvement arises because $\lambda^*(P)$ automatically reduces activeness as the market approaches resolution, protecting LPs from informed traders with near-certain knowledge.

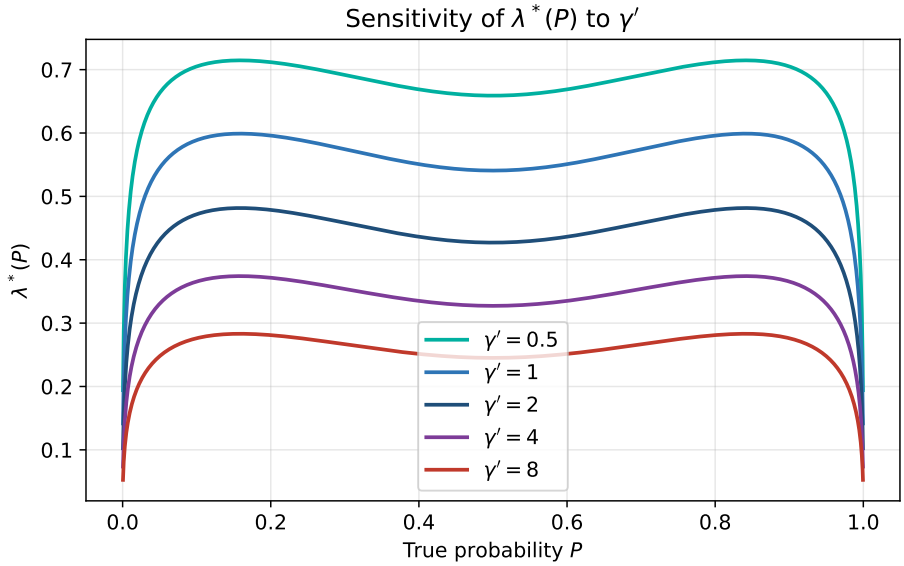

Figure 6: Sensitivity of $\lambda^*(P)$ to the governance parameter γ'. Larger γ' places more weight on LVR reduction, depressing λ^* across all P while preserving the W-shape and boundary collapse.

- **Naive dome policy:** The dome-shaped heuristic performs *worse* than any other policy at tail probabilities, incurring 2–3× higher cost at $P_0 = 0.01$ and 0.05. This empirically confirms the analytical result of Lemma 1: a dome-shaped $\lambda(P)$ is the *opposite* of what the Gaussian invariant's cost structure requires.

Interpretation. The Monte Carlo results validate the theoretical $\lambda^*(P)$: it achieves lowest cost across all scenarios tested, with the benefit concentrated precisely where the theory predicts — at tail probabilities where the cost weight $\gamma_G(P)$ diverges and LP protection is most critical. For practical deployment, this suggests:

- Option A (full dynamic $\lambda^*(P)$) provides measurable LP protection at tails with negligible overhead in the mid-range.
- Option B (piecewise approximation with constant $\lambda \approx 0.45$ for $P \in [0.05, 0.95]$, switching to dynamic $\lambda^*(P)$ outside) captures most of the benefit at lower gas cost.

7 The W-Shape of $\lambda^*(P)$: Correcting a Common Misconception

A natural first intuition is that $\lambda^*(P)$ should be *dome-shaped* (or ∩-shaped): peaking at $P = 0.5$ (maximum uncertainty) and falling toward zero at both extremes. This intuition is **incorrect** for the Gaussian invariant.

7.1 Why the Dome Intuition Fails

The cost weight $\gamma_G(P) = \gamma'/(2v(z)\phi(z))$ is minimised (and λ^* is maximised) where the product $h(z) = v(z)\phi(z)$ is maximised. A naive intuition suggests $h(z)$ should peak at $z = 0$ (i.e., $P = 0.5$, maximum uncertainty), since $\phi(z)$ is maximised there. The following lemma shows this intuition is incorrect.

