.

Internal Fees For Concentrated Liquidity

Dr.Nick
0xcacti
November 2023

This document outlines an algorithm for taking fees in a concentrated liquidity AMM and storing them *internally* as added pool liquidity (analogous to how it is typically done in a constant product pool). Most importantly, the algorithm is designed to maintain mathematical cohesion with respect to the structure of concentrated liquidity (which is non-trivial requirement). This document is divided into five sections:

- **Section 1: The Idea -** Here we describe the main idea of how exactly our proposed algorithm works, but at the highest level, using only pictures and no mathematics.
- **Section 2: The Algorithm -** In this section we describe the algorithm more concretely, outlining the steps and collecting all the necessary formulas
- **Section 3: The Mathematics -** Next, we present a thorough mathematical derivation for the (otherwise opaque) formulas presented in Section 2.
- **Section 4: Simulations -** In this section, we present some results from numerical simulations, comparing the LP experience under internal and external fees structures.
- **Section 5: Extra Math -** In this final section, we give an alternative (and unnecessary) mathematical derivation of the previous results.

In a constant product AMM, swap fees are stored *internally* in the pool as extra liquidity. We illustrate this below by supposing an amount Δx is paid *into* the pool. For some $\gamma < 1$, we use the modified amount $\gamma\Delta x$ to compute the outgoing amount Δy according to the constant product rule $xy = L^2$. Upon depositing the *entire* Δx into the pool, we land *above* the constant product curve, corresponding to a slightly larger value L.

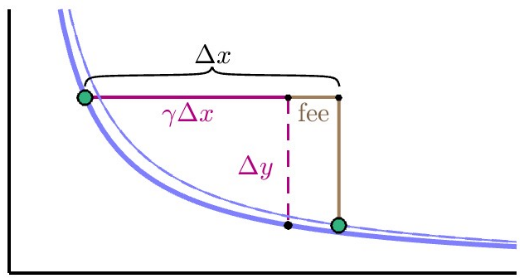

Figure 1. Storing fees internally for constant product

In a concentrated liquidity pool, each LP chooses a price range $[p_a, p_b]$ and a liquidity level L. This partitions the price axis into a sequence of *active ticks*. In any given subrange between two consecutive active ticks $[p_k, p_{k+1}]$, the liquidity available is the sum of all L terms coming from LPs with positions overlapping with this range:

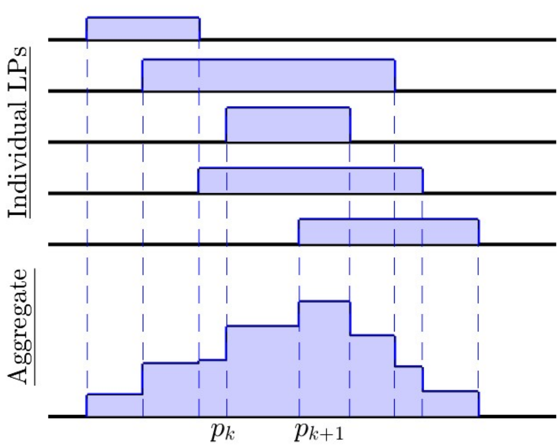

Figure 2. Aggregate liquidity from individual LPs

In each subrange, swaps occur over a curve *segment*, illustrated in figure 3. As the figure shows, one could imagine storing the fees internally in this context as well.

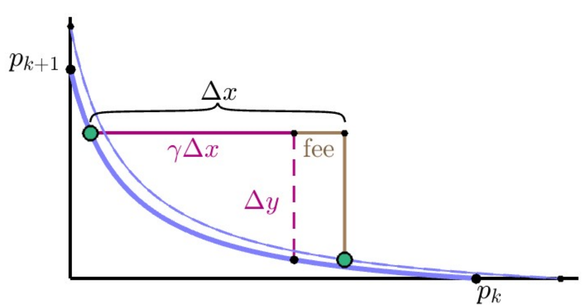

Figure 3. Storing fees internally for concentrated liquidity

As a consequence of the swap, this curve segment must be modified to correspond to a slightly larger value of L (just as it would in a constant product AMM). This requires an update to the LP positions for any LP overlapping with this range (which we will henceforth refer to as *relevant* LPs). However, if we naively increase the liquidity value *only in this particular range*, then this would require a complicated rewriting of LP positions to contain multiple superimposed blocks. This is illustrated in figure 4 below:

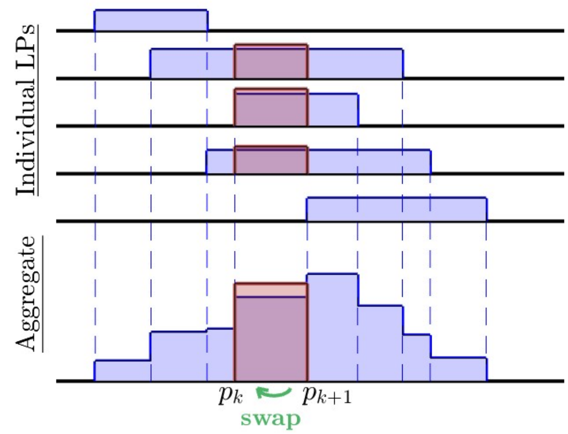

Figure 4. Rewriting LP liquidities (wrong way)

This would be impractical and computationally costly (after a period of trading, LP positions would become enormously complicated). Instead, the wiser way update the liquidity positions for the relevant LPs would be to simply scale each L value uniformly by the same factor, without otherwise altering their price position $[p_a, p_b]$. Notably, this will effect liquidity *outside* of $[p_k, p_{k+1}]$, as illustrated in figure 5 below:

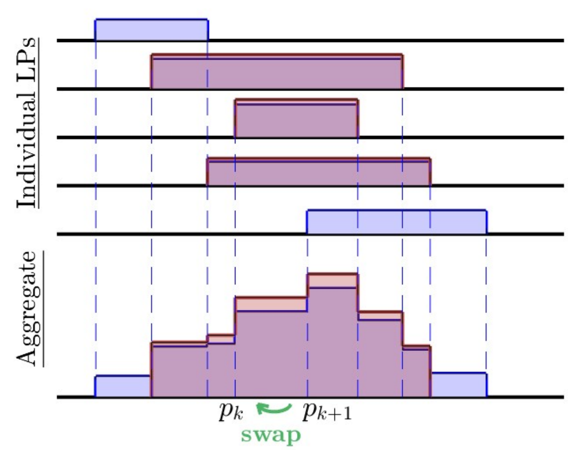

Figure 5. Rewriting LP liquidities (right way)

This avoids the escalation of increasingly complicated LP positions. However, the scale factor must be chosen in a **very precise way**, in order to maintain the strict mathematical structure of concentrating liquidity. The formulas are presented in the Section 2, but their derivations are saved for Section 3 (The Math).

Now we outline the algorithm (at a high level) for achieving the fee structure described in Section 1. Similar to the algorithm already in use for concentrated liquidity, a user specifies an amount of tokens to pay into the pool, and we iterate over active tick ranges for as long as it takes to use up this incoming amount:

(1) - user specifies an incoming token amount, pass this value to the variable *amountRemaining*

(2) - **while** *amountRemaining* > 0

(3) - calculate maximum possible amount for incoming token in current tick range

(4) - calculate the target price, the corresponding incoming/outgoing amounts for this iteration, and the liquidity scaling factor

(5) - deduct from *amountRemaining* and add to *amountGoingOut*

(6) - scale up all relevant LP liquidities

(7) - update pool token reserves

(8) - pay the user *amountGoingOut*

Most of the steps listed above are no different from the current standard concentrated liquidity swap algorithm. There are a few important differences to highlight, however, along with their advantages and disadvantages:

- The calculations in (3) and (4) are more involved than in the current standard. Specifically, both (3) and (4) require a square root, and thus we require **two square roots per iteration**.
- Scaling the liquidity for the relevant LPs in step (6) requires one multiplication for each relevant LP.
- We now have **no fee variables** (and thus neither the storage nor arithmetic associated with them)

In order to articulate these steps in more detail, we first define a few quantities below (more detail in Section 3).

