

Real-Time Probabilistic Pricing of Short-Horizon on-Chain Range-Prediction Markets

Junyop Choe¹, Indran Xavier^{2*}

¹Institute of Mathematics, State Academy of Sciences, Pyongyang, Democratic People's Republic of Korea

²Applied Security Technology (AST), Subang Jaya, Malaysia

*Corresponding Author

DOI: <https://doi.org/10.51584/IJRIAS.2026.11080157>

Received: 06 September 2026; Accepted: 11 September 2026; Published: 21 September 2026

ABSTRACT

Ultra-short-horizon prediction products—where a user wagers that an asset's price will lie inside a chosen price band over a window of a few seconds to a few minutes—have grown rapidly on public blockchains, mirroring the explosion of zero-days-to-expiry (0DTE) options in traditional venues. Unlike inventory-based prediction-market makers (e.g. LMSR) or classical option pricers, these products must quote a fair, stable, and exploit-resistant multiplier continuously (at ~ 1 Hz) directly from a live, sub-second price oracle, without a tradable hedge. We present TESSERA, a deterministic and auditable pricing engine for such grid range-prediction markets. The engine (i) estimates relative (scale-invariant) volatility from oracle ticks via a half-life-weighted EWMA with a shock-mode multiplier, (ii) forms a clipped, horizon-decayed drift, (iii) maps the resulting forward distribution to an in-range probability, and (iv) converts probability to a house-edged multiplier with clamps and state-dependent exponential smoothing. Crucially, a server-authoritative acceptance layer (synchronous re-pricing, quote-age and feed-stale kill switches, and a display=lock=settle invariant) neutralizes timing and latency-arbitrage exploits that are otherwise fatal at these horizons. On a Gaussian-process microbenchmark (40,000 tiles) the terminal-Gaussian variant attains an expected calibration error (ECE) of 0.88% at a realized house margin of 4.67% (target 5%), while a popular barrier $\times$ coverage heuristic miscalibrates severely (ECE 27.3%). We then stress the engine on a deliberately non-circular benchmark whose ticks follow a Heston stochastic-volatility process with Merton jumps—a law the Gaussian pricer cannot fit by construction: a naive fixed-volatility pricer degrades to 6.8% ECE while online EWMA estimation roughly halves it to 3.0%, and for path-dependent (stay-in-range) tiles a double-barrier first-passage model (validated to 1.7% mean error against fine-grid Monte-Carlo) beats a terminal-only approximation $3 \times$ under diffusion, with both degrading under jumps—isolating jump risk as the binding limitation. We prove that stale-quote leakage vanishes as acceptance latency $\lambda \rightarrow 0$ (bounded by $C\sqrt{\lambda}$) and confirm it in simulation ($0.78\% \rightarrow 0$). Finally we replay the engine on 40,000 tiles built from real 1-second BTC/USD ticks: deployed with the synthetic floor it fails (ECE 26%), but re-fitting a single volatility-floor parameter to the data cuts calibration error nearly $5 \times$ to 5.5%—quantifying both the engine's real-tick fidelity and the necessity of per-asset calibration. We release the model, calibration protocol, simulators, and replay data to make short-horizon range pricing reproducible.

Keywords: prediction markets; short-horizon options; realized volatility; barrier options; latency arbitrage.

INTRODUCTION

A grid range-prediction market discretizes the near future of an asset price into a two-dimensional lattice of tiles. Each tile is a rectangle $[L, U] \times [t_s, t_e]$: a price band $[L, U]$ and a time window $[t_s, t_e]$ that opens a few seconds ahead and lasts seconds to minutes. A player stakes on a tile and, if the underlying is inside the band at settlement, receives their stake multiplied by a quoted multiplier M ; otherwise the stake is lost. The platform

quotes every live tile continuously and settles on-chain against a decentralized price oracle. Economically each tile is a self-financing, ultra-short-dated range (double-barrier) contract with no continuously tradable hedge and no counterparty other than the house.

These products sit at the confluence of three trends. First, the migration of speculative flow to ultra-short horizons: 0DTE options now account for a majority of S&P 500 option volume, exhibiting extreme gamma and pin-risk dynamics that stress conventional pricing [6, 7]. Second, the maturation of pull-based decentralized oracles that stream sub-second, multi-publisher prices to smart contracts on demand, making on-chain settlement of second-scale contracts feasible [17]. Third, the rise of prediction and event markets: the on-chain venue Polymarket alone traded $\approx$ \$3.7 billion in November 2025, and combined prediction-market volume (on-chain Polymarket plus the regulated exchange Kalshi) set successive monthly records approaching \$10 billion in late 2025 and rose toward \$24 billion per month by April 2026 [4, 5].

Pricing these tiles is deceptively hard. Classical barrier/first-passage theory [12, 15] assumes a known diffusion and delivers elegant closed forms, but the volatility that dominates a 10-second horizon must itself be estimated live from noisy ticks, and it swings across regimes within a session. Inventory-based prediction-market makers such as LMSR [1, 2] price from bet flow and bounded loss, not from underlying dynamics, and provide no notion of a calibrated hit probability against a real price path. And unlike a quote on a slow order book, a range multiplier that is even a few hundred milliseconds stale is a free option for a latency-advantaged trader [20, 18].

Contributions

1. **A deterministic real-time pricing pipeline** for grid range-prediction markets that maps live oracle ticks to a fair, clamped, smoothed multiplier, with every intermediate quantity versioned and auditable (§4).
2. **A comparative probability model with a non-circular evaluation.** We formalize in-range estimators (terminal-Gaussian, midpoint, and barrier $\times$ coverage) and a validated double-barrier first-passage model, and evaluate them not only under Gaussian ticks but under a Heston stochastic-vol process with Merton jumps whose law the Gaussian pricer cannot fit by construction—separating genuine accuracy from tautology (§4.4, §7).
3. **An exploit-resistance layer with a proof.** We cast quote acceptance as a synchronous-clearing mechanism (server-authoritative re-pricing, quote-age and feed-stale kill switches, display=lock=settle invariant), connect it to frequent batch auctions [20], and prove that stale-quote leakage is bounded by $C\sqrt{\lambda}$ in acceptance latency λ , hence vanishes under synchronous re-pricing (§5, §7).
4. **A real-tick replay and reproducible artifacts** (§7.6, §6)—the engine replayed on 40,000 tiles of real 1-second BTC/USD data, showing a $5 \times$ calibration gap between off-the-shelf and per-asset-calibrated deployment—released together with the GBM/Heston+jumps simulators, the double-barrier validation, and the tick data, so every number is reproducible without proprietary inputs.