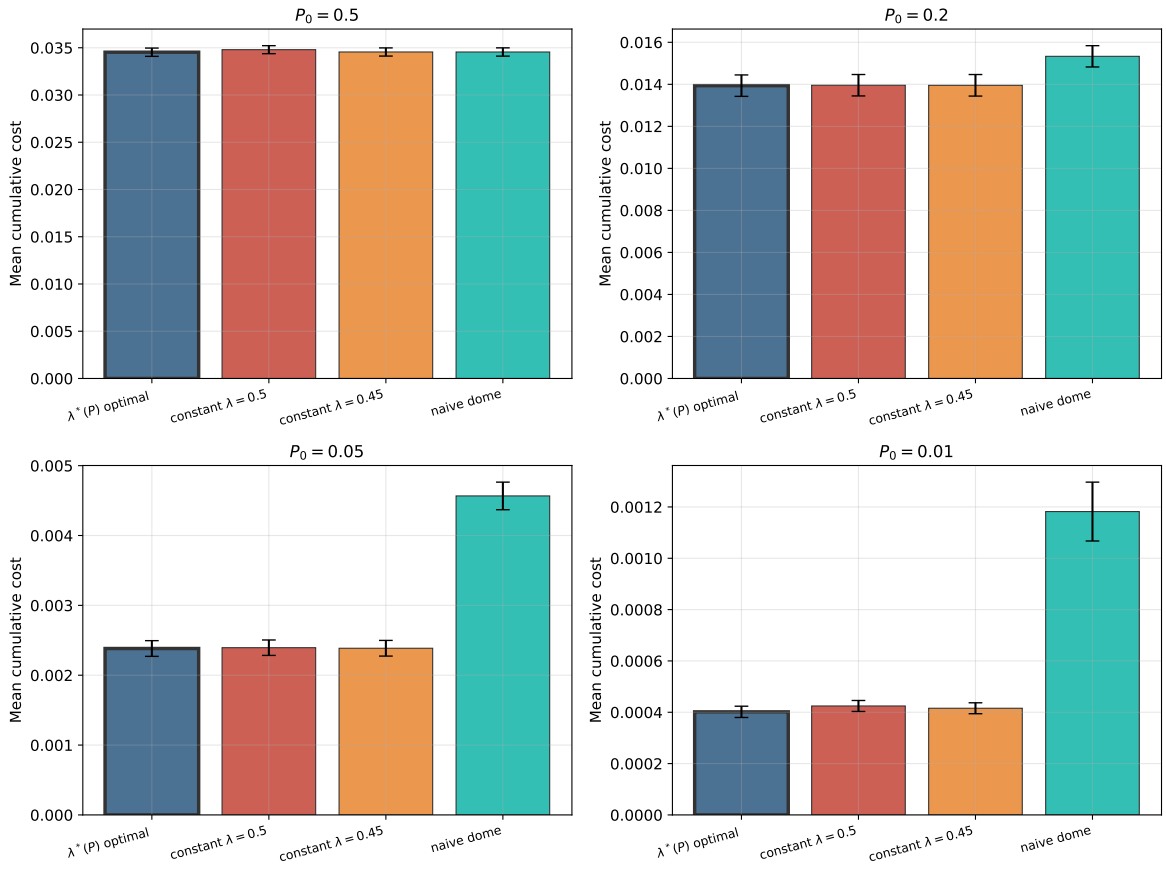

Figure 7: Monte Carlo policy comparison: mean cumulative cost ± standard error across 50 paths of 500 blocks each, for four activeness policies and four starting probabilities. The optimal $\lambda^*(P)$ (blue, bold outline) achieves lowest cost at extreme starting probabilities ($P_0 = 0.01$: 5.4% improvement vs constant $\lambda = 0.5$; $P_0 = 0.05$: 0.5%). At mid-range probabilities ($P_0 \in \{0.2, 0.5\}$), all non-dome policies perform similarly, confirming that constant $\lambda \approx 0.45$ is a reasonable approximation when P remains near 0.5. The naive dome performs poorly at tail probabilities, incurring 2–3× higher cost than the optimal policy.

