Overlay V1 Core

Michael Feldman, Adam Kay, Anantdeep Parihar, Rachel Rybarczyk, and Jonah Lin
(Dated: December 19, 2023)

We extend Kay's original work on the Overlay protocol to attempt stability and robustness. In order to enable markets on *any* type of data stream driven by a random process, Overlay V1 Core relies heavily on the statistical properties of the underlying feed, requiring risk parameters to be calibrated based on fits to historical data. The original vision of the protocol is updated with a three-pronged risk framework: funding payments for open interest balance; bid-ask pricing to deter front-running; and caps to limit damage from tail behavior.

I. INTRODUCTION

Overlay enables markets on streams of data without the need for traditional counterparties or liquidity pools [1]. The protocol accomplishes this through its native token, OV. The mechanism is relatively simple: Traders stake OV as collateral on an Overlay market to enter a position on a particular stream. If the trade is profitable when the trader looks to unwind, the protocol mints more OV to the circulating supply to compensate the trader for their gains. Conversely, if the trade is at a loss, the protocol burns OV from the position's staked collateral, removing the tokens from the circulating supply.

An example to help clarify. Imagine we think the floor price on PUNK NFTs is not sustainable and will likely go down in ETH terms. We wish to short the floor. As a trader, we enter a short position on the associated Overlay market for the PUNK/ETH floor price feed:

- We stake 100 OV short at an entry price of 80 ETH for the PUNK/ETH floor
- The PUNK/ETH floor then drops 10 ETH (-12.5%) to 70 over the next week
- We unwind the position to take profit: Overlay mints 12.5 OV for the PnL and returns a total of 112.5 OV to us for the trade.

If the PUNK floor had gone up 12.5% to 90 ETH, Overlay would burn 12.5 OV from the stake and return 87.5 OV back to us.

The original whitepaper left some unanswered questions. The most important being: how to limit excessive inflation of the currency supply? While a particular inflation rate for the currency supply is not guaranteed by any means given the approach in this work, the tools outlined below offer the ability to manage the risk assumed by the protocol at any given time, in a probabilistic sense. This, in turn, can act as a guide for token holders as to the expected inflation rate.

Since OV functions as the governance token of the protocol, it is up to token holders to decide what balance they wish to take between inflation risk vs increased potential volumes and platform usage. On one end of the spectrum, risk parameters can be tuned so strict as to inhibit all trading volume for zero inflation. On the other end of the spectrum, they can be tuned so loose as to enable large trading volumes but with significant risk of hyperinflation. This work gives guidance on how to set risk parameters to balance these two opposing forces, using the statistical properties of each market's underlying feed to calibrate appropriately.

For V1, focus on price feeds as the initial markets. Following Mandelbrot [2], assume the underlying price P from the stream is driven by a Lévy process L_τ of the form

$$P(t+\tau) = P(t) e^{\mu \tau + \sigma L_\tau} \tag{1}$$

with Lévy stable [3][4] increments

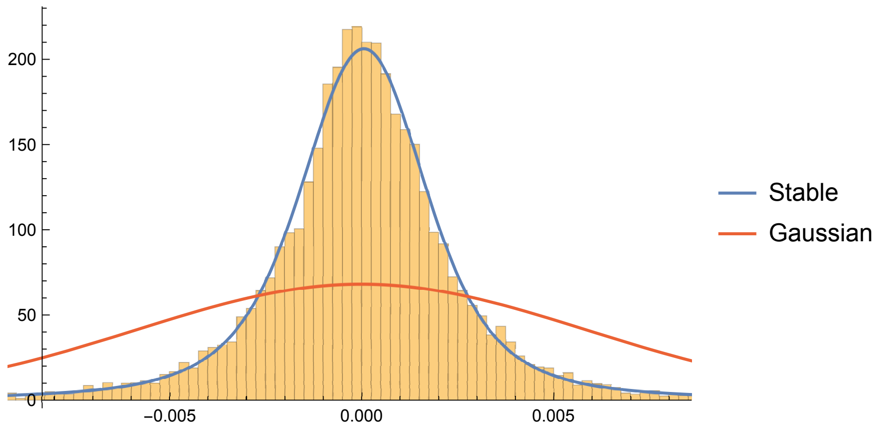
$$L_{t+\tau} - L_t \sim S(a, b, 0, (\tau / a)^{1/a}) \tag{2}$$

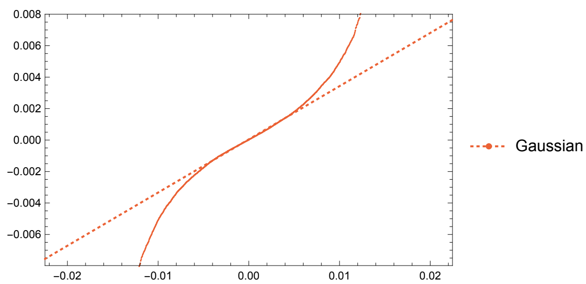
Stable distribution S parameters should be fit to historical data: $a \in (0, 2]$ for the stability parameter, $b \in [-1, 1]$ for the skewness parameter, 0 for the location parameter, and $(\tau/a)^{1/a}$ for the scale parameter. μ and σ are drift and volatility parameters. The price process reduces to Geometric Brownian motion (GBM) when $a = 2$. For simplicity, take

$$X_\tau \equiv \mu\tau + \sigma L_\tau \sim S(a, b, \mu\tau, \sigma (\tau/a)^{1/a}) \tag{3}$$

throughout. Given the Q-Q plots of log-price for e.g. WETH/USDC in Fig. 1, the log-stable assumption seems reasonable. Maximum likelihood estimation, such as in Fig. 2, as well as calculations involving the PDF f_{X_τ}, CDF F_{X_τ} and inverse CDF $F^{-1}_{X_\tau}$ use the `pystable` wrapper [5] for the libstable C library [6].

With stable random variables outside the Gaussian, the p-th moment of the distribution is only finite for $p < a$, which causes issues from a risk perspective when the payoff for the contract is a function of $P(t+\tau)/P(t) = e^{X_\tau}$. This is especially the case for Overlay as passive token holders, through inflation risk, act as the counterparty to any imbalance in open interest on markets offered. Variance of the stable process is infinite when $a \neq 2$. Mean of the stable process will be undefined when $a \leq 1$. The expected value (EV) of the position contract's payoff $\propto e^{X_\tau}$ will be undefined for a non-Gaussian X_τ process. To appropriately manage risk associated with the extreme behavior of the tails, however, the payoff function $g(X_\tau)$ can be capped [7]. This implicitly mitigates the damage associated with power-law tail behavior of the underlying, as it is now known what





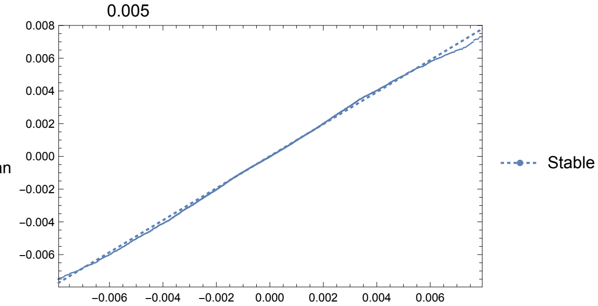


FIG. 1. Log-price fits in Mathematica for the 1 hour rolling TWAP on SushiSwap's WETH/USDC pool, sampled every 10 minutes from April 18, 2021 to July 16, 2021. Q-Q plots show the inability of the Gaussian to properly account for the observed tail behavior the stable fit can account for.

the trade's worst case payoff is *regardless* of how often extreme tail events occur: for a probability density function f_{X_τ}, $\int_{-\infty}^{g^{-1}(C_P)} g(x) f_{X_\tau}(x) + \int_{g^{-1}(C_P)}^{\infty} C_P f_{X_\tau}(x) < \infty$ is now finite and computable, where C_P is the constant payoff cap.

For Overlay, the PnL offered to a position contract should be linear in price, yet capped to guard against tail behavior. A position contract receives

$$\mathrm{PnL}(t, t+\tau) = \pm \mathrm{OI}(t+\tau) \cdot [P_{exit}(t+\tau) - P_{entry}(t)] \quad (4)$$

in PnL where OI is the open interest occupied by the position in units of number of position contracts, P_{entry} is the entry price given to the position at time t, P_{exit} is the exit price the position would receive at time $t+\tau$. $\pm = +1$ for a long and $= -1$ for a short. When $\pm[\frac{P_{exit}}{P_{entry}} - 1] > C_P$, the position contract PnL is limited to $\mathrm{PnL}(t, t+\tau) = \pm \mathrm{OI}(t+\tau) \cdot P_{entry}(t) \cdot C_P$, implementing the payoff cap. C_P is assumed to be larger than one, such that the payoff cap is irrelevant for shorts. The relevant uncapped payoff function used is

$$g(X_\tau) \equiv e^{X_\tau} - 1 \quad (5)$$

throughout.