Related Work

Prediction-Market Pricing

The dominant academic lens is the automated market maker. LMSR provides always-on, bounded-loss quoting with axiomatic guarantees [1], and recent work generalizes liquidity provisioning to thin and adversarial markets [2] and pushes toward a “Black–Scholes for prediction markets” via unified kernels [3]. These mechanisms price the book; they are agnostic to the stochastic process of an external reference. Our setting is complementary: the payoff is defined by an exogenous price path, so pricing is a statistical estimation problem (what is the probability the path lands in the band?) fused with a house-edge mechanism, rather than an inventory-scoring problem.

Short-Dated Options and Realized Volatility

The ODTE literature documents that ultra-short horizons magnify gamma and make the volatility input the first-order driver of price [6, 7]. For the volatility input itself, exponentially weighted moving averages remain a practical workhorse [8, 9]; realized-volatility theory formalizes high-frequency estimation [10]; and recent crypto work prices options with volatility-of-volatility dynamics on high-frequency data [11]. We adopt a half-life EWMA on relative (log-return) volatility for scale invariance across price levels—a property the internal reviews of dollar-based variants flagged as essential.

Barrier / Range / First-Passage Pricing

A tile is a double-barrier range contract. Closed forms and approximations exist under diffusion and jump models [12, 15, 16], but they presuppose a fixed, known process. We use a reflection-style barrier approximation and a calibrated coverage term as one candidate estimator, and show that a direct terminal-distribution estimator is markedly better calibrated at these horizons when the volatility is estimated online.

Oracles, MEV and Latency Arbitrage

Pull-based oracles enable on-demand sub-second prices on chain [17]. But sub-second pricing invites sub-second exploitation: front-running and MEV on decentralized venues [18, 19], and the general latency-arbitrage critique of continuous markets, whose canonical remedy is discrete/synchronous clearing [20]. Our acceptance layer is a domain-specific instance of that remedy: quotes are made binding only at a synchronous, server-authoritative re-price.

Contract And System Model

The tile Contract

Fix a reference asset with mid price S_t (we use BTC/USD in experiments). A tile is $\tau = ([L, U], [t_s, t_e])$ with $L < U$ and $t_0 \leq t_s < t_e$, where t_0 is the quote time. Define

$$T_1 = \max(t_s - t_0, 0), \quad T_2 = \max(t_e - t_s, T_{\min}), \quad H = T_1 + T_2, \quad (1)$$

the time-to-start, active duration, and total horizon. The canonical payoff is terminal-in-range,

$$\text{payoff}(\tau) = M(\tau) \cdot \mathbf{1}\{S_{t_e} \in [L, U]\} \cdot (\text{stake}), \quad (2)$$

with quoted multiplier $M(\tau) \geq 1$. A touch/occupation variant, which pays if S enters $[L, U]$ at any time in $[t_s, t_e]$, is discussed in §8; our pricing and evaluation target (2).

Architecture and Trust

An off-chain quoting service consumes a pull-based oracle tick stream $\{(t_i, S_i)\}$ (multi-publisher, aggregated, sub-second) and publishes a multiplier for every live tile. The server is the source of truth: a bet is bound only when the server accepts it, at which point it re-prices against the freshest tick and persists (lockedMult, quoteVersion, serverTimestamp, input digest). Settlement occurs on an EVM-compatible rollup: the settlement contract reads the oracle price at t_e and pays lockedMult exactly. Determinism and full traceability are first-class requirements—identical tick stream and configuration must yield an identical quote, and every accepted bet embeds the formula version.

Threat Model

We consider a profit-seeking bettor who (a) may observe the public oracle stream with lower latency than the platform's quote refresh; (b) may time bet submission relative to tile activation; and (c) knows the pricing formula and configuration. The bettor cannot forge oracle prices (the oracle's multi-publisher aggregation is

trusted [17]) and cannot alter settlement. The concrete exploits are: stale-quote sniping (betting a tile whose quote has not yet absorbed a just-observed price move), lock-boundary timing, and smoothing exploitation (predicting the EMA's lag). We quantify the first in §7 and neutralize all three in §5.

Pricing Model

All quantities are computed per tick from the tick buffer. We write t_{now} for the current server time.

Return Basis and Relative Volatility

To be invariant to the price level, we use log returns normalized by the square-root of the inter-tick gap,

$$r_i = \frac{\ln(S_i/S_{i-1})}{\sqrt{\Delta t_i}}, \quad \Delta t_i = t_i - t_{i-1} \text{ (s)}. \quad (3)$$

Over a window VOL_WINDOW_S , pairs are weighted by an exponential half-life VOL_HALF_LIFE_S :

$$w_i = \exp\left(-\ln 2 \cdot \frac{t_{\text{now}} - t_i}{\text{VOL_HALF_LIFE_S}}\right), \sigma_{\text{rel}} = \max\left(\sqrt{\frac{\sum_i w_i r_i^2}{\sum_i w_i}}, \text{SIGMA_REL_FLOOR}\right). \quad (4)$$

The floor prevents divide-by-near-zero on quiet feeds. The dollar spread over a horizon h is $\sigma_{\$}(h) = S_0 \sigma_{\text{rel}} \sqrt{h}$, with $S_0 = S_{t_0}$.

Shock Mode

A sudden move should widen spreads immediately rather than wait for the EWMA to catch up. Let $z_{\text{last}} = |r_{\text{last}}|/\sigma_{\text{rel}}$. If $z_{\text{last}} \geq \text{SHOCK_Z}$ the tick enters shock mode and

$$\sigma_{\text{rel,eff}} = \sigma_{\text{rel}} \cdot \text{SHOCK_SIGMA_MULT}, \quad \text{else } \sigma_{\text{rel,eff}} = \sigma_{\text{rel}}. \quad (5)$$

Shock mode also selects a more aggressive smoothing weight (§4.6).