- We will use the superscript n to refer to the n^{th} LP. Each LP chooses a liquidity position that is defined by the parameters $\{p_a^n, p_b^n, L^n\}$.
- The collection of positions $\{(p_a^n, p_b^n)\}$ (over all n) can be arranged in ascending order, giving us the complete collection of active ticks $\{p_k\}$.
- For the range $[p_k, p_{k+1}]$, the set N_k of *relevant* LPs is defined by $N_k = \{n \mid [p_k, p_{k+1}] \subseteq [p_a^n, p_b^n]\}$
- Then define the following three quantities:

$$L := \sum\nolimits_{n\in N_k} L^n$$
$$L^+ := \sum\nolimits_{n\in N_k} L^n \sqrt{p_a^n}$$
$$L^- := \sum\nolimits_{n\in N_k} L^n / \sqrt{p_b^n}$$

We note that while the quantities $\{L, L^+, L^-\}$ change from one tick range to the next, they need not be calculated from scratch each time we cross a tick, but only slightly modified (the variable L is already used and calculated this way in V3, referred to as '*liquidity*').

Now, the incoming and outgoing quantities may be Δx or Δy, depending on the direction of the trade. In order to deal with both cases at once, we denote the incoming and outgoing deltas by Δt_{in} and Δt_{out}. Moreover, our formulas contain $\pm$ and $\mp$ symbols that depend on the trade direction. These should be read as:

direction:	price goes down	price goes up
$(\Delta t_{in}, \Delta t_{out})$	$(\Delta x, \Delta y)$	$(\Delta y, \Delta x)$
$(\pm, \mp)$	$(+, -)$	$(-, +)$

(where price is the conventional $\Delta y/\Delta x$). With these conventions, we can now list the mathematical formulas.

First, in step (3) of the sequence outlined previously, we need to calculate the maximum amount M we may pay into the current tick range. Let p be the current pool price and let p_{edge} be the nearest active tick in the direction of the trade. Moreover, let γ be the standard fee parameter. Then we must calculate the following:

$$a_1 := \big(L\sqrt{p_{\text{edge}}}^{\mp} - L^{\mp}\big)/\big(L\sqrt{p_{\text{edge}}}^{\pm} - L^{\pm}\big)$$
$$a_2 := \big(L\sqrt{p}^{\mp} - L^{\mp}\big) - a_1\big(L\sqrt{p}^{\pm} - L^{\pm}\big)$$
$$a_3 := L\sqrt{p}^{\mp 1}(1/\gamma)$$
$$a_4 := a_1\, L\sqrt{p_{\text{edge}}}^{\pm}$$
$$a_5 := -(a_2 + a_3 + a_4)/2$$
$$a_6 := a_2\, a_3$$
$$M := a_5 + \sqrt{(a_5)^2 - a_6}$$

(We use a convention that $\sqrt{p}^{\pm} := (\sqrt{p})^{\pm 1}$.) This value M is the maximum amount that the current tick range can accommodate , i.e. we must have $\Delta t_{in} \leq M$. Using this value, we can determine if our trade will go to the *edge* of the current range, or terminate *within* it. Note also that price is stored as the square root price, so the only real square root to be taken above is in the last line.

Next, we outline the calculations required in step (4). Having determined our incoming quantity Δt_{in} from the previous step, we calculate the outgoing quantity via

$$\Delta t_{out} = L\sqrt{p}^{\pm}(\gamma \Delta t_{in})/\big(L\sqrt{p}^{\mp} + (\gamma \Delta t_{in})\big)$$

Then we calculate the new price by

$$b_1 := L\sqrt{p}^{\pm} - L^{\pm} - \Delta t_o$$
$$b_2 := L\sqrt{p}^{\mp} - L^{\mp} + \Delta t_i$$
$$b_3 := b_1/b_2$$
$$b_4 := \big(L^{\pm} - L^{\mp} b_3\big)/(2L)$$
$$\sqrt{p}^{\pm}_{new} = b_4 + \sqrt{(b_4)^2 + b_3}$$

Lastly, the liquidity scaling factor η is calculated by

$$\eta = b_1/\big(L\sqrt{p}^{\mp}_{new} - L^{\mp}\big)$$

Part 1: Setting

We begin by establishing our framework/notation for a concentrated liquidity AMM. Starting with the constant product AMM with bonding curve $xy = L^2$, the token reserves x and y at current price p are given by

$$x(p) = L/\sqrt{p} \tag{1}$$
$$y(p) = L\sqrt{p} \tag{2}$$

For concentrated liquidity, each LP chooses a liquidity position consisting of a price range $[p_a^n, p_b^n]$ and a liquidity scale factor L^n (the superscript n refers to the n^{th} LP). For prices $p \in [p_a^n, p_b^n]$, their token reserves are given by shifting expression (1)-(2) back to the coordinate axes:

$$x^n(p) = L^n\big(1/\sqrt{p} - 1/\sqrt{p_b^n}\big) \tag{3}$$
$$y^n(p) = L^n\big(\sqrt{p} - \sqrt{p_a^n}\big) \tag{4}$$

If the price hits the upper end of the range p_b^n, we have $x^n = 0$ and $y^n = L^n\big(\sqrt{p_b^n} - \sqrt{p_a^n}\big)$. Beyond this point, the assets cannot change. A similar thing happens if the price moves below the range as well. We can incorporate this fact by defining the notation $[p]^n$ as follows:

$$[p]^n := \left\{ \begin{array}{ll} p_a^n & p < [p_a^n, p_b^n] \\ p & p \in [p_a^n, p_b^n] \\ p_b^n & p > [p_a^n, p_b^n] \end{array} \right\} \tag{5}$$

Then (3)-(4) can be written again,

$$x^n(p) = L^n\big(1/\sqrt{[p]^n} - 1/\sqrt{p_b^n}\big) \tag{6}$$
$$y^n(p) = L^n\big(\sqrt{[p]^n} - \sqrt{p_a^n}\big) \tag{7}$$

but (6)-(7) is now valid for any price p. Furthermore, for a swap moving from price p to p', we have the deltas:

$$\begin{aligned} \Delta x^n &= x^n(p') - x^n(p) \\ &= L^n\big(1/\sqrt{[p']^n} - 1/\sqrt{[p]^n}\big) \end{aligned} \tag{8}$$
$$\begin{aligned} \Delta y^n &= y^n(p') - y^n(p) \\ &= L^n\big(\sqrt{[p']^n} - \sqrt{[p]^n}\big) \end{aligned} \tag{9}$$

The total liquidity available the pool, x and y, are given by summing over all LPs:

$$\begin{aligned} x(p) &= \textstyle\sum_n x^n(p) \\ &= \textstyle\sum_n L^n\big(1/\sqrt{[p]^n} - 1/\sqrt{p_b^n}\big) \end{aligned} \tag{10}$$
$$\begin{aligned} y(p) &= \textstyle\sum_n y^n(p) \\ &= \textstyle\sum_n L^n\big(\sqrt{[p]^n} - \sqrt{p_a^n}\big) \end{aligned} \tag{11}$$

Similarly, for a swap moving from price p to p', we find the deltas by summing over all LPs as well:

$$\Delta x = \textstyle\sum_n \Delta x^n \tag{12}$$
$$\Delta y = \textstyle\sum_n \Delta y^n \tag{13}$$

Part 2: Token In/Token Out amounts

If the set $\{(p_a^n, p_b^n)\}$ (over all n) is arranged in ascending order, we get the complete sequence of active ticks $\{p_k\}$ (the subscript k will always refer to the k^{th} active tick in this ordering). For each interval between adjacent active ticks $[p_k, p_{k+1}]$, we define the set of *relevant LPs* to be the LPs whose liquidity range overlaps with $[p_k, p_{k+1}]$. Their indices comprise the set N_k:

$$N_k := \{\ n \ \mid\ [p_k, p_{k+1}] \subseteq [p_a^n, p_b^n]\ \} \tag{14}$$

The total liquidity available for a trade within $[p_k, p_{k+1}]$ is L_k, defined by summing L^n over all relevant LPs:

$$L_k := \sum_{n \in N_k} L^n \tag{15}$$

Now consider a trade going from a price p to p', entirely contained in some range $p, p' \in [p_k, p_{k+1}]$. Consider, for example, the change Δy given in (13). We can break this up as follows:

$$\Delta y = \sum_{n \in N_k} \Delta y^n + \sum_{n \notin N_k} \Delta y^n \tag{16}$$

For any $n \notin N_k$, we will have $[p]^n = [p']^n$ (because both quantities will be equal to either p_a^n or p_b^n). Thus, we can see from (9) that $\Delta y^n = 0$ for $n \notin N_k$, which leaves us

$$\Delta y = \sum_{n \in N_k} \Delta y^n = \sum_{n \in N_k} L^n\big(\sqrt{p'} - \sqrt{p}\big) = L_k\big(\sqrt{p'} - \sqrt{p}\big) \tag{17}$$

In the same way, one finds

$$\Delta x = L_k\big(1/\sqrt{p'} - 1/\sqrt{p}\big) \tag{18}$$

By eliminating the variable p' from equations (17)-(18), one finds relationships between the deltas that are familiar from constant product:

$$\Delta y = -\frac{L\sqrt{p}\,\Delta x}{L/\sqrt{p} + \Delta x}, \qquad \Delta x = -\frac{L/\sqrt{p}\,\Delta y}{L\sqrt{p} + \Delta y} \tag{19}$$

Note that the deltas always have opposite signs.