There are two quantities of interest needed to determine the value of an Overlay position. They are

- Open interest: OI
- Position size (notional): Q

When an Overlay position is built by a trader at time t, the notional of the position is taken to be

$$Q \equiv N(t) \cdot L \quad (6)$$

where $N(t)$ is the initial OV collateral backing the position and L is the chosen initial leverage. Q is in units of OV.

The initial open interest associated with this position is taken to be the number of contracts the trader has entered into

$$\mathrm{OI}(t) \equiv \frac{Q}{P(t)} \quad (7)$$

where $P(t)$ is the oracle value fetched directly from the feed at time t when the position is built. $P(t)$ and $P(t+\tau)$ will differ from $P_{entry}(t)$ and $P_{exit}(t+\tau)$, respectively, given the pricing mechanisms detailed in Section III.

Given an initial amount $N(t)$ of OV collateral staked to back the position and a leverage value L chosen, Overlay markets track the open interest for the position and store a static reference to the debt

$$D = Q - N(t) = N(t) \cdot (L - 1) \quad (8)$$

"owed" by the position to the protocol. The protocol returns a balance

$$V(t, t+\tau) = N(t+\tau) + \mathrm{PnL}(t, t+\tau) \quad (9)$$

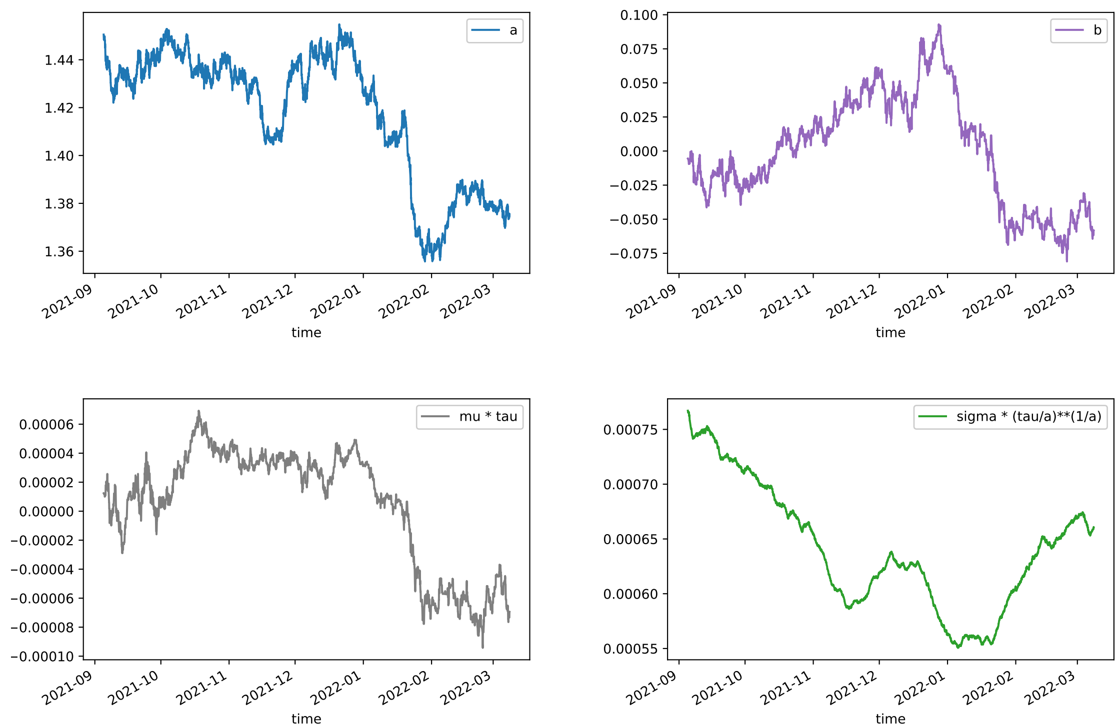


FIG. 2. Plots of maximum likelihood estimates (MLEs) of stable distribution parameters $S(a, b, \mu\tau, \sigma(\tau/a)^{1/a})$ for the 1 hour TWAP on $\tau = 10$ min candles from Uniswap V3's WETH/USDC 0.3% pool. `pystable` fits are over a prior 90 day rolling sample of data with sample end dates ranging from September 5, 2021 to March 8, 2022. Given the heavy use of the inverse CDF $F^{-1}_{X_\tau}(1-\alpha)$ at high confidence levels $1-\alpha$ in this work, Overlay per-market risk parameter calibrations will be most sensitive to variations in MLEs for a and σ. With WETH/USDC, fits fair relatively well over 10 months of data with a and σ estimates nearly constant to $\pm 2\%$ and $\pm 9\%$, respectively. Governance should consider continuous monitoring of risk parameters with potential adjustments every few months to maintain inflation targets in the event MLEs vary significantly over longer time horizons.

to the trader when unwinding their full position at $t+\tau$, given entry at t. Allow for changes in open interest over time to facilitate funding payments, such that the collateral backing the position at time $t+\tau$ will be

$$N(t+\tau) = Q \cdot \frac{\mathrm{OI}(t+\tau)}{\mathrm{OI}(t)} - D \tag{10}$$

Funding payments therefore alter position size and collateral amounts backing the position.

II. FUNDING PAYMENTS

The protocol takes on the profit liability associated with an imbalance in open interest. Funding payments are used to decrease open interest imbalance over time. Effectively, the overweight open interest side, which adds risk to the protocol, pays an interest rate on the number of contracts they hold to the underweight side on the same market. Unlike the traditional approach to funding [8], funding with Overlay is not used as a means to track price but instead as a means to limit the protocol's exposure to the market.

A. Market Exposure

Think about the printing risk the protocol takes on due to an imbalance in open interest in the following manner: Assuming no funding payments, what is the market exposure the protocol assumes when the open interest on both sides of the market remains imbalanced for a period of time?

Take a two trader example. Both initially enter the market at the same time t building the same amount of open interest $\mathrm{OI}_l = \mathrm{OI}_s = \mathrm{OI}_0$. Their positions perfectly balance each other, as the market exposure the protocol assumes through unrealized PnL to both sides is effectively zero. For simplicity, assume exit and entry prices differ insignificantly from oracle values fetched from the feed.

Further, assume the long trader exits a portion $w \cdot \mathrm{OI}_0$ of their contracts at $t+\tau_1$ and re-enters with another $w \cdot \mathrm{OI}_0$ contracts at a later time $t+\tau_2$, where $0 \leq w \leq 1$. The open interest is $(1-w)\cdot \mathrm{OI}_0$ on the long side and OI_0 on the short side when $\tau_1 \leq \tau \leq \tau_2$. Total (realized + unrealized) PnL exposure on both sides when $\tau \geq \tau_2$

will be

$$\mathrm{PnL}_l|_{\tau\geq\tau_2} = (1-w)\cdot \mathrm{OI}_0 \cdot \Big[P(t+\tau) - P(t)\Big] + w\cdot \mathrm{OI}_0 \cdot \Big[P(t+\tau_1) - P(t)\Big] + w\cdot \mathrm{OI}_0 \cdot \Big[P(t+\tau) - P(t+\tau_2)\Big] \quad (11)$$

$$\mathrm{PnL}_s|_{\tau\geq\tau_2} = \mathrm{OI}_0 \cdot \Big[P(t) - P(t+\tau)\Big] \quad (12)$$

such that the total exposure the protocol assumes is the price exposure on the contract imbalance when it exists between $\tau_1 \leq \tau \leq \tau_2$: $\mathrm{PnL}|_{\tau\geq\tau_2} = \mathrm{PnL}_l + \mathrm{PnL}_s = w\cdot\mathrm{OI}_0\cdot[P(t+\tau_1) - P(t+\tau_2)]$. The protocol aims to have this contract imbalance $\mathrm{OI}_l - \mathrm{OI}_s = -w\cdot\mathrm{OI}_0$ from $\tau_1 \leq \tau \leq \tau_2$ decay to zero through funding payments.

B. Imbalance Liability

Let the imbalance liability at time t when the market was last interacted with be

$$\mathrm{OI}_{imb}(t) \equiv \mathrm{OI}_l(t) - \mathrm{OI}_s(t) \quad (13)$$

where $\mathrm{OI}_l(t), \mathrm{OI}_s(t)$ are the aggregate open interest values on the long and short sides of the market at t, respectively. Positions track their share of the aggregate open interest on a side. Funding payments happen continuously in time. Take the drawdown to the imbalance liability to follow

$$\mathrm{OI}_{imb}(t+\tau) = \mathrm{OI}_{imb}(t)\cdot e^{-2k\tau} \quad (14)$$

between interactions with the market, where k is a per-market risk parameter directly linked with the time it takes to draw down the risk associated with an imbalanced market. Suggestions for k calibrations based on the risk the underlying feed adds to the system are given in Section II C.