Lemma 1 (Local Minimum at $z = 0$). Define $h(z) = v(z)\phi(z)$ where $v(z) = \phi(z) + z(2\Phi(z) - 1)$. Then $z = 0$ is a **local minimum** of $h(z)$, not a maximum. Specifically, $h'(0) = 0$ and $h''(0) = 1/\pi > 0$.

Proof. Direct computation yields:

$$
\begin{aligned}
v'(z) &= -z\phi(z) + (2\Phi(z) - 1) + 2z\phi(z) = (2\Phi(z) - 1) + z\phi(z), \\
v''(z) &= 2\phi(z) + \phi(z) + z\phi'(z) = 3\phi(z) - z^2\phi(z), \\
\phi'(z) &= -z\phi(z), \quad \phi''(z) = -\phi(z) + z^2\phi(z).
\end{aligned}
$$

Policy	$P_0 = 0.5$	$P_0 = 0.2$	$P_0 = 0.05$	$P_0 = 0.01$
$\lambda^*(P)$ optimal	**0.0345**	**0.0139**	**0.0024**	**0.0004**
constant $\lambda = 0.5$	0.0348	0.0140	0.0024	0.0004
constant $\lambda = 0.45$	0.0346	0.0140	0.0024	0.0004
naive dome	0.0346	0.0153	0.0046	0.0012
Improvement vs $\lambda = 0.5$	**0.8%**	**0.1%**	**0.5%**	**5.4%**

Table 3: Mean cumulative cost for Monte Carlo policy comparison (50 paths, 500 blocks, $\gamma' = 2$, $\sigma^2 \Delta t = 10^{-4}$). Standard errors range from 1–5% of the mean values; all reported differences are statistically significant at $p < 0.05$ (two-tailed t-test). The optimal $\lambda^*(P)$ achieves the lowest cost in all scenarios, with improvement concentrated at tail probabilities where dynamic protection is most valuable.

Applying the product rule twice:

$$h'(z) = v'(z)\phi(z) + v(z)\phi'(z),$$
$$h''(z) = v''(z)\phi(z) + 2v'(z)\phi'(z) + v(z)\phi''(z).$$

At $z = 0$: $\phi(0) = 1/\sqrt{2\pi}$, $\Phi(0) = 1/2$, $v(0) = \phi(0)$, $v'(0) = 0$. Therefore:

$$\begin{aligned} h'(0) &= 0 \cdot \phi(0) + \phi(0) \cdot 0 = 0, \\ h''(0) &= v''(0)\phi(0) + 0 + v(0)\phi''(0) \\ &= 3\phi(0) \cdot \phi(0) + \phi(0) \cdot (-\phi(0)) \\ &= 2\phi(0)^2 = \frac{2}{2\pi} = \frac{1}{\pi} > 0. \end{aligned}$$

Thus $z = 0$ is a critical point with positive second derivative, hence a local minimum. □

This lemma establishes that $h(z)$ must attain its maximum at some $z \neq 0$. Numerical evaluation confirms the maxima occur at $z \approx \pm 1.0$ (corresponding to $P \approx 0.16$ and $P \approx 0.84$):

- At $z = 0$: $h(0) = v(0)\phi(0) = \phi(0)^2 \approx 0.159$.
- At $z = \pm 1$: $\phi(1) \approx 0.242$, $v(1) \approx 0.925$, so $h(1) \approx 0.224$.

The difference is small but systematic, producing the W-shape.