Clipped, Decayed Drift

A small momentum term corrects the forward mean without inviting trend overfitting. With linear recency weights $\tilde{w}_j = \max(0, 1 - (t_{\text{now}} - t_j)/\text{MOMENTUM_WINDOW_S})$,

$$m = \frac{\sum_j \tilde{w}_j (S_j - S_{j-1})/\Delta t_j}{\sum_j \tilde{w}_j}, \quad \mu = \text{clip}(m, -S_0 \kappa, S_0 \kappa), \quad \mu_{\text{eff}} = \mu e^{-k_{\mu} H}, \quad (6)$$

where $\kappa = \text{DRIFT_CLIP_BPS_PER_S}$ and the horizon decay k_{μ} suppresses drift for distant tiles. The predicted mid-window level is $\hat{S} = S_0 + \mu_{\text{eff}} (T_1 + T_2/2)$.

In-Range Probability: Terminal and Path-Dependent Estimators

Let Φ be the standard normal CDF. For the terminal payoff (2) we compare three estimators of $p = \mathbb{P}(S_{t_e} \in [L, U])$; a fourth, for the path-dependent variant, is given at the end.

(A) Terminal-Gaussian (Recommended)

Approximate S_{t_e} as Gaussian with mean $\hat{S}$ and standard deviation at the settlement horizon $\Sigma_H = S_0 \sigma_{\text{rel,eff}} \sqrt{H}$:

$$p_A = \Phi(U - \hat{S}/\Sigma_H) - \Phi(L - \hat{S}/\Sigma_H). \quad (7)$$

(B) Midpoint-Gaussian (As Shipped)

An early production build evaluated the spread at the window midpoint, $\Sigma_{mid} = S_0 \sigma_{rel,eff} \sqrt{\max(T_1 + T_2/2, 0.1)}$, using (7) with Σ_{mid} in place of Σ_H . Because $\Sigma_{mid} < \Sigma_H$, p_B is systematically over-confident.

(C) Barrier×Coverage Heuristic

A design-spec pipeline combines a reflection-style barrier term with a calibrated coverage factor. With nearest-edge distance $d = \max(L - \hat{S}, \hat{S} - U, 0)$, start spread $\Sigma_{T_1} = S_0 \sigma_{rel,eff} \sqrt{\max(T_1, T_{min})}$, and $x = \text{width}/(2\Sigma_{T_1})$:

$$p_c = 2 \underbrace{\Phi(-d/\Sigma_H)}_{\text{barrier}} \cdot \underbrace{(1 - e^{-ax^b})}_{\text{coverage } C(x)}, \quad a > 0, b \in (0, 1.5]. \quad (8)$$

This mixes an occupation-style barrier term with a terminal coverage shape; we show in §7 that it is poorly calibrated for the terminal payoff (2). Each estimator is clamped to $[P_{floor}, P_{ceil}]$.

(D) Double-Barrier Survival (Path-Dependent Tiles)

For a stay-in-range tile that pays iff the price never leaves $[L, U]$ over the active window (a corridor no-exit / double-barrier contract), the correct object is a first-passage probability, not a terminal one. For driftless Brownian motion with dollar spread $s = S_0 \sigma_{rel,eff} \sqrt{T_2}$ started at $S_0 \in (L, U)$, the method of images gives the survival probability

$$p_D = \sum_{n=-\infty}^{\infty} [(\Phi(U - S_0 - 2n\ell/s) - \Phi(L - S_0 - 2n\ell/s)) - (\Phi(U - \tilde{S}_0 - 2n\ell/s) - \Phi(L - \tilde{S}_0 - 2n\ell/s))], \quad (9)$$

with corridor width $\ell = U - L$ and lower-barrier image $\tilde{S}_0 = 2L - S_0$; the series is truncated at $|n| \leq 6$ (fast convergence for $s \lesssim \ell$). We validate (9) against fine-grid Monte-Carlo in §7 (mean error 1.7%) and contrast it with the naive terminal-only surrogate $\mathbb{P}(S_{t_e} \in [L, U])$, which ignores the path and badly over-credits survival.

Probability to Multiplier

The fair (edge-loaded) multiplier and its clamp are

$$M_{contract} = \text{clip}(1 - \text{HOUSE_EDGE}/p, \text{MIN_PAYOUT}, \text{MAX_PAYOUT}). \quad (10)$$

The clamp is an economic policy, not a modeling nicety: on low- p tiles the cap MAX_PAYOUT truncates the fair payout and thereby raises the realized margin above the configured edge (quantified in §7).

State-Dependent Smoothing

To avoid flickering quotes, the displayed multiplier is an EMA of the contract multiplier and the previous quote, with a faster weight in shock mode:

$$\alpha = \begin{cases} \text{ALPHA_SHOCK} & \text{shock} \\ \text{ALPHA_CALM} & \text{otherwise} \end{cases}$$

$$M_{quoted} = \text{round}(\text{clip}(\alpha M_{contract} + (1 - \alpha) M_{prev}), \text{DP}). \quad (11)$$

Smoothing introduces a small, bounded lag; §5 explains why this lag is not exploitable under synchronous acceptance-time re-pricing.

Exploit-Resistance Mechanism

At second-scale horizons the pricing model is necessary but not sufficient: a correct quote that is bindable while stale is a free option. We treat acceptance as a mechanism with the following rules.

1. **Server authority & synchronous re-pricing.** A bet is bound only at server acceptance, where the multiplier is recomputed against the freshest tick. The effective staleness is the acceptance latency λ (server-side), not the client's display age. This is the domain analogue of frequent batch auctions [20]: bindingness is synchronized to a fresh re-price rather than to a continuously live client quote.
2. **Quote-age rejection.** Reject acceptance if the quote backing the click is older than QUOTE_MAX_AGE_MS, even if the global feed is healthy. Bounds worst-case λ .
3. **No-bet lock window.** Reject new bets when $T_1 \leq \text{NO_BET_LOCK_MS}/1000$; the tile's multiplier is frozen into settlement. Removes lock-boundary timing.
4. **Feed-stale kill switch.** If no tick arrives within FEED_STALE_THRESH_MS, disable all quoting and acceptance and mark the board stale; resume only after N consecutive healthy ticks. Prevents pricing on a frozen feed.
5. **Fairness invariant.** displayed = locked = settled multiplier for every accepted bet, with the model version recorded. Makes disputes decidable and smoothing non-exploitable (the player is bound to the re-priced value, not the lagged display).