Given a state for the open interest on the long and short sides at t, the state at time $t+dt$ for infinitesimally small dt will be

$$d\mathrm{OI}_o(t) = -d\mathrm{FP}(t) \quad (15)$$

$$d\mathrm{OI}_u(t) = [1 - b_r(t)]\cdot d\mathrm{FP}(t) \quad (16)$$

$$d\mathrm{OI}_{b_r}(t) = b_r(t)\cdot d\mathrm{FP}(t) \quad (17)$$

where $d\mathrm{FP}(t)$ is the funding payment paid by the overweight open interest side OI_o to the underweight side OI_u at time t, such that

$$\mathrm{OI}_o = \begin{cases} \mathrm{OI}_l, & \text{if } \mathrm{OI}_l > \mathrm{OI}_s \\ \mathrm{OI}_s, & \text{otherwise} \end{cases} \quad (18)$$

$$\mathrm{OI}_u = \begin{cases} \mathrm{OI}_s, & \text{if } \mathrm{OI}_l > \mathrm{OI}_s \\ \mathrm{OI}_l, & \text{otherwise} \end{cases} \quad (19)$$

OI_{b_r} is the cumulative number of contracts removed from the system to compensate the protocol for its pro-rata share of the imbalance liability (i.e. funding payment "burn" rate):

$$b_r(t) = \frac{|\mathrm{OI}_{imb}(t)|}{\mathrm{OI}_o(t)} \quad (20)$$

Re-framing the infinitesimal time evolution in terms of the total number of contracts outstanding $\mathrm{OI}_{tot}(t) \equiv \mathrm{OI}_l(t) + \mathrm{OI}_s(t)$ and the imbalance in contracts OI_{imb},

$$d\mathrm{OI}_{tot}(t) = -b_r(t)\cdot d\mathrm{FP}(t) \quad (21)$$

$$d\mathrm{OI}_{imb}(t) = -[2 - b_r(t)]\cdot d\mathrm{FP}(t) \quad (22)$$

The form of the imbalance drawdown in (14) implies the infinitesimal funding payment will be

$$d\mathrm{FP}(t) = \frac{\mathrm{OI}_{tot}(t) + |\mathrm{OI}_{imb}(t)|}{\mathrm{OI}_{tot}(t)}\cdot k\cdot \mathrm{OI}_{imb}(t)\cdot dt \quad (23)$$

The change in total contracts then follows

$$\mathrm{OI}_{tot}(t+\tau) = \mathrm{OI}_{tot}(t)\sqrt{1 - \left(\frac{\mathrm{OI}_{imb}(t)}{\mathrm{OI}_{tot}(t)}\right)^2\cdot\left(1 - e^{-4k\tau}\right)} \quad (24)$$

after integrating from t to $t+\tau$. Contracts "burned" can be determined through the relation

$$\mathrm{OI}_{b_r}(t+\tau) = \mathrm{OI}_{tot}(t) - \mathrm{OI}_{tot}(t+\tau) \quad (25)$$

These expressions can be used to determine the state of the open interest on the long and short sides at any time $t+\tau$ in the future between interactions with the market:

$$\mathrm{OI}_l(t+\tau) = \frac{1}{2}\cdot\Big[\mathrm{OI}_{tot}(t+\tau) + \mathrm{OI}_{imb}(t+\tau)\Big] \quad (26)$$

$$\mathrm{OI}_s(t+\tau) = \frac{1}{2}\cdot\Big[\mathrm{OI}_{tot}(t+\tau) - \mathrm{OI}_{imb}(t+\tau)\Big] \quad (27)$$

The product of the open interest on the long and short sides will also be invariant between interactions

$$\mathrm{OI}_l\cdot\mathrm{OI}_s = C_{l\cdot s} \quad (28)$$

for a constant $C_{l\cdot s} = \frac{1}{4}[\mathrm{OI}_{tot}^2 - \mathrm{OI}_{imb}^2]$ in time.

Funding rates paid by the overweight side f_o, received by the underweight side f_u, and burned to compensate the protocol f_{b_r} are of the same magnitude:

$$f_o = -\frac{1}{\mathrm{OI}_o}\frac{d\mathrm{OI}_o}{dt} = 2k\cdot\frac{|\mathrm{OI}_{imb}|}{\mathrm{OI}_{tot}} \quad (29)$$

$$f_u = \frac{1}{\mathrm{OI}_u}\frac{d\mathrm{OI}_u}{dt} = 2k\cdot\frac{|\mathrm{OI}_{imb}|}{\mathrm{OI}_{tot}} \quad (30)$$

$$f_{b_r} = \frac{1}{|\mathrm{OI}_{imb}|}\frac{d\mathrm{OI}_{b_r}}{dt} = 2k\cdot\frac{|\mathrm{OI}_{imb}|}{\mathrm{OI}_{tot}} \quad (31)$$

such that whatever rate the market deems as appropriate at the current time is being paid to the respective counterparty in full for the risk assumed, with $2k$ as the most

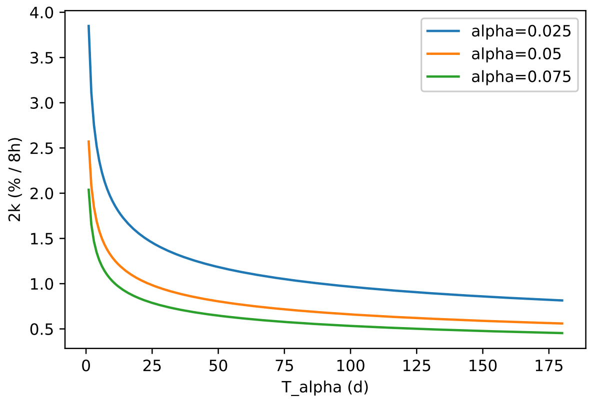


FIG. 3. Funding constant k curves on a WETH/USDC market for different $1-\alpha = [.925, .95, .975]$ confidence levels. y axis is the maximum funding rate experienced by traders on the market ($2k$) expressed as an 8h rate in percentage terms. x axis is the time T_α until the associated value at risk (VaR) to the protocol decays to zero with confidence $1-\alpha$.

extreme rate when the market is completely imbalanced to one side. Market expectations around the future value of the feed should manifest through the funding rate, acting as a carry cost/yield to hold the Overlay position.

Funding rates incentivize arbitrageurs to basis trade: taking positions on Overlay while swapping tokens on spot. Basis arbitrage should trend the open interest imbalance near zero in the process – if spot is liquid enough, funding rates should fluctuate around the difference in "risk-free" rates between quote and base currencies (i.e. no arbitrage condition).

Take, for example, the ETH/OV market on Overlay. Basis traders that prefer to denominate in ETH can:

- $\text{OI}_l < \text{OI}_s$: Swap ETH for OV on spot and Long 1x ETH/OV on Overlay
- $\text{OI}_l > \text{OI}_s$: Borrow OV with ETH collateral on a lending protocol. Swap half their borrowed OV for ETH on spot to deposit back in the lending protocol. Use the other half to Short 1x ETH/OV on Overlay

Similarly, basis traders that prefer to denominate in OV can:

- $\text{OI}_l < \text{OI}_s$: Borrow ETH with OV collateral on a lending protocol. Swap the ETH for OV on spot, and Long 1x ETH/OV on Overlay
- $\text{OI}_l > \text{OI}_s$: Swap half their OV for ETH on spot to deposit in a lending protocol. Use the other half to Short 1x ETH/OV on Overlay

Either trader is able to lock in profit in their currency of preference while helping to balance open interest on the market, regardless of the current funding rate direction.

TABLE I. Table of funding constant values k in units of bps/sec (i.e. 10^{-4}/sec) for different confidence levels $1-\alpha$ on a WETH/USDC market. Calibrations use fits to 10m candles from Fig. 2 with an end date of September 5, 2021: `StableDistribution[1, 1.45, -0.0055, 0.000012, 0.00077]`. Anchor time is taken to be $T_\alpha = 30$ days.

α	0.01	0.025	0.05	0.075	0.1
k	0.00418	0.00239	0.00162	0.00130	0.00110

C. Calibrating k

Funding constants should be calibrated per market based on the printing risk the underlying feed adds to the system. Assume traders all enter their positions at time t. Ignore entry and exit price differences with $P(t)$ and $P(t+\tau)$, respectively, and ignore the payoff cap to produce more conservative calibrations for k. At time $t+\tau$ in the future, the protocol will be liable for

$$\begin{aligned} \text{PnL}_{liability}(t, t+\tau) = &-\text{OI}_{b_r}(t+\tau)\cdot P(t) \\ &+\text{OI}_{imb}(t+\tau)\cdot[P(t+\tau) - P(t)] \end{aligned} \tag{32}$$

due to market exposure from any imbalance in open interest at time t.

