Appendix

Austin Adams Gordon Liao

May 2022

1 Fee returns

1.1 v2 fee returns

Let x_t , y_t be the amount of asset 1 and asset 2 reserves in a constant product function market making (CPFMM) liquidity pool at time t. Let p_t be the marginal price of token y in units of token x, $p_t = \frac{y_t}{x_t}$. Let k_t denote the liquidity that exists in the pool at time t.The constant product formula, $x_t y_t = k_t$ is held in every period. That is, $x_t = k_t/y_t = \sqrt{k_t/p_t}$ and $y_t = \sqrt{k_t p_t}$. k_t can only be changed via accrual of return fees absent of addition and subtraction of reserves through liquidity pool mints and burns.

Let $v_t \equiv v(k_t, p_t)$ be the portfolio value of the liquidity pool expressed in asset 1 as the numeraire at time t. When expressed in terms of the amount of reserve asset 1 and 2, the portfolio value is $v(k_t, p_t) = x_t + y_t p_t^{-1}$. Applying the CPFMM formula, the liquidity pool value can be expressed as

$$v_t \equiv v(k_{t,} p_t) = \sqrt{\frac{k_t}{p_t}} + \sqrt{k_t p_t} p_t^{-1}.$$

Absent of mints and burns, the simple return on a liquidity pool due to fee accrual is

$$r_{t+1}^{fee} = \frac{v_{t+1} - v(k_t, p_{t+1})}{v_t}.$$

Note that $\frac{v(k_t, p_{t+1})}{v_t}$ is the gross return of the liquidity pool portfolio value without fee accrual. Minting and burning of liquidity pool positions can be accounted for as follows. The evolution of liquidity value follows $k_{t+1} = k_t + \kappa_{t+1} + \phi_{t+1}$, where κ_{t+1} is the net mints minus burns that occurs in between t and $t+1$ and ϕ_{t+1} is the fee accrual due to transactions in this period, i.e. $\phi_{t+1} = \sum_i \lambda |s_i|$ $\forall$ swap s_i that occurs between $t$and $t+1$ and fee tier λ. Let $k'_{t+1} \equiv k_{t+1} - \kappa_{t+1}$. Thus, we have the mint-burn adjusted simple fee return on a liquidity pool

$$r_{t+1}^{fee} = \frac{v\left(k'_{t+1}, p_{t+1}\right) - v(k_t, p_{t+1})}{v(k_t, p_t)}.$$

1.2 v3 returns

Uniswap v3 added several pool, tick, and positional indexed state values. The following methodology follows the descriptions in the v3 white paper.

1.2.1 Position- and tick- indexed state

Each position has three state values, two associated with upper and lower tick and a liquidity value l. The liquidity value denotes the virtual liquidity held by the position. The two ticks are the lower tick - i_l - and the upper tick - i_u. These ticks both have two values associated with them - feeGrowthOutside0X128 - $f_{o,0}$ - and feeGrowthOutside1X128 - $f_{o,1}$.

The ticks are determined by the user when they create a position. The tick values $f_{o,0}$ and $f_{o,1}$ track how many fees were accumulated within a certain range.

1.2.2 Pool-indexed state

Pools also track feeGrowthGlobal0X128, $f_{g,0}$, and feeGrowthGlobal1X128 - $f_{g,1}$. These two values track the total amount of fees collected per unit of virtual liquidity l.

1.2.3 General fee calculations

Unclaimed fees are calculate as follows. We drop the subscript indicating token 1 and 2 since the formula for the two are the same.

Cumulative fees per share $f_r(t)$ in the range between two ticks i_l and i_u is

$$f_r(t) = f_{g,t} - f_{b,t}(i_l) - f_{a,t}(i_u)$$

where i_c is the current tick state of the pool, $f_a(i)$ and $f_b(i)$ are defined as

$$f_a(i) = \begin{cases} f_g - f_o(i) & i_c \geq i \\ f_o(i) & i_c < i \end{cases}$$

$$f_b(i) = \begin{cases} f_o(i) & i_c \geq i \\ f_g - f_o(i) & i_c < i \end{cases}$$

The uncollected fees can be defined as

$$f_u(t_1, t_0) = l(f_r(t_1) - f_r(t_0)$$

where t_1 is the current time and t_0 is the time the position was opened. $f_r(t_0)$ can either be calculated from the pool state or from feeGrowthInside value stored in the position state.

1.3 Full-range fee Calculations

For full-range positions, i_c is always between i_l and i_u, i.e. $i_l \leq i_c \leq i_u \forall i_c$

The fee calculation simplifies to

$$f_r(t) = f_{g,t} - f_{o,t}(i_l) - f_{o,t}(i_u)$$

However, $f_{o,t}(i_l)$ and $f_{o,t}(i_u)$ are static values, i.e., $f_{o,t}(i_l) = 0$ and $f_{o,t}(i_u) = c$. The constant c is the value of $f_r(t)$ when tick i_u is initialized and drops out. Thus uncollected full range fee is

$$f_u(t_1, t_0) = l(f_{g,t_1} - f_{g,t_0}).$$

$f_{g,t}$ is the fee growth of one unit of liquidity l since time 0 and $f_{g,t_1} - f_{g,t_0}$ gives the fee growth of one unit of liquidity from t_0 to t_1. The fee return on a full-range $v3$ positions is

$$r_{t+1}^{fee} = l \frac{f_{g,t_{1+1}} - f_{g,t}}{v_t}$$

where v_t is the portfolio value of the full-range liquidity position at time t defined similarly as that in $v2$.[1]

2 Data Methodology

We use data from Uniswap v2 and v3 subgraphs. The data sample is constructed as follows. First, we pull the top 500 pools by volume from Uniswap v3 and remove any pool that has no Uniswap v2 counterpart in the sample. Since multiple v3 pools may use the same v2 pool as Uniswap v3 implemented multiple fee-tiers, we choose the v3 pool with the highest average TVL over the sample period if there are multiple v3 pools (choosing the pool with the highest volume yields a similar sample).

To calculate return on pegged token pairs, we empirically assess the tick range that bounds minor price fluctuations in the pegged asset pairs. Specifically, we winsorize historical peg token prices at the 0.5% threshold symmetrically (capturing 99% of the historical prices). With the observed historical price range, we create synthetic positions on the pegged pairs. This methodology takes a conservative approach in creating liquidity positions that do not have the need to rebalance.

To avoid extreme outliers as a result of small pool size on any particular date, we impute the returns with 0 if the total value locked for a given pool on an observation date is less than \$1000. We chose this data cleaning procedure for two reasons. First, large returns for small TVL pools could not be realized. As we assume that additional liquidity deployed into the pool is marginal. For pools

[1] Equation 2.2 in the v3 white paper has the real reserve of a position defined by the curve $\left(x + \frac{L}{\sqrt{p_b}}\right)(y + L\sqrt{p_a}) = L^2$. In full-range positions, $p_a \to 0$ and $p_b \to \infty$, this indifference curve becomes identical to $xy = L^2 \equiv k$.

with TVL less than \$1000, absolute returns are generally small (e.g. fraction of a cent) despite non-zero returns when expressed as a percentage. Additionally, for small TVL pools, large capital deployment would impact the behavior of other participants in the pool and thus contradict our marginal liquidity assumption.