Now, for concreteness, let us specifically consider a trade such that the Δx is the incoming quantity. Suppose we have a fee parameter γ (in the standard sense that 1-γ is the fraction of the payment taken as a fee). Then the outgoing quantity Δy would be calculated using the modified input $(\gamma\Delta x)$ as follows:

$$\Delta y = \frac{L\sqrt{p}\,(\gamma\Delta x)}{L/\sqrt{p} + (\gamma\Delta x)} \tag{20}$$

Note we will also take Δy to be positive, which we will account for momentarily with a subtraction. Specifically, the token quantities x and y will update according to:

$$x + \Delta x \to x \tag{21}$$
$$y - \Delta y \to y \tag{22}$$

Part 3: Calculating new price & scale factor

Now, the main idea that we proposed in Section 1 was the notion that the change (21)-(22) results in a *scaling* of the available liquidity L_k. We denote this scale factor by the letter η, illustrated in figure 6 below:

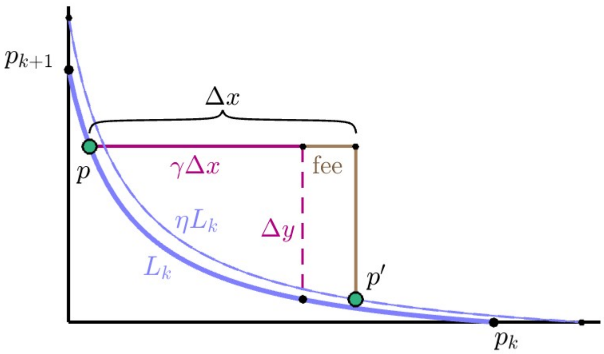

Figure 6. Scaling by eta

As we indicate in the figure above, we begin at price p and end at some new price p'. We would like our update formulas (21)-(22) to take the following form:

$$x(p) + \Delta x = \underbrace{x(p')}_{\text{updated } L} \tag{23}$$

$$y(p) - \Delta y = \underbrace{y(p')}_{\text{updated } L} \tag{24}$$

Of course, as the liquidity is scaled, only the *relevant* LPs should benefit from this scaling (as we illustrated in figure 5). For whatever η ends up being, we define η^n by

$$\eta^n := \left\{ \begin{array}{cc} \eta & n \in N_k \\ 1 & n \notin N_k \end{array} \right\} \tag{25}$$

Then using (10)-(11), we can express the notion given in (23)-(24) by writing the following:

$$\sum\nolimits_n x^n(p) \; + \Delta x = \sum\nolimits_n \eta^n x^n(p') \tag{26}$$

$$\sum\nolimits_n y^n(p) \; - \Delta y = \sum\nolimits_n \eta^n y^n(p') \tag{27}$$

As we previously observed, for two prices p and p', both of which are in the range $[p_k, p_{k+1}]$, and for any $n \notin L_k$, then we will have $[p]^n = [p']^n$. But from the definitions in (6)-(7), we can conclude further that the token amounts will also be unchanged between the two prices:

$$\big\{x^n(p), y^n(p)\big\} = \big\{x^n(p'), y^n(p')\big\} \quad (n \notin L_k) \tag{28}$$

This is simply the obvious fact that LP assets don't change when the price moves outside of their chosen range. Combined with the definition of η^n in (25), then we can see that all $n \notin N_k$ terms will cancel from both sides of (26)-(27), and so we may sum only over $n \in N_k$:

$$\sum_{n \in N_k} x^n(p) \; + \Delta x = \sum_{n \in N_k} \eta\, x^n(p') \tag{29}$$

$$\sum_{n \in N_k} y^n(p) \; - \Delta y = \sum_{n \in N_k} \eta\, y^n(p') \tag{30}$$

To simplify things, we define the following quantities:

$$L_k^+ := \sum_{n \in L_k} L^n \sqrt{p_a^n} \tag{31}$$

$$L_k^- := \sum_{n \in L_k} L^n / \sqrt{p_b^n} \tag{32}$$

With these we can simplify our expressions in (29)-(30):

$$\begin{aligned} \sum_{n \in N_k} x^n(p) &= \sum_{n \in N_k} L^n \big(1/\sqrt{p} - 1/\sqrt{p_b^n}\big) \\ &= L_k/\sqrt{p} - L_k^- \end{aligned} \tag{33}$$

$$\begin{aligned} \sum_{n \in N_k} y^n(p) &= \sum_{n \in N_k} L^n \big(\sqrt{p} - \sqrt{p_a^n}\big) \\ &= L_k \sqrt{p} - L_k^+ \end{aligned} \tag{34}$$

Substituting these into (29)-(30), we find the following:

$$L_k/\sqrt{p} - L_k^- + \Delta x = \eta \Big[L_k/\sqrt{p'} - L_k^- \Big] \tag{35}$$

$$L_k\sqrt{p} - L_k^+ - \Delta y = \eta \Big[L_k\sqrt{p'} - L_k^+ \Big] \tag{36}$$

Assuming that Δx is given (and Δy is thus determined), then two equations (35)-(36) contain only two unknowns; the new price p' and the scale factor η. Thus, we can solve for both of them. Start by defining the following:

$$b_1 := L_k\sqrt{p} - L_k^+ - \Delta y \tag{37}$$

$$b_2 := L_k/\sqrt{p} - L_k^- + \Delta x \tag{38}$$

$$b_3 := b_1/b_2 \tag{39}$$

Then we can eliminate η by dividing (36) by (35) to find

$$b_3 = \frac{L_k\sqrt{p'} - L_k^+}{L_k/\sqrt{p'} - L_k^-} \tag{40}$$

This is rearranged into a quadratic expression for $\sqrt{p'}$:

$$0 = \big(\sqrt{p'}\big)^2 + \left[\frac{b_3 L_k^- - L_k^+}{L_k}\right] \sqrt{p'} - b_3 \tag{41}$$

If we then define

$$b_4 := -\frac{b_3 L^- - L_k^+}{2L_k}, \tag{42}$$

the solution to (41) is given by

$$\sqrt{p'} = b_4 + \sqrt{(b_4)^2 + b^3} \tag{43}$$

(One can check that, because $b_3 > 0$, we *must* take the positive square root in (43) instead of the negative square root, to guarantee the new price is positive.)