As $2k$ represents the most extreme funding rate, take the scenario of all open interest to one side of the market to calibrate. In this scenario, the protocol is effectively the sole counterparty to all positions. When open interest is completely to one side, the total number of contracts and the magnitude of the imbalance in contracts are the same: $\text{OI}_{tot} = |\text{OI}_{imb}|$. The entire funding payment goes toward removing contracts from the system. The protocol's liability becomes

$$\begin{aligned} \text{PnL}_{liability}(t, t+\tau)|_{\text{OI}_{tot}=|\text{OI}_{imb}|} &= \text{OI}_{tot}(t)\cdot P(t) \\ &\cdot \begin{cases} e^{-2k\tau}\cdot\frac{P(t+\tau)}{P(t)} - 1, & \text{if OI}_l > \text{OI}_s \\ e^{-2k\tau}\cdot\left[2 - \frac{P(t+\tau)}{P(t)}\right] - 1, & \text{otherwise} \end{cases} \end{aligned} \tag{33}$$

Calibrate k such that the value at risk (VaR) to the market decays to zero at a time in the future $t + T_\alpha$ – the market should anticipate printing a maximum of zero OV with $1-\alpha$ confidence when positions are held for T_α. VaR to the protocol when all open interest remains to one side is

$$\begin{aligned} \text{VaR}_\alpha(t, t+\tau)|_{\text{OI}_{tot}=|\text{OI}_{imb}|} &= \text{OI}_{tot}(t)\cdot P(t) \\ &\cdot \begin{cases} e^{F^{-1}_{X_\tau}(1-\alpha) - 2k\tau} - 1, & \text{if OI}_l > \text{OI}_s \\ e^{-2k\tau}\cdot\left[2 - e^{F^{-1}_{X_\tau}(\alpha)}\right] - 1, & \text{otherwise} \end{cases} \end{aligned} \tag{34}$$

F_{X_τ} is the CDF of X_τ. $1-\alpha$ is the confidence level for which VaR is expected to be the maximum amount printed by the protocol (α will be small). For VaR to be zero when $\tau = T_\alpha$,

$$k_l = \frac{1}{2T_\alpha} \cdot F^{-1}_{X_{T_\alpha}}(1-\alpha) \tag{35}$$

$$k_s = \frac{1}{2T_\alpha} \cdot \ln\left(2 - e^{F^{-1}_{X_{T_\alpha}}(\alpha)}\right) \tag{36}$$

k_l is the calibration needed when longs outweigh shorts, and k_s for when shorts outweigh longs. Set

$$k = \max(k_l, k_s) \tag{37}$$

to be conservative with the market's funding rate.

Refer to T_α as the anchor time. Typical k values for various anchor times are shown in Fig. 3, with the funding constant expressed as an 8 hour rate.

III. PRICING

Each Overlay market can be thought of as a market maker creating its own level of "liquidity." Prices on Overlay are not dynamic in the traditional sense. Market contracts use values intermittently fetched from the oracle as a starting point for pricing, but apply price impact with bid-ask spreads to deter front-running. Assume either Uniswap V3 [9] or Balancer V2 [10] oracles as feeds, but the results should generalize to any oracle that experiences a time delay of ν between manipulation-resistant information available at the current time and the actual most recent value of the data stream.

A. Bid-Ask Spread

Take the 1 hour rolling TWAP in Fig. 4. While the longer rolling window increases the cost to manipulate the feed, updates to the TWAP lag any real changes in spot as the averaging window is over the *last hour*. It would be trivial for a trader to wait for a jump in spot, then enter a trade on the market in anticipation of the TWAP catching up to spot over the next hour.

To prevent traders from front-running the lag, a bid-ask spread is added to the entry and exit values offered to traders. For a longer TWAP window of Δ and a shorter TWAP window of $\nu \ll \Delta$, Overlay markets offer bid B and ask A prices of

$$B(t, \delta, \lambda q_B) = e^{-\delta - \lambda q_B} \cdot \min\Big[\mathrm{TWAP}(t-\nu, t), \mathrm{TWAP}(t-\Delta, t)\Big] \tag{38}$$

$$A(t, \delta, \lambda q_A) = e^{\delta + \lambda q_A} \cdot \max\Big[\mathrm{TWAP}(t-\nu, t), \mathrm{TWAP}(t-\Delta, t)\Big] \tag{39}$$

Longs receive the ask as their entry price and the bid as their exit price. Shorts receive the bid as their entry price and the ask as their exit price. Entry P_{entry} and exit P_{exit} prices are used to calculate the position contract PnL in Equation (4).

$\mathrm{TWAP}(t-\Delta, t)$ is the longer TWAP used to guard against traders aiming to profit from manipulating the underlying feed after entering a trade on the market. $\mathrm{TWAP}(t-\nu, t)$ is the shorter TWAP used as a proxy for significant changes in the most recent spot price. Using a TWAP as the spot proxy increases the cost of manipulation for the trader attempting to minimize the spread after an actual jump occurs.

δ is a static spread used to discourage front-running of the shorter TWAP. It offers protection against large jumps in spot that happen over shorter timeframes than ν. Fig. 5 plots bid and ask values with the static spread over the same interval of data as Fig. 4. Entering a long immediately after the jump and exiting 1 hour later is no longer profitable. Looking at two days worth of data shows the market remains tradeable.

λq_A, λq_B are market impact terms (i.e. price slippage) offering protection against large jumps in spot that exceed expectations used in calibrating δ. λ is a per-market impact constant that dictates the slippage a trader receives for queuing an additional OI_i worth of open interest to either the bid or the ask side. q_B and q_B are rolling volume cumulative sums of the open interest queued over the last ν period on the bid and the ask side of the market, respectively. These are normalized with respect to the current cap on open interest

$$C_{\mathrm{OI}}(t) \equiv \frac{C_Q}{P(t)} \tag{40}$$

where C_Q is the risk parameter for the notional cap (i.e. $Q \leq C_Q$) and $P(t)$ is the current mid price

$$P(t) = \frac{1}{2}\Big[B(t,0,0) + A(t,0,0)\Big] \tag{41}$$

fetched from the feed. Markets also use the mid price from the oracle feed when calculating the initial open interest to credit a trader in Equation (7).

As the TWAP catches up approximately linearly in time to the current spot price, any volume added at time t should not contribute significantly to market impact received by traders after $t + \nu$. The "roller" algorithm the market uses to limit the memory of the volume cumulative sums in a relatively gas efficient way is as follows. Assume trader i enters a trade for OI_i number of contracts at time $t + \tau$. The volume produced by this trade alone is taken to be

$$v_i(t+\tau)|_{\lambda q} = \frac{\mathrm{OI}_i}{C_{\mathrm{OI}}(t+\tau)} \tag{42}$$

If the trade is on the bid, the rolling cumulative volume $q = q_B$ is updated. Otherwise, if the trade is on the ask, $q = q_A$ is updated. The update to the rolling cumulative

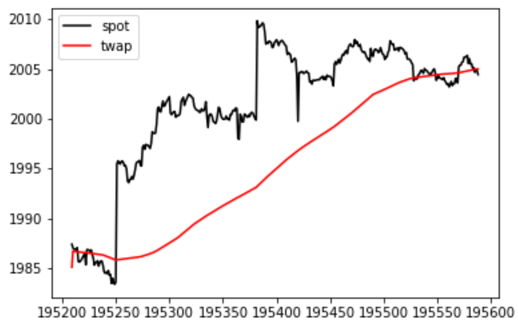


FIG. 4. Plot of the 1 hour rolling TWAP against spot price for Monte Carlo simulated 10m data generated from WETH/USDC fits. The jump in spot is not fully realized in the value of the 1h TWAP until one hour after it occurs.