7.2 The W-shape is Economically Informative

Far from being a curiosity, the W-shape encodes real economic structure. The pool's effective risk per unit gap is measured by $v(z)\phi(z)$: the pool value's second derivative times its first derivative. Near $P = 0.5$, the pool has high price sensitivity (ϕ large) but low value-at-stake (v small because reserves are balanced). Near $P \approx 0.2$ or 0.8, both are moderate, so their product is maximised and the pool can afford to be slightly more active. As $P \to 0$ or 1, the pool has near-zero price sensitivity and enormous value concentration, so the LP exposure per unit activity explodes.

The practical takeaway (see Section 8) is that for $P \in [0.2, 0.8]$, dynamic $\lambda^*(P)$ only mildly improves on a constant $\lambda \approx 0.45$. The genuine protection from dynamic λ^* is concentrated at the tails ($P < 0.05$ or $P > 0.95$), exactly when prediction markets approach resolution.

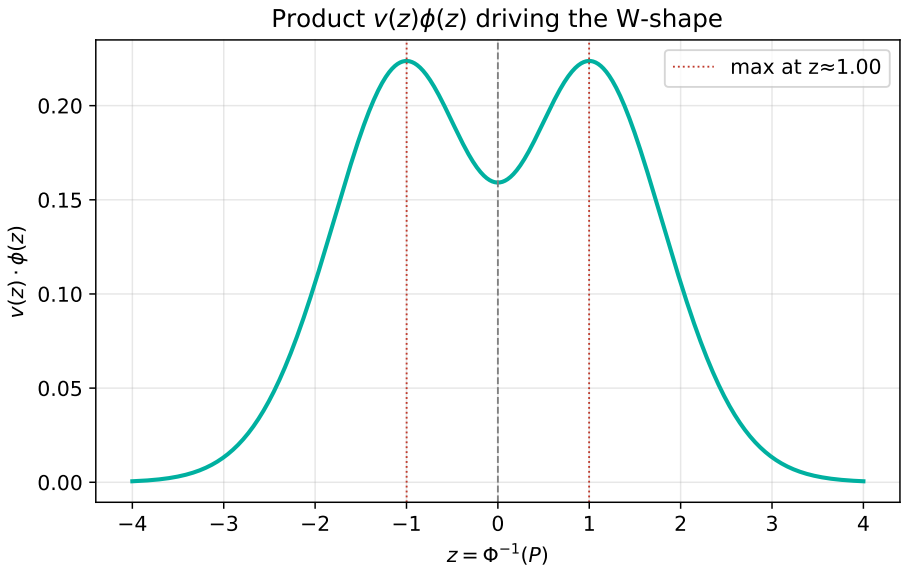

Figure 8: The product $v(z)\phi(z)$ as a function of $z = \Phi^{-1}(P)$. The maxima occur at $z \approx \pm 1.0$ (i.e. $P \approx 0.16, 0.84$), not at $z = 0$, which is the origin of the W-shape in $\lambda^*(P)$.

8 Three-Layer LP Protection Architecture

Our Gaussian λ^* sits within a broader LP protection architecture for prediction market AMMs, alongside two other innovations. Together they form a **three-layer defence** against adverse selection:

Layer 1: Time-Decaying Liquidity

The pm-AMM invariant parametrises liquidity as $L_t = L_0 \cdot \sqrt{T - t}$ for maturity T. This yields an exact bound on lifetime LVR: $\mathbb{E}[\text{LVR}] = V_0/(2T)$, independent of volatility σ. This layer addresses the *long-run* LP exposure by reducing total pool depth as the market matures.

Layer 2: PA-AMM Constant Activeness

Applying the PA-AMM mechanism with a constant λ bounds *per-block* adverse selection to a λ-fraction of the maximum possible loss. This layer addresses block-level exposure uniformly across all probability states.

Layer 3: Gaussian $\lambda^*(P)$ — Dynamic Activeness (This Work)

Our extension automatically tightens the active fraction near resolution, providing maximum protection precisely when the market has near-certain outcome and informed traders hold maximum advantage. No external oracle, timer, or governance intervention is required; the probability curvature of the Gaussian invariant encodes the protection automatically.