Finally, having found p', we can now solve for η. For example, (36) gives us

$$\eta = \frac{b_1}{L_k\sqrt{p'} - L_k^+} \tag{44}$$

which we use to update the relevant LP liquidities:

$$L^n \to \eta\, L^n \quad (n \in L_k) \tag{45}$$

Part 4: Calculating maximum movement in range

We can use these same developments to find the maximum value M of incoming Δx that can be traded into the range $[p_k, p_{k+1}]$. Because we are considering the case of Δx as the incoming quantity, then we are driving the price *down*, so our price boundary is the lower limit p_k. Let us ask the question 'which Δx would take us to p_k'? To solve for this, we take our expression (43) for the resulting price and set it equal to p_k:

$$\sqrt{p_k} = b_4 + \sqrt{(b_4)^2 + b_3} \tag{46}$$

There is an incoming amount Δx implicit in (46), and our job now is to solve for it. By subtracting b_4, squaring, and simplifying, we find

$$p_k - 2b_4\sqrt{p_k} = b_3 \tag{47}$$

Substituting in the expression (42) for b_4, we get

$$p_k + \left(\frac{b_3 L^- - L_k^+}{L_k}\right)\sqrt{p_k} = b_3 \tag{48}$$

Solving (48) for b_3, one finds

$$\frac{L_k\sqrt{p_k} - L_k^+}{L_k/\sqrt{p_k} - L_k^-} = b_3 \tag{49}$$

Using definitions (37)-(39), we can explicitly write b_3 as

$$\frac{L_k\sqrt{p_k} - L_k^+}{L_k/\sqrt{p_k} - L_k^-} = \frac{L_k\sqrt{p} - L_k^+ - \Delta y}{L_k/\sqrt{p} - L_k^- + \Delta x} \tag{50}$$

Recalling that our goal is to solve for Δx, we see that it appears of the right side of (50) explicitly in the denominator and implicitly in the numerator (in Δy). Let's first give a name to (the reciprocal of) the quantity on the left side,

$$a_1 := \frac{L_k/\sqrt{p_k} - L_k^-}{L_k\sqrt{p_k} - L_k^+} \tag{51}$$

so that (50) may be written

$$1/a_1 = \frac{L_k\sqrt{p} - L_k^+ - \Delta y}{L_k/\sqrt{p} - L_k^- + \Delta x} \tag{52}$$

Equation (52) can be rearranged into

$$\left(L_k/\sqrt{p} - L_k^-\right) - a_1\left(L_k\sqrt{p} - L_k^+\right) + \Delta x = -a_1\Delta y \tag{53}$$

To clean up (53), we define

$$a_2 := \left(L_k/\sqrt{p} - L_k^-\right) - a_1\left(L_k\sqrt{p} - L_k^+\right) \tag{54}$$

so that (53) becomes

$$a_2 + \Delta x = -a_1\Delta y \tag{55}$$

Now we restore the Δx dependence in Δy from (20):

$$a_2 + \Delta x = -a_1\left(\frac{L\sqrt{p}\,(\gamma\Delta x)}{L/\sqrt{p} + (\gamma\Delta x)}\right) \tag{56}$$

Equation (56) can be rearranged into a quadratic expression in Δx:

$$(\Delta x)^2 + \left[a_2 + \frac{L_k}{\sqrt{p}\,\gamma} + a_1 L_k\sqrt{p}\right]\Delta x + \left[\frac{a_2 L_k}{\sqrt{p}\,\gamma}\right] = 0 \tag{57}$$

If we define the quantities

$$a_3 := L_k/(\sqrt{p}\,\gamma) \tag{58}$$
$$a_4 := a_1\, L_k\sqrt{p} \tag{59}$$
$$a_5 := -(a_2 + a_3 + a_4)/2 \tag{60}$$
$$a_6 := a_2\, a_3 \tag{61}$$

then the solution to (57) can be expressed as

$$\Delta x = a_5 + \sqrt{(a_5)^2 - a_6} \tag{62}$$

We may take this as our maximum incoming amount M:

$$M := a_5 + \sqrt{(a_5)^2 - a_6} \tag{63}$$

Part 5: Generalizing to trade in either direction

Formulas (43)-(45) and (63) (and their dependencies) are all of the final formulas that we need to execute swap for an incoming Δx. If, on the other hand, it was the quantity Δy that was incoming, then the outgoing Δx would be computed in a way analogous to (20):

$$\Delta y = \frac{L\sqrt{p}\,(\gamma\Delta x)}{L/\sqrt{p} + (\gamma\Delta x)} \tag{64}$$

The resulting set of equations analogous to (35)-(36) that relate the incoming/outgoing amounts to the new price and scaling factor would be

$$L_k/\sqrt{p} - L_k^- - \Delta x = \eta\left[L_k/\sqrt{p'} - L_k^-\right] \tag{65}$$
$$L_k\sqrt{p} - L_k^+ + \Delta y = \eta\left[L_k\sqrt{p'} - L_k^+\right] \tag{66}$$

One notices that this set of equations is identical to the original pair (35)-(36) if we simply do the following:

- switch the roles of $\Delta x \leftrightarrow \Delta y$,
- switch the roles of $L_k^- \leftrightarrow L_k^+$, and
- take the reciprocal of all prices $\sqrt{p} \leftrightarrow 1/\sqrt{p}$

Thus, we can immediately guess the analogous final expressions for $\sqrt{p'}$, η or M. In order to write down both cases simultaneously, we will be general and use the symbols Δt_{in} and Δt_{out} for the incoming and outgoing tokens, respectively. Additionally, we will use either $\pm$ and $\mp$ as needed, all summarized in the following table:

direction:	price goes down	price goes up
Δt_{in}	Δx	Δy
Δt_{out}	Δy	Δx
$\pm$	$+$	$-$
$\mp$	$-$	$+$

With these conventions, then we can say, for example, that for a given incoming quantity Δt_{in}, we compute the outgoing by Δt_{out} according to

$$\Delta t_{out} = \frac{L_k \sqrt{p}^{\pm}(\gamma \Delta t_{in})}{L_k \sqrt{p}^{\mp} + (\gamma \Delta t_{in})} \tag{67}$$

Note, the $\pm$ appearing in the exponent of $\sqrt{p}$ is shorthand for ± 1 (this convention helps for readability). One can easily confirm that (67) reproduces either (20) or (64), depending on the trade direction. More generally, one can see that we have an effective 'recipe' for generalizing our previous formulas for to handle either case, as follows:

$$\sqrt{p} \to \sqrt{p}^{\pm} \tag{68}$$
$$1/\sqrt{p} \to \sqrt{p}^{\mp} \tag{69}$$
$$L_k^{+} \to L_k^{\pm} \tag{70}$$
$$L_k^{-} \to L_k^{\mp} \tag{71}$$

Lastly, we use p_{edge} for the price at the edge of the interval in the direction of trade (either p_k or p_{k+1}). We can now restate the calculation for M. We collect definitions (51), (54), (58)-(63), and apply the recipe. We find:

$$a_1 := \frac{L_k \sqrt{p_{\text{edge}}}^{\mp} - L_k^{\mp}}{L_k \sqrt{p_{\text{edge}}}^{\pm} - L_k^{\pm}} \tag{72}$$
$$a_2 := \left(L_k \sqrt{p}^{\mp} - L_k^{\mp}\right) \tag{73}$$
$$\quad - a_1\left(L_k \sqrt{p}^{\pm} - L_k^{\pm}\right) \tag{74}$$
$$a_3 := L_k \sqrt{p}^{\mp}(1/\gamma) \tag{75}$$
$$a_4 := a_1 L_k \sqrt{p}^{\pm} \tag{76}$$
$$a_5 := -(a_2 + a_3 + a_4)/2 \tag{77}$$
$$a_6 := a_2 a_3 \tag{78}$$
$$M := a_5 + \sqrt{(a_5)^2 - a_6} \tag{79}$$

We can then apply the same recipe to the formulas (37)-(39) and (42)-(45), which produce the new price $\sqrt{p'}$ and the scaling factor η. This gives us the following:

$$b_1 := L_k \sqrt{p}^{\pm} - L_k^{\pm} - \Delta t_{out} \tag{80}$$
$$b_2 := L_k \sqrt{p}^{\mp} - L_k^{\mp} + \Delta t_{in} \tag{81}$$
$$b_3 := b_1/b_2 \tag{82}$$
$$b_4 := \frac{L_k^{\pm} - b_3 L^{\mp}}{2L_k} \tag{83}$$
$$\sqrt{p'} := b_4 + \sqrt{(b_4)^2 + b^3} \tag{84}$$
$$\eta := \frac{b_1}{L_k \sqrt{p'}^{\pm} - L_k^{\pm}} \tag{85}$$
$$L^n \to \eta L^n \quad (n \in L_k) \tag{86}$$

In total, we can use equations (67)-(86) to count the total arithmetic load per iteration. We have the following:

- 18 additions/subtractions
- $(13+|N_k|)$ multiplications
- 6 divisions
- 2 square roots

Part 6: A telescoping identity

Finally, we will derive here an interesting identity. Consider a sequence of iterations over a sequence of prices $p_0 \to p_1 \to p_2 \to \ldots \to p_m$. Let N_i be the set of relevant LPs for the i^{th} trade in this sequence, and suppose that η_i is the corresponding scale factor, with η_i^n defined analogously to (25). Each stage of this sequence corresponds to two points that change the token quantities:

$$p_i \to p_{i+1}, \qquad L^n \to \eta_i^n L^n$$

Let us explicitly follow these changes for the y token (an analogous calculation will hold for the x tokens). Starting with y_0, we alternate, adding Δy_i, and then scaling by a factor η_i, which itself can be represented by *adding* an amount scaled by $(\eta_i - 1)$. The calculation is enormously tedious, but most terms cancel due to telescoping:

$$\begin{aligned}
&\sum_n L^n\left([\sqrt{p}_0]^n - \sqrt{p}_a^n\right) \quad (y_0)\\
&+ \sum_n L^n\left([\sqrt{p}_1]^n - [\sqrt{p}_0]^n\right) \quad (p_0 \to p_1)\\
&+ \sum_n \left(\eta_1^n - 1\right) L^n\left([\sqrt{p}_1]^n - \sqrt{p}_a^n\right) \quad (L \to \eta_1 L)\\
&+ \sum_n \left(\eta_1^n\right) L^n\left([\sqrt{p}_2]^n - [\sqrt{p}_1]^n\right) \quad (p_1 \to p_2)\\
&+ \sum_n \left(\eta_2^n - 1\right)\left(\eta_1^n\right) L^n\left([\sqrt{p}_2]^n - \sqrt{p}_a^n\right) \quad (L \to \eta_2 L)\\
&+ \sum_n \left(\eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_3]^n - [\sqrt{p}_2]^n\right) \quad (p_2 \to p_3)\\
&+ \sum_n \left(\eta_3^n - 1\right)\left(\eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_3]^n - \sqrt{p}_a^n\right) \quad (L \to \eta_3 L)\\
&\qquad \vdots\\
&+ \sum_n \left(\eta_{m-1}^n \ldots \eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_m]^n - [\sqrt{p}_{m-1}]^n\right) \quad (p_{m-1} \to p_m)\\
&+ \sum_n \left(\eta_m^n - 1\right)\left(\eta_{m-1}^n \ldots \eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_m]^n - \sqrt{p}_a^n\right) \quad (L \to \eta_m L)\\
&= \sum_n \left(\eta_m^n \eta_{m-1}^n \ldots \eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_m]^n - \sqrt{p}_a^n\right) \quad (y_m)
\end{aligned} \tag{87}$$

Thus, the total change in y tokens is given by

$$\begin{aligned}
\Delta y = &\sum_n \left(\eta_m^n \eta_{m-1}^n \ldots \eta_2^n \eta_1^n\right) L^n\left([\sqrt{p}_m]^n - \sqrt{p}_a^n\right)\\
&- \sum_n L^n\left([\sqrt{p}_0]^n - \sqrt{p}_a^n\right)
\end{aligned} \tag{88}$$

By adding and subtracting $\sum_n L^n \sqrt{p_m}$, our expression in (88) can be rearranged into

$$\begin{aligned}
\Delta y = &\sum\nolimits_n L^n\left([\sqrt{p}_m]^n - [\sqrt{p}_0]^n\right)\\
&+ \sum\nolimits_n \left(\eta_m^n \eta_{m-1}^n \ldots \eta_2^n \eta_1^n - 1\right) L^n\left([\sqrt{p}_m]^n - \sqrt{p}_a^n\right)
\end{aligned} \tag{89}$$

Equation (89) can be interpreted as a trade directly from p_0 to p_m, followed by an overall one-time liquidity boost by a factor of $\left(\eta_m^n \eta_{m-1}^n \ldots \eta_2^n \eta_1^n\right)$. This doesn't actually save computational cost, but it shows that the process is, in some sense, **associative** and **commutative.**

As we mentioned in Section 2, implementing internal fees in concentrated liquidity comes with a computational tradeoff; more arithmetic for less storage. There is reason to believe that this may result in a net reduction of gas costs, but this remains to be seen. In the meantime, however, there is a more important feature of our fee structure that we should discuss; the payoff for LPs.

For this, we will run some market simulations. Our primary metric will be the impermanent loss (IL). The typical mathematical treatment of IL does not take fees into account. We will of course factor in the the fees, and we denote this quantity by ILF ('Impermanent Loss with Fees'). The structure of our simulations is as follows:

- **Initialize** identical LP positions in two distinct concentrated liquidity pools; one with internal fees and one with external fees. Each pool starts at the same initial price.
- **Iterate** over a random walk of price movements. For each new price $p(t)$, make an appropriate trade within each pool to move each pool to $p(t)$.
- **Calculate** the running ILF for each pool, as measured relative to the initial state.

For our random walk of prices, we will simulate lognormal Brownian motion, with a standard deviation σ which we refer to as *volatility* (with an appropriate time scale):

$$p(t+\Delta t) = p(t)e^{W} \quad \left(W \sim N(0, \sigma\sqrt{\Delta t})\right).$$

We choose a time step of $\Delta t = 1$ minute, and an initial volatility of $\sigma = 1.0$ (or 100%, in options-speak). For each simulation, we plot the running ILF over a total period of 3 months, as well as the time series of prices driving the simulation (see Figure 2 for our first example).

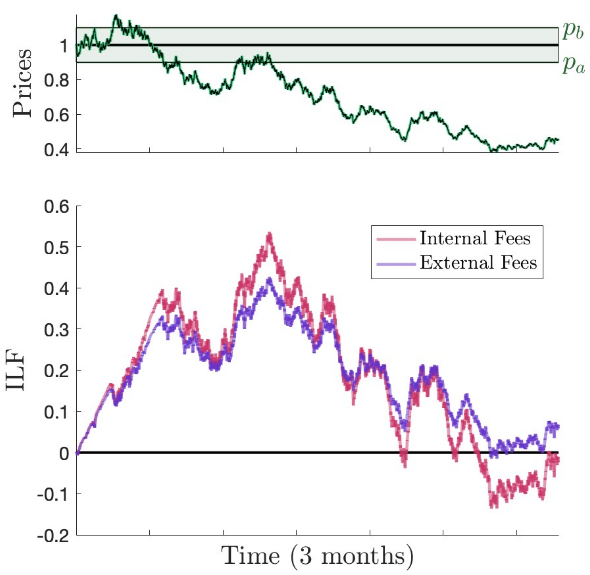

Figure 2. A single simulation.
Fee=0.3%, vol=100%, $[p_a, p_b] = [0.9, 1.1]$

In Figure 2, we begin with a fee size of 0.3%, an initial price of $p_0 = 1$, and an LP position of $[p_a, p_b] = [0.9, 1.1]$ We see that when the ILF is growing, it tends to grow *more* for internal fees, but when it is falling, it tends to fall more as well. We see also how the ILF depends directly on the price moving in or out of the LP position.

In fact, in Figure 3 we have a particular run where the price stays within our position most of the time, and the internal fee structure performs dramatically well.

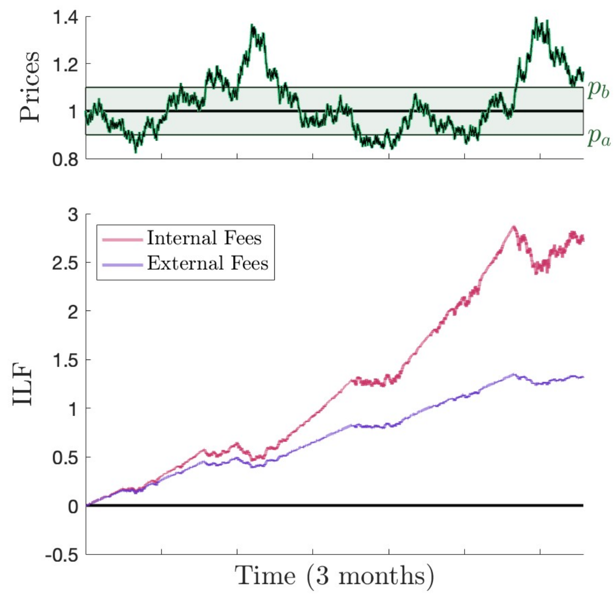

Figure 3. A single simulation
Fee=0.3%, vol=100%, $[p_a, p_b] = [0.9, 1.1]$

If we widen the LP position to $[p_a, p_b] = [0.5, 2.0]$, then we can almost guarantee that the market price stays within our position. However, this comes at a cost; the relative advantage of internal fees tends to diminish (compare the scale on the vertical axes between Figures 3 and 4).