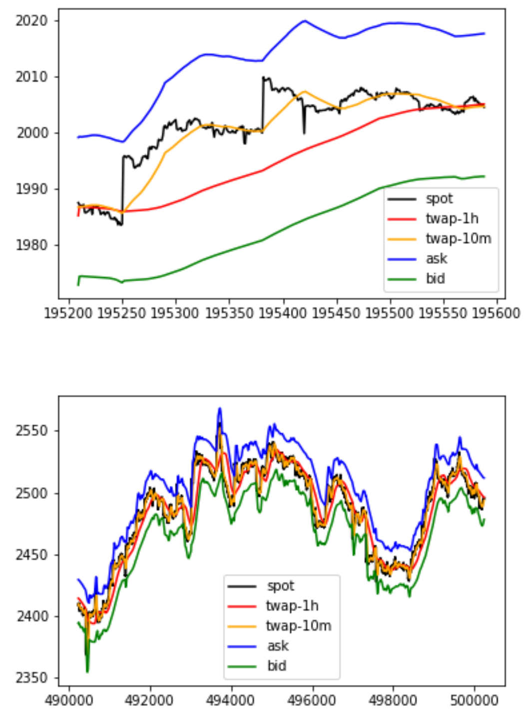


FIG. 5. Plots of the bid-ask spread around 1h and 10m rolling TWAPs against the same set of simulated data as Fig. 4. Static spread on the more extreme end of $\delta = 0.00625$ has been applied. Front-running the spot jump at $t = 195250$ is no longer profitable. Zooming out with the bottom plot to look at 2 days of simulated data shows the market remains tradeable.

volume given this trade will be

$$q(t+\tau) = \begin{cases} q(t) \cdot \left(1 - \frac{\tau}{W(t)}\right) + v_i(t+\tau), & \text{if } \tau < W(t) \\ v_i(t+\tau), & \text{otherwise} \end{cases} \tag{43}$$

The prior value of the running sum for the cumulative volume is linear decayed to account for the time elapsed since the last interaction with the market. The volume associated with the current trade $v_i(t+\tau)$ is then added to the adjusted running sum. If more time has passed than the prior calculated decay window $W(t)$ (i.e. $\tau > W(t)$), the prior value for q has decayed to zero and the new value is simply v_i. In this way, historical volume numbers are removed from the impact calculation as the TWAP catches up linearly in time.

The window over which to decay the rolling volume into the future is updated to

$$W(t+\tau) = \begin{cases} \frac{w_1 \cdot (W(t) - \tau) + w_2 \cdot W_0}{w_1 + w_2}, & \text{if } \tau < W(t) \\ W_0, & \text{otherwise} \end{cases} \tag{44}$$

where

$$w_1 = |q(t+\tau) - v_i(t+\tau)| \tag{45}$$

$$w_2 = |v_i(t+\tau)| \tag{46}$$

and $W_0 = \nu$ for market impact. The updated decay window is a weighted average of the time left in the prior window and W_0, with weights corresponding to the magnitude of the adjusted running sum value and the newly added volume, respectively.

B. Calibrating δ

Static spreads should be calibrated per market based on the risk of a jump in spot price over the ν period required for the shorter TWAP to catch up to spot. Aim to produce bid and ask values that will be worse than any jumps that are likely to occur in the spot price over ν, for a given confidence level $1-\alpha$: e.g. estimated 95% of the time, jumps in spot over a 10m interval won't overcome the static spread. Use VaR again to calibrate, and set the value at risk to the system from a trader looking to front-run over the next ν period equal to zero, with confidence $1-\alpha$: $1 - \alpha = \mathbb{P}[\text{PnL}(Q, t+\nu) \leq 0 | \mathcal{F}_t]$. Some assumptions help to simplify. Take entry and exit TWAP values

- $\text{TWAP}(t-\nu, t) \sim P(t-\nu)$
- $\text{TWAP}(t, t+\nu) \sim \text{TWAP}(t, t+\Delta) \sim P(t)$

to be similar to the spot price ν in the past, due to the timelag. The market contract has access to manipulation-resistant information only up to time $t-\nu$: $\mathcal{F}_{t-\nu}$.

Profitability of the front-run on a jump up in spot (long trade) is approximately $\text{PnL}(Q, t+\nu) \approx Q \cdot g(X_\nu - 2\delta)$, ignoring market impact. VaR for the front-run will be $\text{VaR}(\alpha, t+\nu) = Q \cdot g(F^{-1}_{X_\nu}[1-\alpha] - 2\delta_l)$, which is zero when $F^{-1}_{X_\nu}[1-\alpha] - 2\delta_l = 0$. Similar logic can be applied to the

TABLE II. Table of static spread values δ for different confidence levels $1-\alpha$ on a WETH/USDC market. Calibrations use the same `pystable` fits to Uniswap V3's WETH/USDC 0.3% pool as Table I. Shorter TWAP is averaged over a $\nu = 10$ min window.

α	0.01	0.025	0.05	0.075	0.1
δ	0.00331	0.00185	0.00123	0.00097	0.00081

TABLE III. Table of market impact constant values λ for different (α, q_0) combinations on a WETH/USDC market. $1-\alpha$ is the confidence level. q_0 is the fraction of the open interest cap above which front-running becomes negative EV, if spot jumps more than the spread δ. For every q fraction of the open interest cap taken by the trade, the trader experiences $\sim \lambda q$ in slippage. Calibrations use the same assumptions and fits as Table II, with a payoff cap of $C_P = 10$ for a max return of 10x.

(α, q_0)	0.005	0.010	0.015	0.020	0.025	0.030
0.01	1.610	0.805	0.537	0.402	0.322	0.268
0.025	0.827	0.414	0.276	0.207	0.165	0.138
0.05	0.502	0.251	0.167	0.126	0.100	0.084
0.075	0.377	0.189	0.126	0.094	0.075	0.063
0.1	0.310	0.155	0.103	0.078	0.062	0.052

spread calibration for the short trade, δ_s. Together,

$$\delta_l = \frac{1}{2} F^{-1}_{X_\nu}(1-\alpha) \tag{47}$$

$$\delta_s = -\frac{1}{2} F^{-1}_{X_\nu}(\alpha) \tag{48}$$

Set

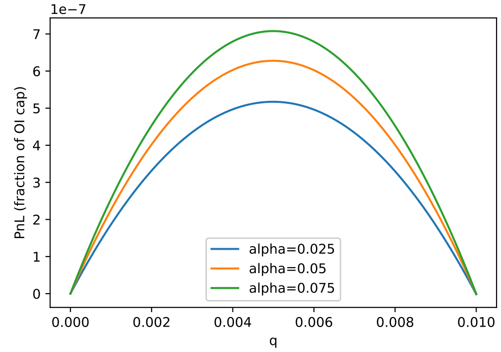
$$\delta = \max(\delta_l, \delta_s) \tag{49}$$

to be conservative against traders looking to front-run the TWAP lag. Table II gives suggested spread values for different confidence levels.

C. Calibrating λ

Impact constants should be calibrated per market based on the risk a large jump in spot price actually exceeds the static spread over the ν period required for the shorter TWAP to catch up to spot. Aim to produce price slippage that will minimize the expected shortfall (ES) for the same confidence level $1-\alpha$: e.g. estimated 5% of the time when spot jumps more than the static spread, most position sizes will have negative EV if attempting to front-run the TWAP. Use ES to calibrate, and set the expected shortfall to the system for confidence $1-\alpha$ from a trader looking to front-run over the next ν period to be less than zero for position sizes larger than q_0 fraction of the cap: $\mathbb{E}[\text{PnL}(Q, t+\tau) | \text{PnL} > 0]|_{q \geq q_0} \leq 0$.

Use the same assumptions as with the static spread, but employ the payoff cap for finite expected values.



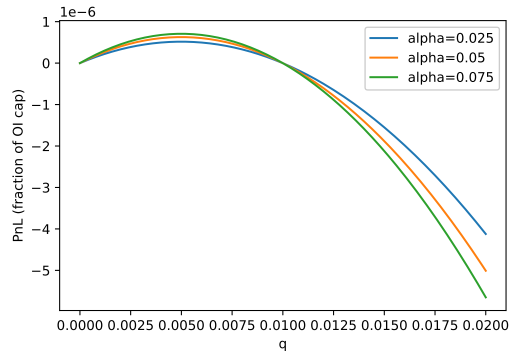


FIG. 6. Plots of the unconditional expected shortfall lost to front-running on a WETH/USDC market for different $1-\alpha$ confidence levels. $1-\alpha = [0.925, 0.95, 0.975]$ represent the three separate curves. y axis is the notional amount lost to the trader in units of fraction of open interest cap. x axis of q is the queued size of the front-running trade normalized with respect to the open interest cap. Assumed $q_0 = 0.01$. Notice the trade is expected to be unprofitable when $q > q_0$, given the market impact calibration for λ.

When including market impact, profitability of the front-run on the long trade is approximately $\text{PnL}(Q, t+\nu) \approx Q \cdot \min[g(Y_{\nu_-} - \lambda Q), C_P]$, where $Y_{\nu_\mp} \equiv X_\nu \mp 2\delta$. Conditional expected shortfall for confidence level $1-\alpha$ will be $\text{ES}(\alpha, t+\nu) = \frac{Q}{\alpha} \cdot \{e^{-2\lambda q} \int_0^\infty dy \; f_{Y_{\nu_-}}(y) \cdot \min(e^y, 1+C_P) - \alpha\}$. Short trade follows similar logic but with Y_{ν_+} and without the payoff cap C_P.