We now make the central claim precise and prove it.

Leakage Vanishes Under Synchronous Re-Pricing

Consider an adversary who binds a bet on a tile using information $\mathcal{G}$ (the public feed, possibly fresher than the platform's last published quote). At binding the server re-prices using its own information $\mathcal{F}_\lambda$, which is $\mathcal{G}$ lagged by the acceptance latency $\lambda \geq 0$; it quotes $M = \text{clip}((1 - e)/\hat{p}(\mathcal{F}_\lambda), M_{\min}, M_{\max})$ with $e = \text{HOUSE_EDGE}$ and model probability $\hat{p}$. Let $p^* = \mathbb{P}(\text{win} \mid \mathcal{G})$ be the adversary's true conditional win probability. The adversary's expected profit per unit stake on a bound tile is $\pi = p^*M - 1$, and the platform's leakage on a tile is $\max(0, \pi)$.

Proposition 1 (Synchronous re-pricing closes stale-quote leakage). Assume (A1) the model is conditionally calibrated on its own inputs in the unclamped region, $\mathbb{P}(\text{win} \mid \mathcal{F}_\lambda) = \hat{p}(\mathcal{F}_\lambda)$; (A2) binding uses the freshest tick, so $\mathcal{F}_0 = \mathcal{G}$; and (A3) the map from the latest price to $\hat{p}$ is L_0 -Lipschitz and the price has bounded expected move $\mathbb{E}[S_{t_{\text{now}}} - S_{t_{\text{now}}-\lambda}] \leq \bar{S} \bar{\sigma} \sqrt{\lambda}$ over the latency. Then: (i) under synchronous re-pricing $\lambda = 0$, every tile satisfies $\mathbb{E}[\pi \mid \mathcal{G}] = -e < 0$, so no tile is positive-expectation and leakage is identically zero; and (ii) for $\lambda > 0$ the expected leakage per offered tile is bounded by $\mathbb{E}[\max(0, \pi)] \leq \frac{(1-e)L_0 \bar{S} \bar{\sigma}}{P_{\text{floor}}} \sqrt{\lambda} = C \sqrt{\lambda} \xrightarrow{\lambda \rightarrow 0} 0$.

Proof. (i) With $\lambda = 0$, (A2) gives $\mathcal{F}_0 = \mathcal{G}$, so by (A1) $p^* = \mathbb{P}(\text{win} \mid \mathcal{G}) = \hat{p}(\mathcal{F}_0)$. In the unclamped region $M = (1 - e)/\hat{p}$, hence $\mathbb{E}[\pi \mid \mathcal{G}] = p^* \cdot (1 - e)/p^* - 1 = -e < 0$; clamping only lowers M on extreme tiles, which cannot raise π . No tile has $\pi > 0$, so a rational adversary binds none and leakage is zero. (ii) Write $\pi = (p^* - \hat{p})M + (\hat{p}M - 1) = (p^* - \hat{p})M - e$ in the unclamped region. By (A3) and calibration of $\hat{p}$ at $\mathcal{F}_\lambda$, $\mathbb{E}[p^* - \hat{p}] \leq L_0 \mathbb{E}[S_{t_{\text{now}}} - S_{t_{\text{now}}-\lambda}] \leq L_0 \bar{S} \bar{\sigma} \sqrt{\lambda}$. Since $M \leq (1 - e)/P_{\text{floor}}$ and $\max(0, \pi) \leq M \max(0, p^* - \hat{p})$, $\mathbb{E}[\max(0, \pi)] \leq 1 - e/P_{\text{floor}} \mathbb{E}[p^* - \hat{p}] \leq C \sqrt{\lambda}$ with $C = (1 - e)L_0 \bar{S} \bar{\sigma}/P_{\text{floor}}$. $\square$

The $\sqrt{\lambda}$ rate is a conservative upper bound from the diffusion scale of the price move; empirically (§7) leakage falls faster than $\sqrt{\lambda}$ because the fraction of positive-expectation tiles also shrinks to zero. Bounding staleness (quote-age rejection) caps λ ; only synchronous re-pricing sets $\lambda = 0$ and removes the residual entirely.

Assumption (A1) is the model's job (§4, §7); where it fails—e.g. under jumps—the residual inherits a calibration term, which motivates the jump-aware extensions of §8.

Calibration Protocol

Free parameters are fit by bucketed historical replay rather than global averages, because calibration error concentrates in corners of the tile space. The replay is stratified by (i) time-to-start T_1 , (ii) band width, (iii) distance from mid to the nearest edge, and (iv) a volatility-regime label derived from σ_{rel} . Within each bucket we compare predicted p to the realized hit frequency and fit: $\{a, b\}$ in $\mathcal{C}(x)$; SIGMA_REL_FLOOR; the shock parameters; the smoothing α 's; and the lock/quote-age thresholds. Acceptance gates before launch are:

1. per-bucket calibration error (predicted vs. realized hit rate) within tolerance;
2. realized margin near the configured edge after clamps;
3. no adversarial strategy in simulation profitable beyond tolerance.

In production, calibration drift, realized-vs-configured edge by bucket, $M_{quoted} - M_{contract}$ dispersion, shock entry/exit rates, and lock-window reject reasons are monitored continuously.

Synthetic Evaluation

Setup.

Because production data are proprietary, we evaluate on fully synthetic, released, pure-Python generators with fixed seeds. We use two data-generating processes. The Gaussian microbenchmark (tessera_sim.py) draws ticks at 2 Hz from a geometric Brownian motion whose per- $\sqrt{s}$ relative volatility is drawn per tile from four regimes—calm (8×10^{-5}), normal (1.6×10^{-4}), active (3.2×10^{-4}), stress (6.0×10^{-4}). The non-circular benchmark (tessera_sim_v2.py) instead uses a Heston stochastic-volatility process [14] (mean-reverting variance, vol-of-vol ξ , leverage $\rho = -0.4$) with Merton jumps [13] ($\sim 1.3/\text{min}$, ~ 16 bp size)—a law with fat tails and volatility clustering that the Gaussian pricer cannot represent, so calibration there measures genuine accuracy rather than tautology. For each tile we build a 60 s buffer, estimate $\sigma_{rel,eff}$ and μ_{eff} online, sample $T_1 \in \{2, 5, 10, 20, 40\}$ s, $T_2 \in \{5, 10, 20\}$ s and a band, then simulate to t_e . All estimators price the same tiles. Configuration: HOUSE_EDGE= 5%, $M \in [1.1, 5.0]$, $p \in [0.02, 0.92]$.