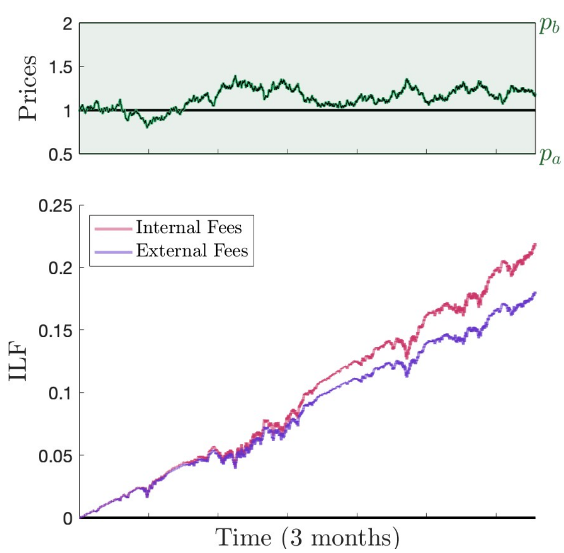

Figure 4. A single simulation.
Fee=0.3%, vol=100%, $[p_a, p_b] = [0.5, 2.0]$

While these individual simulation runs are interesting, we should really collect results over *many* trials. Once again beginning with a fee size of 0.3%, volatility at 100%, and an LP position of $[p_a, p_b] = [0.9, 1.1]$, we run 1,000 trials and collect the *final* ILF value over the 3 month period for each trial. The results are plotted in Figure 5.

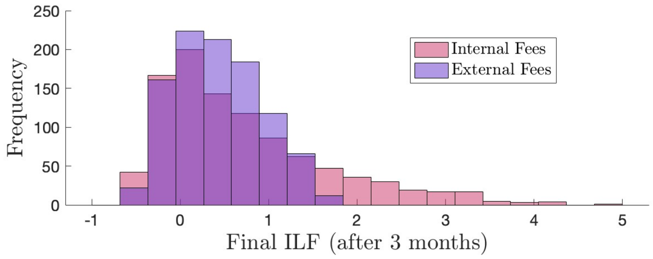

Figure 5. Distribution of final ILF for 1000 trials. Fee=0.3%, vol=100%, $[p_a, p_b] = [0.9, 1.1]$

These distributions seem to confirm our previous anecdotal observation that internal fees can produce *more extreme* ILF values. In particular, we see in Figure 5 that the the final ILF values for internal fees are over represented in both the positive and negative end of the distribution, while the distribution for external fees is bulkier in the middle.

(We should note that despite the apparent skewness of the distributions seen here, the situation is actually quite symmetric in the sense that negative ILF values live in the range $[-1, 0]$, while positive ILF values live in the range $[0, \infty)$. In some sense, the ILF should be judged from a *logarithmic* perspective.)

More tangibly, we can look at the difference in final ILF value *per trial*. We compute the difference as

$$\begin{bmatrix} \text{per trial} \\ \text{difference} \end{bmatrix} = \begin{bmatrix} \text{Final ILF} \\ \text{(internal fees)} \end{bmatrix} - \begin{bmatrix} \text{Final ILF} \\ \text{(external fees)} \end{bmatrix}$$

Note that the internal fee structure is preferable when this difference is *positive*.

For the same data set giving us Figure 5, we plot these per-trial-differences in Figure 6 below.

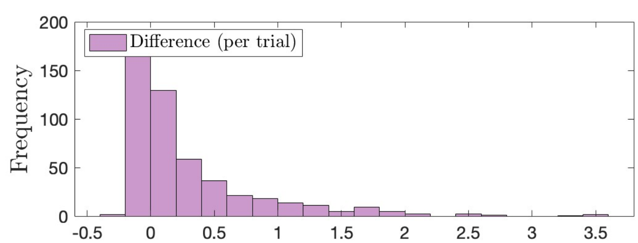

Figure 6. Difference, *per trial*, for the trials in Figure 4. Fee=0.3%, vol=100%, $[p_a, p_b] = [0.9, 1.1]$

We can see from the figure above that the distribution is split pretty evenly between positive/negative values, but there does exist a slight advantage for using internal fees. Specifically, internal fees resulted in a better ILF value in 63% of the trials.

Next, we widen the LP range to $[p_a, p_b] = [0.5, 2.0]$. Again, we run 1,000 trials, and the aggregated results are shown in Figure 7.

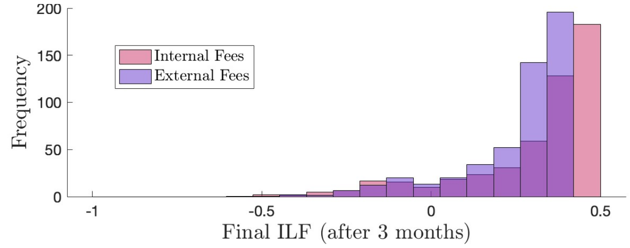

Figure 7. Distribution of final ILF for 1000 trials. Fee=0.3%, vol=100%, $[p_a, p_b] = [0.5, 2.0]$

This agrees with what we would expect in that the wider LP range will bring the distribution inward (less extreme ILF values). However, it is worth noting that in this case, using internal fees was preferable in 86% of the trials. Also, we note that the shape of the distribution has changed notably.

Returning to our initial LP position of $[0.9, 1.1]$, we next increase the volatility to 200%. The results of 1,000 trials are shown in Figure 8. As we'd expect, the ILF values are more extreme overall, but now the internal fees were only preferable in 46% of the trials (a minority!).

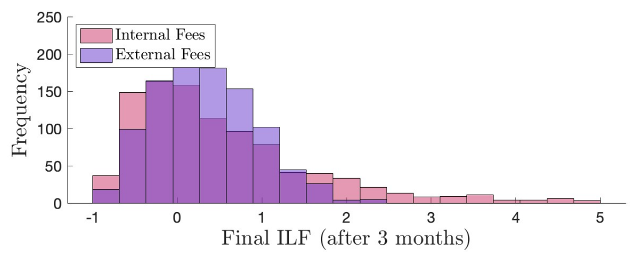

Figure 8. Distribution of final ILF for 1000 trials. Fee=0.3%, vol=200%, $[p_a, p_b] = [0.9, 1.1]$

Lastly, we raise the fee size to 1%. In this case, the ILF is able to reach comically high values in some trials, and overall, internal fees won out in 88% of the trials.

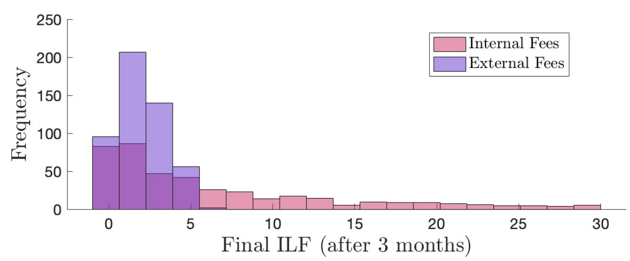

Figure 9. Distribution of final ILF for 1000 trials. Fee=1%, vol=100%, $[p_a, p_b] = [0.9, 1.1]$

Even if we bring the volatility back up to 200% and run 1,000 more trials, the internal fees are still preferable 73% of the time (compared to the 46% when the fee size was at 0.3%).

Based on these simulations, we can conjecture that using internal fees will be generally preferable to the LP whenever volatility is low, and/or the fee size is large.

A more general derivation

Now we will re-derive our swap equations (72)-(86). In the original approach, we derived the equations for one particular trade direction, and then proposed the analogous equations for the opposite trade direction by symmetry arguments. This time, we will derive them for both trade directions simultaneously from the start.