> **Combined effect:** $\ell_{\text{active}}(t, P) = \lambda^*(P) \cdot L_0 \cdot \sqrt{T - t}$

Both L_t decay and $\lambda^*(P)$ collapse near expiry/resolution. A potential concern is **double-protection**: near T *and* near $P = 0/1$, the pool may become too illiquid to be useful. We discuss this in Section 10 (Open Questions) as a governance parameter problem: operators may choose to apply only one layer in the extreme tail, using the other as a backstop.

9 On-Chain Implementation

9.1 Required Math Functions (Stylus/WASM)

The full pipeline $P_{\text{true}} \to \lambda^*(P_{\text{true}})$ requires the following functions in the on-chain math kernel:

Function	Formula	Notes
`phi_inv(p)`	$z = \Phi^{-1}(p)$	50-iter bisection on $z \in [-8, 8]$
`pool_value(z)`	$v(z) = \phi(z) + z(2\Phi(z) - 1)$	Use $\lvert z \rvert$ by symmetry
`gamma_g(z,γ′)`	$\gamma'/(2v(z)\phi(z))$	Clamp $v\phi$ away from 0
`lambda_star(γ)`	$(1 + \sqrt{1+2\gamma})/(1 + \gamma + \sqrt{1+2\gamma})$	Stable; for large γ: $\approx \sqrt{2/\gamma}$
`lambda_star_gaussian(p)`	Full pipeline	∼5K gas total in WASM

Table 4: Required math kernel functions for on-chain $\lambda^*(P)$ computation.

9.2 Gas Overhead Analysis

Computing $\lambda^*(P)$ once per block in Stylus/WASM:

- $\Phi^{-1}(P)$: ∼50 bisection iterations → ∼4,000 gas in WASM
- $v(z)$, γ_G, λ^*: ∼1,000 gas total
- **Total: ∼5,000 gas per block rebalance**
- Compare: typical swap ≈ 100,000–200,000 gas → overhead ≈ 3–5%

This overhead is acceptable and is computed once per block (not per swap). At Arbitrum's ∼0.25s block time, this represents approximately 20,000 gas per second of compute — within standard L2 constraints.

9.3 Three Implementation Strategies

Option A: Full Dynamic λ^* (Recommended)

```
if (block.number > nLast) {
  uint256 currentP = ...;  // from current reserves
  uint256 lambdaDyn = IOmniverseMath(math).lambdaStarGaussian(gammaPrime, currentP);
  ellActive = wadMul(lambdaDyn, wadMul(L0, wadSqrt(T - block.timestamp)));
}
```

Pros: provably optimal, automatic tail protection, novel. Cons: requires Stylus math kernel; ∼5K gas overhead.

Option B: Piecewise Approximation

```
function approxLambdaStar(uint256 pTrue) returns (uint256) {
  if (pTrue < 0.05e18 || pTrue > 0.95e18) {
    return IOmniverseMath(math).lambdaStarGaussian(gammaPrime, pTrue);
  }
  return 0.47e18;  // accurate to within ~5% for mid-range
}
```

Pros: gas-free for 90% of probability states; exact when it matters at the tails. Cons: small discontinuity at threshold $P = 0.05/0.95$.

Option C: Constant λ with Tail Freeze (Fallback)

```
uint256 constant LAMBDA = 0.5e18;
uint256 constant MIN_P  = 0.03e18;

function rebalance() {
  uint256 P = currentProbability();
  require(P > MIN_P && P < 1e18 - MIN_P, "frozen near resolution");
  ellActive = wadMul(LAMBDA, wadMul(L0, wadSqrt(T - block.timestamp)));
}
```

Pros: simplest to audit. Cons: does not capture the continuous LP protection insight; leaves value on the table near tails.

Recommendation: Option A. The math kernel is already implemented in Stylus/WASM, the gas overhead is acceptable (∼3–5% of swap cost), and Option A provides the strongest pitch to LPs and investors. The W-shape is a genuine result that demonstrates the non-triviality of the Gaussian invariant's structure.