Calibration on the Gaussian microbenchmark (control)

Under GBM the terminal law is Gaussian, so this regime is a correctness check, not evidence of generalization. Table 1 reports ECE and realized margin per estimator; Figure 1 is the reliability diagram for estimator A ($N = 40,000$ tiles). The terminal-Gaussian estimator is essentially perfectly calibrated (ECE 0.88%; points on the diagonal), the shipped midpoint variant is mildly over-confident (ECE 3.46%), and the barrier $\times$ coverage heuristic is badly miscalibrated for the terminal payoff (ECE 27.3%)—confirming that its “blend” is not a principled hit probability.

Table 1. Calibration and economics by estimator ($N = 40,000$ tiles). ECE is volume-weighted mean $|pred - realized|$ over 20 probability bins; realized margin is $1 - \text{payout/stake}$ (target 5%).

Estimator	ECE	Realized house margin
A Terminal-Gaussian (recommended)	0.88%	4.67%
B Midpoint-Gaussian (as shipped)	3.46%	12.47%
C Barrier $\times$ coverage heuristic	27.28%	40.66%

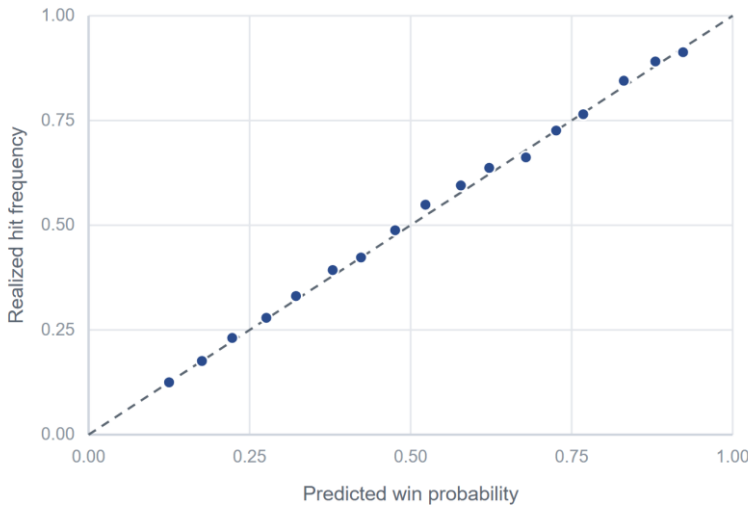


Figure 1. Reliability diagram for the terminal-Gaussian estimator (bins with ≥ 200 tiles). Points track the diagonal; ECE = 0.88%.

Table 2 shows estimator A holds within ± 2 percentage points across all T_1 buckets. The decisive test is the next subsection.

Table 2. Calibration of estimator A by time-to-start bucket. Gap = mean predicted – realized (percentage points).

T_1 (s)	2	5	10	20	40
mean predicted	0.327	0.418	0.479	0.521	0.545
realized	0.334	0.439	0.478	0.529	0.542
gap (pp)	−0.71	−2.05	+0.03	−0.81	+0.28

Breaking Circularity: Stochastic Volatility and Jumps

We now replace the volatility-input choice with three baselines—NAIVE (a fixed, unconditional volatility that ignores live estimation), EWMA (the online half-life estimate of §4), and EWMA+SHOCK (with the jump-triggered widening)—and price the same tiles under both generators (Table 3). On GBM all three are indistinguishable (ECE 0.91%): the Gaussian pricer is correct by construction and even a fixed volatility suffices because the regime is constant. Under Heston+jumps the story separates sharply: the naive fixed-volatility pricer degrades to ECE 6.77% because it cannot track the time-varying regime, whereas online EWMA estimation roughly halves the error to 2.98%. The residual $\sim 3\%$ is the honest cost of a Gaussian terminal approximation against a fat-tailed, jumpy law—not an artifact of a self-fulfilling generator. Shock widening does not improve aggregate ECE (its value is economic tail-safety, §7.4); jump risk is the binding limitation, which §8 addresses.

Table 3. Terminal-payoff calibration under the Gaussian control vs. the non-circular Heston+jumps benchmark ($N = 30,000$ tiles each). Live estimation (EWMA) roughly halves the naive error once the process is no longer the model’s own.

	GBM (control)		Heston + jumps	
	ECE	margin	ECE	margin
2-3(lr)4-5 Volatility input				
NAIVE (fixed vol)	0.91%	6.00%	6.77%	15.78%

EWMA (online)	0.91%	5.44%	2.98%	3.52%
EWMA+SHOCK	0.91%	5.41%	3.02%	3.29%

Path-Dependent Tiles and the Double-Barrier Model

For a stay-in-range tile (pays iff the price never leaves $[L, U]$ over the window) we first validate the image-series survival probability (9): against fine-grid ($\Delta t = 0.05$ s) Monte-Carlo of driftless Brownian motion over 1,200 random corridors, the mean absolute error is 1.7% (residual dominated by MC sampling noise on near-zero-probability corridors), confirming the formula. On discretely-settled tiles (Table 4) the proper double-barrier model beats the naive terminal-only surrogate $3 \times$ under GBM (ECE 5.1% vs. 16.1%; the 5% residual is discrete-monitoring bias relative to the continuous formula). Under Heston+jumps both degrade and the ordering inverts: a single jump can exit the corridor instantaneously, which the continuous-diffusion survival over-credits, so path-dependent tiles cannot be priced accurately by any diffusion-only model. This localizes exactly where a jump model is required rather than merely nice to have.

Table 4. Stay-in-range (double-barrier) calibration. The image-series model (9) is validated to 1.7% vs. continuous MC; under jumps all diffusion models break because corridor exit becomes discontinuous.

Survival estimator	ECE (GBM)	ECE (Heston+jumps)
Double-barrier images (9)	5.13%	17.33%
Terminal-only approximation	16.09%	8.44%

Economics and the Clamp Distortion

Aggregate realized margin for the calibrated estimator on the microbenchmark (4.67%) is close to the 5% target, but two effects redistribute it. First, the MAX_PAYOUT cap (Table 5). On low-probability tiles ($p \in [0.10, 0.25]$) the fair multiplier exceeds the 5.0 cap, so the house over-collects (+10.6%), whereas moderate/high-probability tiles sit near the intended $\sim 3\text{--}4\%$. This is a design lever, not a bug: operators trading off tail payout attractiveness against risk should monitor per-bucket realized edge, as the calibration protocol mandates.