Again we suppose a trade starting at price p and staying within a range $[p_k, p_{k+1}]$. Just as before, we will be general and use the symbols Δt_{in} and Δt_{out} for the incoming and outgoing tokens, respectively. In the same spirit, we use t_{in} and t_{out} to represent the total token amounts themselves, either x or y. And finally, we will use either $\pm$ and $\mp$ as needed, summarized in the table:

direction:	price goes down	price goes up
$t_{in}(p)$	$x(p)$	$y(p)$
$t_{out}(p)$	$y(p)$	$x(p)$
Δt_{in}	Δx	Δy
Δt_{out}	Δy	Δx
$\pm$	$+$	$-$
$\mp$	$-$	$+$

Finally, to take full advantage of the power of '$\pm$', we relable the LP price ranges. Instead of the customary $[p_a, p_b]$, we will use $\pm$ subscripts:

$$p_+^n := p_a^n \tag{90}$$

$$p_-^n := p_b^n \tag{91}$$

so that each LP corresponds now to the range $[p_+^n, p_-^n]$. With this notation, equations (3)-(4) can be written as

$$t_{in}^n(p) = L^n\big(\sqrt{[p]^n}^{\mp} - \sqrt{p_{\mp}^n}^{\mp}\big) \tag{92}$$

$$t_{out}^n(p) = L^n\big(\sqrt{[p]^n}^{\pm} - \sqrt{p_{\pm}^n}^{\pm}\big) \tag{93}$$

The total token amounts are given by summing over n:

$$t_{in}(p) = \sum\nolimits_n t_{in}^n(p) = \sum\nolimits_n L^n\big(\sqrt{[p]^n}^{\mp} - \sqrt{p_{\mp}^n}^{\mp}\big) \tag{94}$$

$$t_{out}(p) = \sum\nolimits_n t_{out}^n(p) = \sum\nolimits_n L^n\big(\sqrt{[p]^n}^{\pm} - \sqrt{p_{\pm}^n}^{\pm}\big) \tag{95}$$

Similarly, for a swap moving from price p to p', we find the deltas by summing over all LPs as well:

$$\Delta t_{in} = \sum\nolimits_n L^n\big(\sqrt{[p']^n}^{\mp} - \sqrt{[p]^n}^{\mp}\big) \tag{96}$$

$$\Delta t_{out} = \sum\nolimits_n L^n\big(\sqrt{[p']^n}^{\pm} - \sqrt{[p]^n}^{\pm}\big) \tag{97}$$

Just as before, we think of the set of active ticks $\{(p_+^n, p_-^n)\}$ as being sorted into $\{p_k\}$. Then, for a given range $[p_k, p_{k+1}]$, we define the set of *relevant LPs* by

$$N_k := \{\ n \ \mid\ [p_k, p_{k+1}] \subseteq [p_+^n, p_-^n]\ \} \tag{98}$$

Using this, we once again define the total liquidity available in the current range by

$$L_k := \sum\nolimits_{n\in N_k} L^n \tag{99}$$

Now we consider some specific trade going from a price p to p', entirely contained in some range $[p_k, p_{k+1}]$. By the definition of the 'box' function,

$$[p]^n := \left\{ \begin{array}{cc} p_+^n & p < [p_+^n, p_-^n] \\ p & p \in [p_+^n, p_-^n] \\ p_-^n & p > [p_+^n, p_-^n] \end{array} \right\} \tag{100}$$

we can see that $[p']^n = [p]^n$ for any $n \notin N_k$ (because both quantities will be equal to the nearest edge of the LP interval $[p_+^n, p_-^n]$). Thus, for the trade $p \to p'$, we can see that the $n \notin N_k$ terms in (96)-(97) will vanish. On the other hand, for $n \in N_k$, then we will have $[p]^n = p$ and $[p']^n = p'$. and so we may write

$$\Delta t_{in} = \sum_{n\in N_k} L^n\big(\sqrt{p'}^{\mp} - \sqrt{p}^{\mp}\big) = L_k\big(\sqrt{p'}^{\mp} - \sqrt{p}^{\mp}\big) \tag{101}$$

$$\Delta t_{out} = \sum_{n\in N_k} L^n\big(\sqrt{p'}^{\pm} - \sqrt{p}^{\pm}\big) = L_k\big(\sqrt{p'}^{\pm} - \sqrt{p}^{\pm}\big) \tag{102}$$

By eliminating the variable p' from equations (101)-(102), one finds relationships between the incoming and outgoing deltas similar to the constant product formulas:

$$\Delta t_{out} = -\frac{L\sqrt{p}^{\pm}\,\Delta t_{in}}{L\sqrt{p}^{\mp} + \Delta t_{in}} \tag{103}$$

If we are storing fees internally, then we take a fee parameter $\gamma < 1$ and use the modified incoming amount $(\gamma\Delta t_{in})$ to compute the outgoing amount. Additionally, we will take the outgoing amount to be *positive*, and simply subtract it (this is purely for readability). Altogether, our delta conversion formula will be the following:

$$\Delta t_{out} = \frac{L\sqrt{p}^{\pm}\,(\gamma\Delta t_{in})}{L\sqrt{p}^{\mp} + (\gamma\Delta t_{in})} \tag{104}$$

When storing the fees internally, we deposit the *entire* amount Δt_{in} into the pool. Thus, the token quantity update should be given by

$$t_{in} + \Delta t_{in} \to t_{in} \tag{105}$$

$$t_{out} - \Delta t_{out} \to t_{out} \tag{106}$$

with the understanding that the Δt_{out} in (106) is given by expression (104). The primary idea that was introduced in Section 1 was the notion that the change (105)-(106) results in an effective *scaling* of the available liquidity L_k. We will denote this scale factor by the letter η, illustrated in figure 7 below:

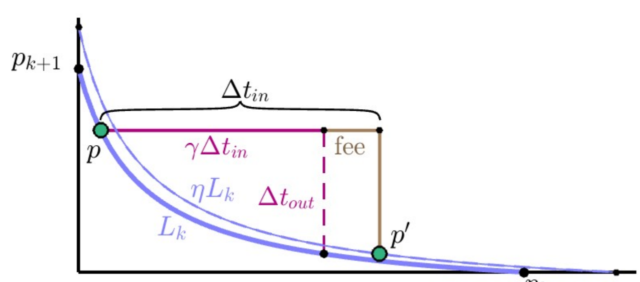

Figure 7. Scaling by eta

In the figure above, we begin at price p and end at some new price p' (for the sake of illustration, we chose $p > p'$). Of course, only the relevant LPs should benefit from this scaling (as we illustrated in figure 5). If we define η^n by

$$\eta^n := \left\{ \begin{array}{ll} \eta & n \in N_k \\ 1 & n \notin N_k \end{array} \right\} \tag{107}$$

then we can articulate the update step illustrated in figure 7 by rewriting (105)-(106) as the following:

$$\sum\nolimits_n t^n_{in}(p) \; + \Delta t_{in} = \sum\nolimits_n \eta^n \, t^n_{in}(p') \tag{108}$$

$$\sum\nolimits_n t^n_{out}(p) \; - \Delta t_{out} = \sum\nolimits_n \eta^n \, t^n_{out}(p') \tag{109}$$

However, as we previously observed, for the two prices p and p', both in the range $[p_k, p_{k+1}]$, then for any $n \notin L_k$, we will have $[p]^n = [p']^n$. But we can see from the definitions in (92)-(93) that this means

$$t^n_{in}(p) = t^n_{out}(p'), \quad t^n_{out}(p) = t^n_{out}(p') \quad (n \notin L_k) \tag{110}$$

Combined with the fact that $\eta^n = 1$ for $n \notin L_k$, then we can see that all $n \notin L_k$ terms will cancel from both sides of (108)-(109), and so we may simply write the following:

$$\sum_{n \in N_k} t^n_{in}(p) \; + \Delta t_{in} = \sum_{n \in N_k} \eta \, t^n_{in}(p') \tag{111}$$

$$\sum_{n \in N_k} t^n_{out}(p) \; - \Delta t_{out} = \sum_{n \in N_k} \eta \, t^n_{out}(p') \tag{112}$$