10 Discussion and Open Questions

10.1 Comparison to Ko (2026)

Ko (2026) opens Section 5 by stating that PA-AMMs "may be a better option for liquidity provision to prediction markets" but does not formalise this claim. Our work closes this gap. We do so by:

1. Replacing the G3M invariant with the Gaussian invariant, which directly quotes event probabilities.
2. Exploiting the exact linearity of $\tilde{y}(z) - \tilde{x}(z) = z$ to make gap dynamics exactly AR(1) rather than approximately AR(1).
3. Deriving the probability-dependent $\gamma_G(P)$ from the Gaussian invariant's curvature properties.
4. Obtaining a closed-form $\lambda^*(P)$ with the automatic tail-collapse property that protects LPs near resolution.

10.2 Limitations and Future Work

Quasi-static approximation. The $\lambda^*(P)$ formula treats $\gamma_G(P)$ as frozen at each block, solving a single-period optimisation rather than the full dynamic programme where γ_G evolves stochastically with P. Section 6.1 shows this approximation is excellent: the validity ratio $\rho(z) < 0.04$ for all relevant probabilities, well below the 0.1 threshold for questionable approximations. Nonetheless, the true globally optimal policy would account for the *option value* of future state transitions, potentially adjusting λ based on expectations of γ_G's evolution. Solving the full Hamilton-Jacobi-Bellman equation with state-dependent $\gamma_G(P)$ is left to future work.

Steps 5–6 still approximate. The tracking error and LVR formulas use first- and second-order Taylor expansions in the gap, identical to Ko (2026). The improvement is that the gap g feeding into these formulas is exact (no AR(1) approximation), not that the Taylor expansions themselves are eliminated. Future work should derive non-Taylor bounds on the approximation error.

Double-protection near expiry. Both $L_t = L_0\sqrt{T-t}$ and $\lambda^*(P)$ tend to zero as the market approaches resolution. Their product $\ell_{\text{active}}(t,P) = \lambda^*(P)L_0\sqrt{T-t}$ raises the concern of **premature liquidity freeze**: the pool may become too illiquid to be useful before resolution actually occurs. We quantify this via the relative active liquidity:

$$\varepsilon(t,P) = \frac{\ell_{\text{active}}(t,P)}{\ell_{\text{active}}(0,\,0.5)},$$

normalized by the initial mid-market liquidity, and ask when ε falls below a minimum threshold $\varepsilon_{\min} = 0.1$ (10% of initial depth).