Table 5. Realized house margin by win-probability bucket, estimator A. The $M \leq 5.0$ cap inflates margin on low- p tiles.

p bucket	[0.10, 0.25)	[0.25, 0.50)	[0.50, 0.75)	≥ 0.75	aggregate
realized margin	+10.55%	+3.10%	+3.60%	+2.78%	+4.67%

Second, and more consequentially, fat tails erode the edge. Reading the margin columns of Table 3: under Heston+jumps the calibrated EWMA pricer realizes only 3.52% margin against the 5% target—jump tails let players win high-multiplier tiles more often than a Gaussian predicts, costing ~ 1.5 percentage points of edge. The naive pricer’s apparent 15.78% “margin” is not a safe edge but systematic mispricing (it would bleed on the tiles it underprices to an informed player). The operational implication is a small jump premium added to the edge on high-multiplier tiles, calibrated by the bucketed replay of §6.

Stale-Quote Sniping: Confirming Proposition 1

We instantiate the §3.3 sniper: on 60,000 short tiles the adversary observes the fresh price after an effective quote-staleness $\lambda \sim \mathcal{U}(0, \lambda_{max})$ and bets only tiles whose true conditional win probability, given the fresh price and remaining horizon, makes the (stale) quoted multiplier positive-expectation. We report the adversary’s expected ROI on placed bets, the fraction of tiles bet, and the resulting house leakage per offered tile. Table 6 and Figure 2 show a clean monotone collapse: naive quoting ($\lambda_{max} = 1$ s) leaks 0.78% of tile volume; tightening the acceptance re-price drives leakage to 0.04% at $\lambda_{max} = 0.1$ s and to exactly zero under

synchronous re-pricing ($\lambda_{max} = 0$), where no tile is positive-expectation for the adversary. This is exactly Proposition 1: the empirical leakage lies below the $C\sqrt{\lambda}$ bound and, as predicted, the conditional ROI floor ($\sim 5\%$) persists—an adversary who does find a stale tile still profits—while the fraction of stale tiles collapses to zero, so at $\lambda = 0$ no positive-expectation tile exists. The defense works by removing exploitable tiles, not by shrinking their per-tile edge, matching the synchronous-clearing intuition of [20].

Table 6. Stale-quote sniping vs. acceptance staleness λ_{max} (60,000 tiles). Leakage/tile = expected adversary profit per offered tile.

λ_{max} (s)	Adversary exp. ROI	Tiles bet	Leakage / tile
1.00 (no re-price)	+4.99%	15.7%	0.78%
0.50	+4.06%	9.2%	0.38%
0.25	+3.05%	5.3%	0.16%
0.10 (acceptance re-price)	+2.16%	2.0%	0.04%
0.05	+1.69%	0.8%	0.01%
0.00 (synchronous)	—	0.0%	0.00%

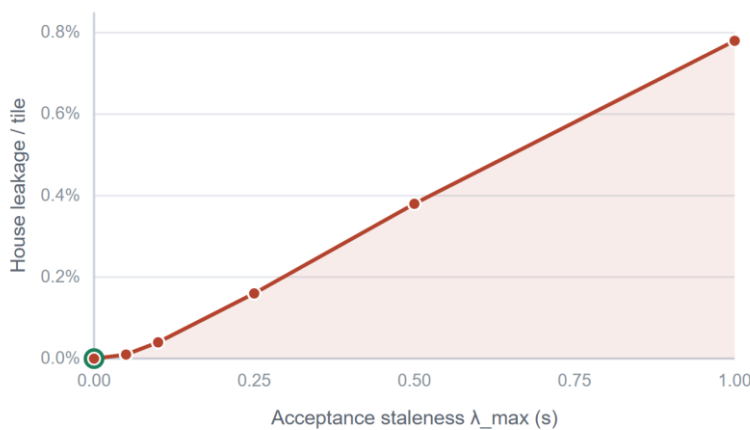


Figure 2. Stale-quote leakage collapses to zero as acceptance-time re-pricing becomes synchronous ($\lambda_{max} \rightarrow 0$).

Real-Tick Replay: BTC/USD 1-Second Data

Synthetic processes cannot exhibit the microstructure of a live feed, so we finally replay the engine on real ticks: 40,000 contiguous 1-second BTC/USD close prices (≈ 11.1 h, public exchange market data, btc_1s.csv). We form 40,000 tiles exactly as before—build the 60 s buffer, estimate $\sigma_{rel,eff}$ and μ_{eff} online, and settle terminal-in-range against the actual future price. The sample is an unusually calm regime (price range $\pm 0.5\%$, net $-\$32$ over 11 h), which makes it a stringent test of the volatility floor.

Scaling Diagnostic.

The realized 1-second log-return volatility is $\sigma_{rel} = 2.77 \times 10^{-5}/\sqrt{s}$ —below the synthetic study’s floor SIGMA_REL_FLOOR = 5×10^{-5} . Moreover, q -second dispersion grows slightly faster than $\sqrt{q}$: the ratio $std_q/(\sqrt{q} std_1)$ rises from 1.03 ($q = 2$) to 1.24 ($q = 300$), i.e. an effective scaling exponent ≈ 0.54 with weak positive return autocorrelation (+0.06)—mild super-diffusion, not the microstructure mean-reversion one might expect.

Result.

Table 7 is decisive in two directions. Applied off the shelf with the synthetic floor, the engine fails badly (ECE 26.0%, realized margin -81.8%): the floor exceeds the true volatility, so the model assumes $\sim 1.8 \times$ the real dispersion and systematically under-prices win probability. Re-fitting that single parameter to the data (the calibration protocol of §6, floor 1×10^{-5}) cuts ECE to 5.5% and margin to -3.3% —a validation both of the engine and of the paper’s central insistence that $\sigma_{rel,eff}$ -related constants are calibrated per asset. The naive fixed-volatility and barrier $\times$ coverage baselines remain far worse. The 5.5% residual is structured (Figure 3: mild under-prediction of low probabilities, over-prediction of high; and a T_1 trend from -15 to $+8$ pp), a fingerprint of the super-diffusive scaling that a fitted exponent—not the assumed 0.5—would absorb. That the realized margin is still slightly negative (-3.3% vs. the $+5\%$ target) on this one calm sample is the honest headline: a single recalibration makes the engine usable on real ticks but not yet provably profitable; the scaling correction and the live monitoring of §6 are prerequisites, and a longer multi-regime tape is required before any production claim.