Impact constants

$$\lambda_l = \frac{1}{2q_0} \cdot \ln \rho_{\lambda_l} \tag{50}$$

$$\lambda_s = \frac{1}{2q_0} \cdot \ln \rho_{\lambda_s} \tag{51}$$

where

$$\rho_{\lambda_l} \equiv \frac{\int_0^{g^{-1}(C_P)} dy \; e^y f_{Y_{\nu_-}}(y)}{\alpha - (1+C_P) \cdot [1 - F_{Y_{\nu_-}}(g^{-1}(C_P))]} \tag{52}$$

$$\rho_{\lambda_s} \equiv \frac{\alpha}{\int_{-\infty}^0 dy \; e^y f_{Y_{\nu_+}}(y)} \tag{53}$$

should result in $\mathbb{E}[\text{PnL}(Q, t+\tau) | \text{PnL} > 0] \leq 0$ when $q \geq q_0$. $f_{Y_{\nu_\mp}}$ and $F_{Y_{\nu_\mp}}$ are, respectively, the PDF and

CDF of $Y_{\nu_\mp} \sim S(a, b, \mu\nu \mp 2\delta, \sigma \cdot (\frac{\nu}{a})^{1/a})$. Set

$$\lambda = \max(\lambda_l, \lambda_s) \tag{54}$$

to be conservative again. Table III gives suggested impact values for different confidence levels and position size cutoffs.

Fig. 6 plots the unconditional expected shortfall against q for the same market. For larger confidence levels, the peaks of ES for a given q decrease significantly. The market is susceptible to losing the area under the curve of the first plot divided by q_0, on average every ν period. This can be recovered simply through inclusion in inflation expectations when calibrating the notional cap C_Q in Section IV B.

D. Spot Manipulation

Uniswap V3 and Balancer V2 offer the geometric TWAP as a manipulation-resistant on-chain oracle. With enough capital, traders can move the spot price significantly. Overlay markets place bounds on the notional cap to render spot manipulation attacks unprofitable in most reasonable instances. Further, the market imposes bounds on what it considers to be valid price changes over the last Δ time period as a way to reject extreme price changes over the longer TWAP averaging window likely caused by manipulation of the spot pool.

1. Front-Running Attack

The purpose of using the shorter TWAP as a proxy for spot is to increase the amount of capital required for a trader to manipulate the spread in their favor after spot realizes an actual jump in price. However, market impact also guards us against the following manipulation relevant to on-chain oracles. Over several blocks, a trader could route a spot swap through a feed supported by the protocol into OV (e.g. USDC $\rightarrow$ ETH $\rightarrow$ OV). They then use the OV proceeds to immediately enter a trade on the associated Overlay market before the shorter TWAP can catch up to the spot move, essentially front-running themselves if little volume exists on the spot pool. Preventing this trade from being profitable puts an upper bound on the notional cap C_Q through our market impact λq term.

Say the front-runner takes out Q worth of 1x long open interest on the Overlay market immediately after the spot swap to bump up the price. Q is the capital swapped for on spot. The capital required to bump the Uniswap spot price up by $0 < \epsilon < \infty$ can be expressed as $Q = y \cdot [\sqrt{1+\epsilon} - 1]$, where y are the virtual reserves for `token1` in OV terms. Roughly, the PnL obtained on the Overlay leg of the front-run will be $\text{PnL} = Q \cdot [e^{-2\lambda q} \cdot (1+\epsilon) - 1]$, assuming a worst case scenario of no arbitrageurs bringing the spot pool back. For this front-run to be unprofitable, markets must impose $C_Q \leq 2\lambda y \cdot (\sqrt{1+\epsilon} - 1) / \ln(1+\epsilon)$. The bound on the cap is smallest when $\epsilon \rightarrow 0$, with $\lim_{\epsilon \rightarrow 0} \ln(1+\epsilon)/(\sqrt{1+\epsilon} - 1) = 2$. The notional cap must have an upper bound of

$$C_Q \leq \lambda y \tag{55}$$

to ensure this oracle front-run is unprofitable. The same bound is obtained for the short. Higher leverages on the Overlay leg of the front-run also produce worse market impact, so the bound is conservative. This notional cap constraint is programmed directly into Overlay markets using Uniswap V3 or Balancer V2's liquidity oracle, adjusting the cap downward in the event spot liquidity drops significantly.

2. Back-Running Attack

The purpose of using the longer TWAP in bid-ask spread pricing is to increase the amount of capital required for a trader to manipulate the underlying spot price after entering a trade on the associated Overlay market. The back-running trade works in the following way. The trader first enters a position on the Overlay market. Over several subsequent blocks, the trader swaps tokens in the spot pool to either increase or decrease the TWAP value in their favor. They finally exit their Overlay position for an attempted overall profit. Preventing this trade from being profitable puts an upper bound on the notional cap C_Q through the spread term δ.

Ignoring market impact and assuming arbitrageurs bring the spot price back to its original value each block after the manipulation is registered by the oracle accumulator, the profitability of the attack is given by the profits gained from the Overlay leg of the trade minus the slippage incurred from spot manipulation. Intuitively, the trader needs to overcome the spread δ imposed on the TWAP prices by the Overlay market as well as the associated tokens lost to slippage to move the TWAP itself. One finds an upper bound of [11]

$$C_Q \leq \Delta_B \cdot B_i \cdot \frac{w_o}{w_o + w_i} \cdot 4\delta \tag{56}$$

renders the back-running trade unprofitable in all reasonable scenarios for a Balancer V2 pool. Δ_B is the longer TWAP averaging window in number of blocks. B_i are the reserves for the token swapped into the Balancer pool in OV terms. w_i, w_o are the pool weights for the spot token swapped in and received out, respectively.

The bound is set by the minimum open interest needed for the trader to break-even on the attack in the scenario where they manipulate spot a small amount over every block for Δ_B in blocks, which requires the least amount of capital to execute the attack. For the appropriate upper bound programmed directly into Overlay market contracts, take the minimum of the expression on the right-hand side of the inequality for each Balancer pool swap possibility $i \to o$. For Uniswap V3, the expression reduces to

$$C_Q \leq \Delta_B \cdot y \cdot 2\delta \tag{57}$$

with y as the virtual reserves for `token1` in OV terms.

3. Price Change Bounds

With Uniswap V3, concentrated liquidity presents challenges in assessing the cost of attack to manipulate the spot oracle. This is due to the liquidity profile no longer being deterministic [12]. Further, liquidity values returned by the spot oracle are time-weighted averages around realized ticks over the past averaging interval – this says nothing about the liquidity profile outside of the current tick range. Meaning, the liquidity profile could drop off substantially outside of the current tick range, but Overlay markets would be unaware. An attacker could take advantage of this knowing less slippage occurs on spot than what the Overlay market anticipates from the liquidity oracle.

To limit the damage associated with an unknown liquidity profile, Overlay markets place limits on the prices they will honor from the oracle fetch. The market compares the value of the longer TWAP at the current time t with what its value was one Δ averaging period ago at time $t - \Delta$. If the magnitude of the logarithm of the TWAP has changed more than a maximum amount $\mu_{max} \cdot \Delta$, the market won't honor the price, and the trade reverts. Valid prices fetched from the oracle must satisfy

$$e^{-\mu_{\max} \cdot \Delta} \leq \frac{P_{now}}{P_{last}} \leq e^{\mu_{\max} \cdot \Delta} \tag{58}$$

$$P_{now} = \text{TWAP}(t - \Delta, t) \tag{59}$$

$$P_{last} = \text{TWAP}(t - 2 \cdot \Delta, t - \Delta) \tag{60}$$

for the trade to execute. This limits price changes the attacker can take advantage of to a maximum of $e^{\pm \mu_{\max} \cdot \Delta}$ every Δ period. The downside to this bound is traders may not be able to execute trades in periods of extreme volatility when large price swings really can occur.

This bound should be used cautiously, with only extreme circumstances triggering the revert. To calibrate $\mu_{\max}$, take the probability that the price change is within the bounds to be approximately one:

$$\begin{aligned} 1 - \alpha &= \mathbb{P}\left[\ln \left| \frac{P_{now}}{P_{last}} \right| \leq \mu_{\max} \cdot \Delta \right] \\ &\approx F_{X_\Delta}(\mu_{\max} \cdot \Delta) - F_{X_\Delta}(-\mu_{\max} \cdot \Delta) \end{aligned} \tag{61}$$

where α is very small.

Set the drift on the price bound to

$$\mu_{\max_l} = \frac{1}{\Delta} \cdot F^{-1}_{X_\Delta}\left(1 - \frac{\alpha}{2}\right) \tag{62}$$

$$\mu_{\max_s} = -\frac{1}{\Delta} \cdot F^{-1}_{X_\Delta}\left(\frac{\alpha}{2}\right) \tag{63}$$

$$\mu_{\max} = \max(\mu_{\max_l}, \mu_{\max_s}) \tag{64}$$

to be conservative. Prices should then be within the bounds approximately $1 - \alpha$ of the time (i.e. bounds hit α percent of the time).