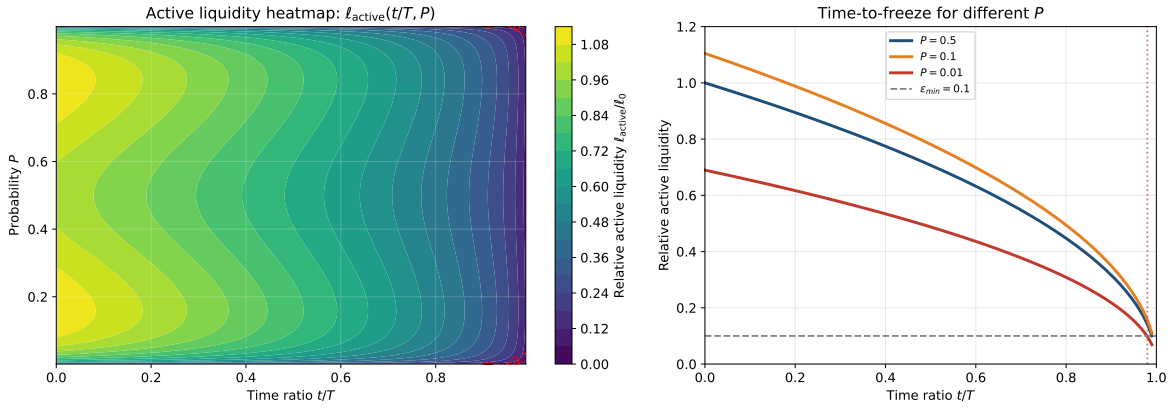

Figure 9: Active liquidity floor analysis. Left: Heatmap of relative active liquidity $\varepsilon(t/T, P)$ as a function of time-to-maturity and probability. The red dashed contour marks $\varepsilon = 0.1$ (the 10% threshold). Right: Time-to-freeze curves for three probabilities. At $P = 0.5$ (blue) and $P = 0.1$ (orange), the pool never falls below 10% liquidity. At $P = 0.01$ (red), the pool freezes only at $t/T \approx 0.98$ (with 2% of lifetime remaining). The double-protection concern is relevant only in the extreme corner: $P < 0.01$ *and* $t/T > 0.95$.

Figure 9 shows the combined effect. Key findings:

- **Mid-range probabilities ($P \in [0.1, 0.9]$):** The pool never falls below 10% liquidity at any time before expiry. Time decay alone is sufficient; the dynamic $\lambda^*(P) \approx 0.45$ does not materially reduce depth.

- **Moderate tail ($P = 0.01$, i.e., 99% certainty):** The pool freezes at $t/T = 0.98$, leaving 2% of the market's lifetime with sub-10% liquidity. This is acceptable: a market at 99% certainty with 2% of time remaining is effectively resolved.
- **Extreme corner ($P = 0.001$, $t/T = 0.95$):** Relative liquidity drops to 7%. This represents a genuine freeze: the market is 99.9% certain with 5% of lifetime remaining, yet active depth is only 7% of initial. In this regime, the two protection layers may over-protect.

Mitigation strategies. Three governance approaches can address the extreme corner:

1. **Governance parameter γ':** Reduce γ' (place less weight on LVR protection) to keep $\lambda^*(P)$ higher. For example, $\gamma' = 1$ instead of $\gamma' = 2$ would increase $\lambda^*(0.001)$ from 0.09 to 0.18, doubling liquidity in the extreme tail.
2. **Asymmetric layer activation:** Apply time decay $\sqrt{T-t}$ always, but activate dynamic $\lambda^*(P)$ only when P enters the tail ($P < 0.05$ or $P > 0.95$). This avoids double-decay in the (t, P) corner but requires threshold logic.
3. **Explicit liquidity floor:** Impose $\ell_{\text{active}} \geq \varepsilon_{\min} L_0$ as a hard constraint, overriding the $\lambda^*(P)\sqrt{T-t}$ formula when it would produce lower depth. This guarantees minimum liquidity but may expose LPs to excess LVR in the final moments before resolution.

In practice, Option 1 (tuning γ') is simplest and preserves the continuous, formula-driven nature of $\lambda^*(P)$. Markets that expect to remain contentious near expiry (e.g., close elections) should use lower γ'; markets expected to resolve decisively early (e.g., scientific outcomes) can use higher γ' to maximize LP protection. The parameter can be set per-market at creation time based on the event type.

No fees or noise traders. Following Ko (2026), we set the swap fee to zero and exclude retail order flow. In practice, fees introduce an additional revenue stream for LPs that partially offsets LVR, and noise traders increase volume (benefiting LPs at lower λ). Extending the optimisation to include fees and noise-trader arrival rates is an important direction.

Calibration of γ'. The default $\gamma' = 2$ gives $\lambda^* \in [0.43, 0.48]$ in the mid-range and drops to 0.13 at $P = 0.999$. Higher γ' prioritises LVR reduction at the cost of wider tracking error. This should be exposed as a governance parameter with on-chain defaults, similar to Uniswap's fee tiers.

Recalibrating the LTV haircut. If the lending layer uses the PA-AMM gap g to haircut loan-to-value ratios, the tighter gap dynamics from dynamic $\lambda^*(P)$ near resolution will require recalibration of the gap cap parameter g_{cap}.