Table 7. Real-tick replay on BTC/USD 1-second data (40,000 tiles, ≈ 11.1 h). Re-fitting the single volatility-floor parameter to the data cuts calibration error nearly $5 \times$; uncalibrated deployment fails.

Estimator (terminal-in-range)	ECE	Realized margin
EWMA, synthetic floor (5×10^{-5} , uncalibrated)	25.99%	-81.83%
Naive fixed volatility	14.57%	-38.87%
Barrier $\times$ coverage heuristic	25.52%	$+38.96\%$
EWMA, data-calibrated floor (1×10^{-5})	5.52%	-3.28%

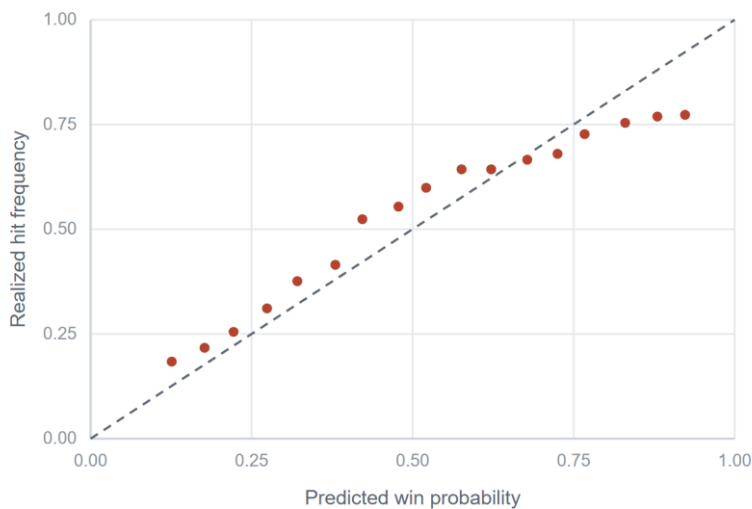


Figure 3. Reliability on real BTC/USD 1-second ticks after single-parameter recalibration (ECE 5.5%). The residual S-shape—under-prediction of low and over-prediction of high probabilities—is the signature of mild super-diffusion ($\text{std}_q \propto q^{0.54}$), which a fitted scaling exponent would correct.

DISCUSSION AND LIMITATIONS

Heuristic Vs. Principled Probability

The barrier $\times$ coverage estimator is intuitive and reactive but conflates an occupation (touch) term with a terminal coverage shape; our results quantify how badly this miscalibrates the terminal payoff. Practitioners

should either price the payoff they actually settle (terminal $\Rightarrow$ estimator A) or, for a genuine touch/occupation product, use a double-barrier occupation probability rather than the reflection single-barrier proxy [16, 15].

Model Risk (Measured)

Estimator A assumes a locally Gaussian terminal return with an online volatility estimate. Our non-circular benchmark makes the cost explicit: against Heston+jumps the calibrated pricer degrades from 0.9% to 3.0% ECE and loses ~ 1.5 percentage points of realized edge to fat tails (§7.2, §7.4), and diffusion-only survival pricing fails outright for path-dependent tiles once jumps are present (§7.3). The two natural fixes are a Merton jump term in the terminal distribution [12, 11] and explicit volatility-of-volatility [11]; the shock multiplier and drift clip are pragmatic guards, not a substitute for either. Extending calibration to a jump-aware terminal law is the highest-value next step, and—because a real deployment settles on discrete oracle ticks—a discrete-monitoring correction to (9) should accompany it.

Calibration is Not Optional (Measured On Real Data)

The real-tick replay (§7.6) is the sharpest lesson: an engine that is near-perfect on its own generator and well-behaved under Heston+jumps still fails on real BTC ticks when deployed with an uncalibrated volatility floor (ECE 26%, margin -82%), and is restored to ECE 5.5% by re-fitting one parameter. Real BTC also shows mild super-diffusion ($\text{std}_q \propto q^{0.54}$), so the assumed $\sqrt{H}$ scaling should become a fitted exponent. The residual negative margin on a single calm sample means we claim usable after calibration, not provably profitable; a longer multi-regime tape and the scaling correction are the remaining steps.

Oracle and Settlement

We trust the oracle's aggregated price; a compromised attestation or a publisher outage is out of scope but is precisely why the feed-stale kill switch is mandatory. Settlement latency and on-chain finality add a small terminal uncertainty not modeled here.

Mechanism Residuals

Synchronous re-pricing eliminates stale-quote leakage in simulation, but a truly latency-advantaged adversary who can also influence the oracle's marginal publisher, or who exploits cross-venue information ahead of the feed, is not addressed by acceptance-time re-pricing alone. Batching or randomized acceptance delays [20] are complementary defenses.

Generality

The pipeline is asset- and venue-agnostic: any sub-second reference feed and any range/terminal payoff on a lattice fit the same estimator and mechanism. Extending calibration jointly across correlated assets is future work.

CONCLUSION

Short-horizon on-chain range-prediction markets demand a pricing engine that is simultaneously well-calibrated, economically faithful, and exploit-resistant in real time. A scale-invariant EWMA volatility estimate feeding a terminal-Gaussian in-range probability, edge-loaded and clamped, is well calibrated, and—tested on a deliberately non-circular Heston+jumps benchmark rather than only on the Gaussian process it can fit—online estimation roughly halves the error of a naive fixed-volatility pricer, while a validated double-barrier model handles path-dependent tiles under diffusion. We isolated jump risk as the binding limitation (a ~ 1.5 pp edge cost and the failure point of diffusion survival pricing) and named the concrete fix. Finally we cast quote acceptance as synchronous clearing and proved that stale-quote leakage is $O(\sqrt{\lambda})$ in acceptance latency, vanishing under synchronous re-pricing—confirmed in simulation. Replayed on real 1-second BTC/USD ticks, the engine fails off the shelf but recovers to 5.5% ECE after re-fitting a single volatility-floor

parameter, quantifying exactly how much per-asset calibration matters. Model, calibration protocol, simulators, and replay data are released. The remaining steps toward a production claim are a longer multi-regime real tape, a fitted volatility-scaling exponent, and a jump-aware terminal law.