Ideally, the spot pool has a substantial base layer of liquidity across the max tick range. However, this price bound check by the Overlay market provides an additional backstop in the event this isn't the case.

IV. CAPS

Overlay markets have two types of caps: payoff caps C_P and open interest caps $C_{\text{OI}}(t)$.

A. Payoff Caps

Payoff caps, imposed on a position's PnL, limit the damage associated with the tail behavior of the underlying data stream. The protocol is then only liable for a maximum C_P in price feed returns per position contract: price exposure transitions from $\pm g(X_\tau) \to \min[\pm g(X_\tau), C_P]$. This prevents an infinite payout on any single trade, giving the ability to quantify risk for heavier-tailed feeds. It is easiest to see how important a payoff cap per position is through the example of the ETH/OV market in Fig. 7: with the inverse market, infinite gains occur in the region $(0, 1]$. Without the payoff cap, it is possible the system prints an infinite amount of OV on only one trade if the inverse price feed declines significantly. The payoff cap eliminates this possibility entirely, deterministically setting a worst case scenario per trade for any potential future price movement. Payoff caps can be set by governance based off trader appetite, given the manner in which we derive the rest of our risk framework through $g^{-1}(C_P)$.

B. Open Interest Caps

Open interest caps, imposed on the aggregate open interest for the long and short sides, limit the amount of position contracts the Overlay market is willing to take on at any point in time. As the payoff cap applies on a per trade basis, it is useless without restricting the total amount of open interest allowed. Whenever a trader looks to build a new position, an initial check is made to ensure their queued open interest plus the existing aggregate open interest on the side of their trade does not breach the cap:

$$\text{OI}_\chi + \text{OI} \leq C_{\text{OI}}(t) \tag{65}$$

OI_χ is the existing aggregate open interest on the side the trader is looking to build a position on, with $\chi \in \{l, s\}$ for either the long or short side depending on the side of the trade. $C_{\text{OI}}(t)$ is given by the notional cap C_Q set by governance divided by the current mid price $P(t)$ as defined in Equation (40).

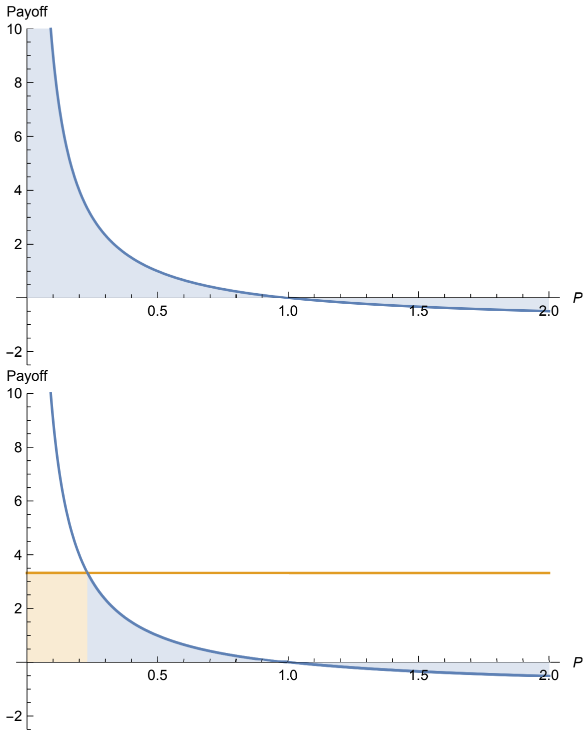


FIG. 7. Plots of the inverse market payoff with and without a cap as a function of price P, normalized with respect to the initial price at entry. P is the number of ETH per OV. Without the cap, a single position could print an infinite amount of OV. The payoff cap (orange line), however, eliminates this possibility by limiting the damage for each trade. For an ETH/OV market, Uniswap V3 price ratio used would be y/x where y is the number of OV virtual reserves in the pool and x is the number of ETH virtual reserves (i.e. $1/P$).

Notional caps (and therefore open interest caps) set expectations for the amount printed by a market due to the imbalance liability, as they constrain the largest possible value for the initial notional:

$$\mathrm{OI}(t) \cdot P(t) \leq C_Q \tag{66}$$

Take the time average of the unconditional expected shortfall of the imbalance liability

$$\frac{\alpha}{T_\alpha} \int_0^{T_\alpha} d\tau \; \mathbb{E}[\mathrm{PnL}_{liability}(t, t+\tau) | \mathrm{PnL}_{liability} > \mathrm{VaR}_\alpha] \tag{67}$$

from t to $t + T_\alpha$, where

$$\mathrm{ES}_\alpha[\mathrm{PnL}_{liability}(t, t+\tau)] = \mathbb{E}[\mathrm{PnL}_{liability}(t, t+\tau) | \mathrm{PnL}_{liability} > \mathrm{VaR}_\alpha] \tag{68}$$

The time average is the amount the market should expect to print due to times when profits exceed the value at risk from Equation (34), averaged over the anchor time period T_α. To calibrate the notional cap, assume the worst case scenario: every T_α, the imbalance is completely to one side and equal to the cap. Total notional for the imbalance liability in Equation (33) becomes $\mathrm{OI}_{tot}(t) \cdot P(t) = C_Q$. Employ the payoff cap C_P for finite expected values. On average, governance should prepare for the market to print

$$I(t, t+T_\alpha) = C_Q \cdot \alpha \cdot \Bigg\{ \frac{1 - e^{-2kT_\alpha}}{2kT_\alpha} - 1 \pm \frac{1}{T_\alpha} \int_0^{T_\alpha} d\tau \; e^{-2k\tau} \cdot \mathrm{ES}_\alpha \Big[\min \Big(g(X_\tau), C_P \Big) \Big] \Bigg\} \tag{69}$$

over the future period of time $0 \leq \tau \leq T_\alpha$. $\pm = +1$ when $\mathrm{OI}_l > \mathrm{OI}_s$ and $\pm = -1$ otherwise.

Set the notional cap to

$$C_Q = \frac{I_\alpha}{\{\}|_I} \tag{70}$$

using a governance accepted printing expectation of $I(t, t+T_\alpha) = I_\alpha$ over the future T_α period. The bracketed expression $\{\}|_I$ is shorthand for

$$\{\}|_I = \alpha \cdot \Bigg\{ \frac{1 - e^{-2kT_\alpha}}{2kT_\alpha} - 1 \pm \frac{1}{T_\alpha} \int_0^{T_\alpha} d\tau \; e^{-2k\tau} \cdot \mathrm{ES}_\alpha \Big[\min \Big(g(X_\tau), C_P \Big) \Big] \Bigg\} \tag{71}$$

Similar to δ and λ calibrations, choose the maximum of time averaged expected values between a long vs short imbalance for the bracketed term $\{\}|_I$ in the denominator.

C_Q should be calibrated based off the amount of OV governance contributors would be willing to tolerate printing per market. Table **??** gives suggested open interest caps for different anchor times T_α on a WETH/USDC market.

C. Circuit Breakers

Adding feedback to the notional caps (and therefore open interest caps) offers the ability to limit *new* position builds when printing in the prior T_α period exceeds expectations. Lowering the notional cap based off prior realized market prints acts like a circuit breaker in the event of a large payout. Together, payoff caps C_P and static notional caps C_Q limit the amount the market can print on any single trade. Circuit breakers take this a step further by limiting the amount the market can print over multiple trades in a given period of time. When an excess amount of OV has been printed in the recent past, the possible notional size of new position builds offered is significantly reduced for an extended period of time to cool down the market.

Take I_α as shorthand for the expected per-market inflation rate used to calibrate C_Q over a period of T_α. This is a static parameter representing the governance targeted printing rate for the market. In comparison, the realized rate

$$I_R(t - T_\alpha, t) = \frac{1}{T_\alpha} \int_{t'=t-T_\alpha}^{t'=t} dt' \; \mathrm{PnL}(t') \tag{72}$$

is the actual inflation rate over the last T_α period. $\int_{t-T_\alpha}^{t} dt' \ \mathrm{PnL}(t')$ is the total amount of OV the market has printed or burned over the last T_α period.