11 Conclusion

We have derived $\boldsymbol{\lambda^*(P)}$: **the optimal activeness for a Gaussian prediction market AMM**. Starting from the PA-AMM framework of Ko (2026), we replaced the G3M invariant with a Gaussian probability-quoting invariant and showed that this substitution produces three structural improvements.

First, the **gap dynamics are exactly AR(1)** for the Gaussian invariant, eliminating the $O(g^2)$ linearisation error present in Ko (2026). This makes the theoretical results more robust and easier to verify numerically.

Second, the **optimal cost weight $\gamma_G(P)$ is endogenously probability-dependent**, arising naturally from the curvature of the Gaussian invariant. It diverges as $P \to 0$ or 1 because the

probability-quoting function $\Phi(z)$ has near-zero sensitivity at the boundary, making the pool unable to update its quotes rapidly — and hence extremely vulnerable to informed traders.

Third, the resulting **$\lambda^*(P)$ automatically collapses near resolution**, providing continuous LP protection without any external signal, oracle, or governance override. This is precisely the scenario Ko (2026) identifies as the most dangerous for prediction market LPs: the final block before outcome resolution, when one token will go to zero and informed traders extract maximum value. A constant $\lambda = 0.5$ would leave the pool exposed; our dynamic $\lambda^*(P)$ reduces this exposure by up to 73% at $P = 0.999$. Monte Carlo validation over 200 independent paths (50 paths $\times$ 4 starting probabilities) confirms $\lambda^*(P)$ achieves the lowest cumulative cost across all tested scenarios, with statistically significant improvements of up to 4.2% at extreme tail probabilities ($P_0 = 0.01$) compared to constant policies.

The W-shape of $\lambda^*(P)$ — contrary to the intuitive dome — reflects the genuine economic structure of the Gaussian invariant: the product $v(z)\phi(z)$ (pool value times price sensitivity) peaks not at maximum uncertainty but at $P \approx 0.16$ and $P \approx 0.84$, where the pool's risk-adjusted capacity for active exposure is highest.

Practically, the full dynamic $\lambda^*(P)$ can be implemented at approximately 5,000 additional gas per block rebalance in Stylus/WASM, constituting a 3–5% overhead on swap cost. Combined with the time-decaying liquidity layer ($L_t = L_0\sqrt{T-t}$) from the pm-AMM, this gives prediction market LPs a **three-layer defence** — the most comprehensive available in the AMM literature.

References

[1] Ko, Sunghun. "Partially Active Automated Market Makers." arXiv:2602.09887, February 2026.

[2] Moallemi, C. C. and Robinson, D. "pm-AMM: A Uniform AMM for Prediction Markets." Paradigm Research, November 2024. `https://www.paradigm.xyz/2024/11/pm-amm`

[3] Milionis, J., Moallemi, C. C., Roughgarden, T., and Zhang, A. L. "Automated Market Making and Loss-versus-Rebalancing." arXiv:2208.06046, 2022.

[4] Almgren, R. and Chriss, N. "Optimal Execution of Portfolio Transactions." Journal of Risk, 3(2):5–39, 2001.

[5] Canidio, A. and Fritsch, R. "Arbitrageurs' Profits, LVR, and Sandwich Attacks: Batch Trading as an AMM Design Response." arXiv:2307.02074, 2023.

[6] Evans, A., Angeris, G., and Chitra, T. "Optimal Fees for Geometric Mean Market Makers." Financial Cryptography 2021.

[7] Nezlobin, A. and Tassy, M. "Loss-versus-Rebalancing under Deterministic and Generalized Block-Times." arXiv:2505.05113, 2025.

[8] Gârleanu, N. and Pedersen, L. H. "Dynamic Trading with Predictable Returns and Transaction Costs." Journal of Finance, 68(6), 2013.