Data Availability and Reproducibility

No proprietary or production data were used. Synthetic figures and tables are produced by pure-Python simulators with fixed seeds: `tessera_sim.py` (Gaussian microbenchmark, adversarial study), `tessera_sim_v2.py` (Heston+jumps robustness, baselines, double-barrier tiles), and `validate_double_barrier.py` (formula validation). The real-tick replay (§7.6) uses `btc_1s.csv`—40,000 public 1-second BTC/USD close prices retrieved from a public exchange market-data endpoint—via `replay_real.py` and `vol_scaling_diagnostic.py`. Author names, affiliation, and platform identity are anonymized placeholders. Reviewers can regenerate every number in §7 by running the released scripts.

Acknowledgments

Omitted for anonymized review.

REFERENCES

1. R. Hanson. Logarithmic market scoring rules for modular combinatorial information aggregation. *Journal of Prediction Markets*, 1(1):3–15, 2007.
2. A general theory of liquidity provisioning for prediction markets. arXiv:2311.08725, 2023. <https://arxiv.org/abs/2311.08725>.
2. Toward Black–Scholes for prediction markets: a unified kernel and market maker’s handbook. arXiv:2510.15205, 2025. <https://arxiv.org/abs/2510.15205>.
3. Pew Research Center. Trading volume on prediction markets has soared in recent months. . <https://www.pewresearch.org/short-reads/2026/05/27/trading-volume-on-prediction-markets-has-soared-in-recent-months/>.
4. The Block. Kalshi and Polymarket post strongest months yet with nearly \$8 billion in combined November volume. . <https://www.theblock.co/post/380909>.
5. C. Dim, B. Eraker, and G. Vilkov. 0DTEs: trading, gamma risk and volatility propagation. SSRN 4692190, 2024. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4692190.
6. Revisiting local expansions for zero days-to-expiry option pricing. ScienceDirect, 2025. <https://www.sciencedirect.com/science/article/pii/S3050700625000532>.
7. J.P. Morgan/Reuters. RiskMetrics—Technical Document, 4th ed., 1996.
8. Asset volatility forecasting: the optimal decay parameter in the EWMA model. arXiv:2105.14382, 2021. <https://arxiv.org/abs/2105.14382>.
9. T. G. Andersen, T. Bollerslev, F. X. Diebold, and P. Labys. Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625, 2003.
10. Y. Du et al. Pricing cryptocurrency options with volatility of volatility. *Journal of Futures Markets*, 2025. <https://onlinelibrary.wiley.com/doi/10.1002/fut.70029>.
11. R. C. Merton. Theory of rational option pricing. *Bell Journal of Economics and Management Science*, 4(1):141–183, 1973.
12. R. C. Merton. Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, 3(1–2):125–144, 1976.
13. S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6(2):327–343, 1993.
14. S. Kou. On pricing of discrete barrier options. *Statistica Sinica*, 13:955–964, 2003. <https://www.columbia.edu/~sk75/sinica.pdf>.
15. N. Kunitomo and M. Ikeda. Pricing options with curved boundaries. *Mathematical Finance*, 2(4):275–298, 1992.
16. Pyth Network. Pull, don’t push: a new price oracle architecture. . <https://www.pyth.network/blog/pyth-a-new-model-to-the-price-oracle>.

17. P. Daian, S. Goldfeder, T. Kell, et al. Flash Boys 2.0: frontrunning, transaction reordering, and consensus instability in decentralized exchanges. IEEE S&P, 2020.
18. L. Zhou, K. Qin, et al. High-frequency trading on decentralized on-chain exchanges. arXiv:2009.14021, 2021. <https://arxiv.org/abs/2009.14021>.
19. E. Budish, P. Cramton, and J. Shim. The high-frequency trading arms race: frequent batch auctions as a market design response. Quarterly Journal of Economics, 130(4):1547–1621, 2015.

Configuration Parameters

Parameter	Meaning	Example
VOL_WINDOW_S	volatility look-back window	60 s
VOL_HALF_LIFE_S	EWMA half-life	15 s
SIGMA_REL_FLOOR	min relative vol / $\sqrt{s}$	1×10^{-4}
MOMENTUM_WINDOW_S	drift look-back	10–20 s
DRIFT_CLIP_BPS_PER_S	max drift magnitude	1 bp/s
SHOCK_Z	shock trigger (std. returns)	3.0
SHOCK_SIGMA_MULT	shock spread multiplier	1.15–1.40
HOUSE_EDGE	configured edge	0.05
P_FLOOR, P_CEIL	probability clamps	0.02, 0.92
MIN_PAYOUT, MAX_PAYOUT	multiplier clamps	1.1, 5.0
ALPHA_CALM, ALPHA_SHOCK	smoothing weights	0.25, 0.70
QUOTE_MAX_AGE_MS	max quote age to accept	1200
NO_BET_LOCK_MS	pre-start lock window	400–700
FEED_STALE_THRESH_MS	feed kill-switch threshold	2000

Reference Pricing Pseudocode

```

sigma_rel  = ewma_relative_vol(ticks)      # Eq. (4), floored
sigma_eff  = shock_adjust(sigma_rel, r_last) # Eq. (5)
mu_eff     = decay(clip(momentum(ticks)), H) # Sec. 4.3
S_hat      = S0 + mu_eff * (T1 + T2/2)
Sigma_H    = max(S0 * sigma_eff * sqrt(T1+T2), EPS)
p          = Phi((U-S_hat)/Sigma_H) - Phi((L-S_hat)/Sigma_H) # estimator A
p          = clip(p, P_FLOOR, P_CEIL)
M_contract = clip((1-HOUSE_EDGE)/p, MIN_PAYOUT, MAX_PAYOUT)
alpha      = shock ? ALPHA_SHOCK : ALPHA_CALM
M_quoted   = round(clip(alpha*M_contract + (1-alpha)*M_prev), DP)
# acceptance: re-price synchronously against freshest tick; persist locked value

```