To modify the target rate based off realized prints from the recent past, use an "effective" inflation rate target I_E at the current time t

$$\begin{aligned} I_E(t - T_\alpha, t) &= I_\alpha - \bigg(I_R(t - T_\alpha, t) - I_\alpha \bigg) \\ &= 2I_\alpha - I_R(t - T_\alpha, t) \end{aligned} \tag{73}$$

This is the desired inflation rate I_α adjusted for its difference with the realized rate over the last T_α period. The effective rate increases for smaller realized profits and decreases for larger realized profits over the rolling window $[t - T_\alpha, t]$. Expressing the effective rate in terms of an effective notional cap:

$$\begin{aligned} C_Q|_E(t - T_\alpha, t) &= \frac{I_E(t - T_\alpha, t)}{\{\}|_I} \\ &= C_Q \cdot \frac{I_E(t - T_\alpha, t)}{I_\alpha} \end{aligned} \tag{74}$$

The circuit breaker on the notional cap $C_Q|_B(t-T_\alpha, t)$ is the minimum of the effective cap and the static cap C_Q:

$$C_Q|_B(t - T_\alpha, t) = C_Q \cdot \min \left[1, 2 - \frac{I_R(t - T_\alpha, t)}{I_\alpha} \right] \tag{75}$$

$I_R(t - T_\alpha, t) \geq 2I_\alpha$ brings the circuit breaker down to zero, which slowly increases back up to C_Q as the rolling window progresses from $[t - T_\alpha, t] \to [t, t + T_\alpha]$. This is the mechanism needed to prevent multiple large prints over T_α. If desired inflation expectations are actually met $I_R(t - T_\alpha, t) \leq I_\alpha$ over the prior T_α period, the notional cap remains at the C_Q value set by governance.

To implement the rolling window over which the market measures cumulative realized printing amounts, markets use the same roller algorithm as price impact from Section III A: i.e. Equations (43) and (44). However, the value added to the roller is the quantity of OV minted or burned (i.e. realized PnL) each time a position is unwound. Assume trader i has exited a trade of OI_i for a profit or loss of PnL_i at time $t + \tau$. The value added to the rolling cumulative printed sum is taken to be

$$v_i(t + \tau)|_{I_R} = \mathrm{PnL}_i(t + \tau) \tag{76}$$

$q = q_{I_R}$ is updated each unwind according to Equation (43) with $v_i = v_i|_{I_R}$. The window over which to decay the rolling cumulative printed sum is given by Equation (44) again, but with $W_0 = T_\alpha$ for circuit breaker calculations.

V. LIQUIDATIONS

Each position requires a level of maintenance to protect against bankruptcies. If the position value in (9) is less than the required maintenance amount

$$V(t, t + \tau) < \mathrm{MM} \cdot \mathrm{OI}(t) \tag{77}$$

the position can be liquidated. MM is a per-market maintenance margin constant set by governance. Losses, $V(t, t) - V(t, t + \tau)$, are burned upon triggering a liquidation on a position that does not meet the maintenance requirement. A portion of the remaining value, $\beta \cdot V(t, t + \tau)$, is burned as insurance against instances when positions are not liquidated in time. The remaining fraction, $1 - \beta$, is split between a reward for the liquidator to incentivize timely liquidations and a liquidation fee charged by the protocol.

A. Calibrating MM

MM should be calibrated per market based on the risk the position value goes below zero at a future time $t + \tau_1$, assuming the maintenance amount was breached at an earlier time $t + \tau_0$. Aim to minimize the probability the position goes negative within a period of $\omega \equiv \tau_1 - \tau_0$, with a degree of uncertainty α: $\alpha = \mathbb{P}[V(t, t + \tau_1) \leq 0 | V(t, t + \tau_0) = \mathrm{MM} \cdot \mathrm{OI}(t)]$. Assuming funding does not change open interest significantly over ω and calibrating based off larger leverage values to be safe ($L \to \infty$), one finds

$$\mathrm{MM}_l = e^{-F^{-1}_{X_\omega}(\alpha)} - 1 \tag{78}$$

$$\mathrm{MM}_s = 1 - e^{-F^{-1}_{X_\omega}(1-\alpha)} \tag{79}$$

provides the needed buffer for long and short positions. Set

$$\mathrm{MM} = \max(\mathrm{MM}_l, \mathrm{MM}_s) \tag{80}$$

to be conservative. The probability of the maintenance requirement not being enough of a buffer to prevent bankruptcy within ω is then at worst equal to α. ω should be realistically chosen based on the expected time for network participants to liquidate.

B. Calibrating β

The fraction of liquidated value to burn, β, should be calibrated per market based on the risk the value of a position actually does turn negative within ω. As insurance, burn the expected shortfall each time a liquidation occurs to compensate for the the times when no remaining value exists. If the uncertainty used in calibrating the maintenance requirement is α, the conditional expected shortfall associated with this confidence level is the expected value of the position given the position has turned negative: $\mathrm{ES}_\alpha = \mathbb{E}[V(t, t + \tau_1) | V(t, t + \tau_1) \leq 0]$. Assume still that the maintenance amount is breached at time $t + \tau_0$. The negative of the unconditional expected shortfall is what should be burned each time a liquidation occurs.

Making the same assumptions as the prior section, the unconditional ES can be written as $\text{UES}_\alpha = -\beta \cdot \text{MM} \cdot \text{OI}(t)$. Set

$$\beta = \max(\beta_l, \beta_s) \tag{81}$$

where

$$\beta_l = \alpha \cdot \left[\left(1 - \rho_{M_l}\right)\left(1 + \frac{1}{\text{MM}}\right) - 1 \right] \tag{82}$$

$$\beta_s = \alpha \cdot \left[\left(1 - \rho_{M_s}\right)\left(1 - \frac{1}{\text{MM}}\right) - 1 \right] \tag{83}$$

and

$$\rho_{M_l} \equiv \frac{1}{\alpha} \int_{-\infty}^{-\ln(1+\text{MM})} dx \; e^x f_{X_\omega}(x) \tag{84}$$

$$\rho_{M_s} \equiv \frac{1}{\alpha} \int_{-\ln(1-\text{MM})}^{g^{-1}(C_P)} dx \; e^x f_{X_\omega}(x) \tag{85}$$

Note $\alpha = \mathbb{P}[V(t, t+\tau_1) \leq 0 | V(t, t+\tau_0)]$. The form $\beta \cdot \text{MM} \cdot \text{OI}(t)$ for the burn quantity implicitly assumes most liquidations occur near the maintenance amount when liquidated in time. This seems unlikely and should be kept in mind when using the expressions in this section.

VI. CONCLUSION

We've outlined a risk framework for the Overlay protocol to attempt long-term stability of the currency supply. Through this work, readers may implement Overlay market smart contracts that enable long or short exposure to Uniswap V3 or Balancer V2 price feeds without the need to hold the underlying spot tokens. The approach should generalize beyond price feeds to any data stream driven by a random process. It seems plausible the protocol's original inflation problem can be managed probabilistically through a combination of open interest caps, payoff caps, and a funding rate between longs and shorts. With these levers, governance contributors can target a tolerated amount of OV they would be willing to print per market, tuned based off the historical properties of the spot feed. These targets determine the expected inflation rate for the currency supply.

ACKNOWLEDGMENTS

Many thanks to Daniel Wasserman for pushing to directly relate funding with risk, Robert Koschig for feedback and discussions on early versions of notes as well as the paper, Jack Sun for encouraging instant quotes through bid-ask pricing to combat front-running instead of delayed settlement, Mikhail Vladimirov for critiques and suggested revisions to the funding mechanism, and Dmitriy Berenzon for comments on addressing the broader questions of why the paper is relevant.

[1] A. Kay, Overlay (2018), overlay.market/WPv3.pdf.
[2] B. Mandelbrot, The variation of certain speculative prices, The Journal of Business **36**, 394 (1963).
[3] J. P. Nolan, Financial modeling with heavy-tailed stable distributions, WIREs Computational Statistics **6**, 45 (2014).
[4] J. P. Nolan, *Univariate Stable Distributions: Models for Heavy Tailed Data*, Springer Series in Operations Research and Financial Engineering (Springer International Publishing, 2020).
[5] pystable, pypi.org/project/pystable.
[6] J. R. del Val, F. Simmross-Wattenberg, and C. Alberola-López, libstable: Fast, Parallel, and High-Precision Computation of -Stable Distributions in R, C/C++, and MATLAB, Journal of Statistical Software, Articles **78**, 1 (2017).
[7] N. N. Taleb, Errors, robustness, and the fourth quadrant, International Journal of Forecasting **25**, 744 (2009), special section: Decision making and planning under low levels of predictability.
[8] R. J. Shiller, Measuring Asset Values for Cash Settlement in Derivative Markets: Hedonic Repeated Measures Indices and Perpetual Futures, Journal of Finance **48**, 911 (1993).
[9] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson, Uniswap v3 Core (2021), uniswap.org/whitepaper-v3.pdf.
[10] F. Martinelli and N. Mushegian, Balancer Whitepaper (2019), balancer.fi/whitepaper.pdf.
[11] M. Feldman, Note 10: Cost of Attack - Balancer V2 (2021), oips.overlay.market/notes/note-10.
[12] C. Michel, Replaying Ethereum Hacks - Rari Fuse VUSD Price Manipulation (2021), cmichel.io/replaying-ethereum-hacks-rari-fuse-vusd-price-manipulation/.