Now we substitute in definitions (92)-(93), keeping in mind that for $n \in L_k$, then $[p]^n = p$ and $[p']^n = p'$:

$$\sum_{n \in N_k} L^n \big(\sqrt{p}^{\mp} \text{-} \sqrt{p^n_{\mp}}^{\mp}\big) + \Delta t_{in} = \eta \sum_{n \in N_k} L^n \big(\sqrt{p}^{\mp} \text{-} \sqrt{p^n_{\mp}}^{\mp}\big) \tag{113}$$

$$\sum_{n \in N_k} L^n \big(\sqrt{p}^{\pm} \text{-} \sqrt{p^n_{\pm}}^{\pm}\big) - \Delta t_{out} = \eta \sum_{n \in N_k} L^n \big(\sqrt{p}^{\pm} \text{-} \sqrt{p^n_{\pm}}^{\pm}\big) \tag{114}$$

If we define the quantities

$$L^{\pm}_k : \sum_{n \in N_k} L^n \sqrt{p^n_{\pm}}^{\pm} \tag{115}$$

then equations (113)-(114) can be written more simply:

$$L_k \sqrt{p}^{\mp} - L^{\mp}_k + \Delta t_{in} = \eta \Big[L_k \sqrt{p'}^{\mp} - L^{\mp}_k \Big] \tag{116}$$

$$L_k \sqrt{p}^{\pm} - L^{\pm}_k - \Delta t_{out} = \eta \Big[L_k \sqrt{p'}^{\pm} - L^{\pm}_k \Big] \tag{117}$$

Just as before, we notice that, at a current price p, with computable quantities L_k, $L^{\pm}_k$, a prescribed incoming amount Δt_{in} and a resulting outgoing quantity Δt_{out}, then the two equations (116)-(117) contain only two unknowns, p' and η. We will start by eliminating η, and to this end we divide equation (117) by equation (116):

$$\frac{L_k \sqrt{p}^{\pm} - L^{\pm}_k - \Delta t_{out}}{L_k \sqrt{p}^{\mp} - L^{\mp}_k + \Delta t_{in}} = \frac{L_k \sqrt{p'}^{\pm} - L^{\pm}_k}{L_k \sqrt{p'}^{\mp} - L^{\mp}_k} \tag{118}$$

To simplify (118), we define the following quantities:

$$b_1 := L_k \sqrt{p}^{\pm} - L^{\pm}_k - \Delta t_{out} \tag{119}$$

$$b_2 := L_k \sqrt{p}^{\mp} - L^{\mp}_k + \Delta t_{in} \tag{120}$$

$$b_3 := b_1 / b_2 \tag{121}$$

Then (118) can be written as the following:

$$b_3 = \frac{L_k \sqrt{p'}^{\pm} - L^{\pm}_k}{L_k \sqrt{p'}^{\mp} - L^{\mp}_k} \tag{122}$$

Using the fact that $\left(\sqrt{p}^{\pm} \sqrt{p}^{\mp}\right) = 1$, then we can easily rearrange (122) into a quadratic expression for $\sqrt{p'}^{\pm}$:

$$0 = \left(\sqrt{p'}^{\pm}\right)^2 + \left[\frac{b_3 L^{\mp}_k - L^{\pm}_k}{L_k} \right] \left(\sqrt{p'}^{\pm}\right) - b_3 \tag{123}$$

If we then define

$$b_4 := - \frac{b_3 L^{\mp} - L^{\pm}_k}{2 L_k}, \tag{124}$$

the solution to (124) is given by

$$\sqrt{p'}^{\pm} = b_4 + \sqrt{(b_4)^2 + b^3} \tag{125}$$

(One can check that, because $b_3 > 0$, we *must* take the positive square root in (125) instead of the negative square root, to guarantee the new price is positive.)

Finally, having found p', we can now solve for η. For example, (117) gives us

$$\eta = \frac{b_1}{L_k \sqrt{p'}^{\pm} - L^{\pm}_k} \tag{126}$$

which we use to update the relevant LP liquidities:

$$L^n \to \eta \, L^n \quad (n \in L_k) \tag{127}$$

We can use this same framework to determine the upper bound M for an incoming amount into the particular range $[p_k, p_{k+1}]$. To stay general, we will use p_{edge} to denote the price at the edge of the interval, depending on the direction of trade (either p_k or p_{k+1}). Then we set (125) equal to $\sqrt{p_{\text{edge}}}^{\pm}$ as follows:

$$\sqrt{p_{\text{edge}}}^{\pm} = b_4 + \sqrt{(b_4)^2 + b^3} \tag{128}$$

Our goal now will simply be to solve for the Δt_{in} that is implicitly defined above within the equation (128). First, by subtracting b_4, squaring, and simplifying, we find

$$(p_{\text{edge}})^{\pm} - 2 b_4 \sqrt{p_{\text{edge}}}^{\pm} = b_3 \tag{129}$$

Substituting in the expression (124) for b_4, we get

$$(p_{\text{edge}})^{\pm} + \left(\frac{b_3 L^{\mp} - L^{\pm}_k}{L_k} \right) \sqrt{p_{\text{edge}}}^{\pm} = b_3 \tag{130}$$

Solving (130) for b_3, one finds

$$\frac{L_k p^{\pm}_{\text{edge}} - L^{\pm}_k}{L_k p^{\mp}_{\text{edge}} - L^{\mp}_k} = b_3 \tag{131}$$

Using definitions (119)-(121), we can write b_3 explicitly:

$$\frac{L_k\, p^{\pm}_{\text{edge}} - L^{\pm}_k}{L_k\, p^{\mp}_{\text{edge}} - L^{\mp}_k} = \frac{L_k\sqrt{p}^{\pm} - L^{\pm}_k - \Delta t_{out}}{L_k\sqrt{p}^{\mp} - L^{\mp}_k + \Delta t_{in}} \tag{132}$$

Recalling that our goal is to solve for Δt_{in}, we see it appears explicitly in the denominator and implicitly in the numerator (within Δt_{out}) of the right side of (132). Let's first give a name to (the reciprocal) of the quantity on the left side,

$$a_1 := \frac{L_k\, p^{\mp}_{\text{edge}} - L^{\mp}_k}{L_k\, p^{\pm}_{\text{edge}} - L^{\pm}_k} \tag{133}$$

so that (132) may be written

$$1/a_1 = \frac{L_k\sqrt{p}^{\pm} - L^{\pm}_k - \Delta t_{out}}{L_k\sqrt{p}^{\mp} - L^{\mp}_k + \Delta t_{in}} \tag{134}$$

(The reason for including reciprocal in the definition of our a_1 is to reduce the overall computational cost later.) Equation (134) can be rearranged into

$$\big(L_k\sqrt{p}^{\mp} - L^{\mp}_k\big) - a_1\big(L_k\sqrt{p}^{\pm} - L^{\pm}_k\big) + \Delta t_{in} = \text{-}a_1\Delta t_{out} \tag{135}$$

To clean up (135), we define

$$a_2 := \big(L_k\sqrt{p}^{\mp} - L^{\mp}_k\big) - a_1\big(L_k\sqrt{p}^{\pm} - L^{\pm}_k\big) \tag{136}$$

so that (135) becomes

$$a_2 + \Delta t_{in} = -a_1\Delta t_{out}. \tag{137}$$

Now we restore the Δt_{in} dependence in Δt_{out} from (137):

$$a_2 + \Delta t_{in} = -a_1\left(\frac{L\sqrt{p}^{\pm}\,(\gamma\Delta t_{in})}{L\sqrt{p}^{\mp} + (\gamma\Delta t_{in})}\right) \tag{138}$$

Equation (138) can be rearranged into a quadratic expression in Δt_{in}:

$$\big(\Delta t_{in}\big)^2 + \left[a_2 + \frac{L_k}{\sqrt{p}^{\pm}\,\gamma} + a_1 L_k\sqrt{p}^{\pm}\right]\Delta x + \left[\frac{a_2 L_k}{\sqrt{p}\,\gamma}\right] = 0 \tag{139}$$

If we define the quantities

$$a_3 := L_k/(\sqrt{p}^{\pm}\,\gamma) \tag{140}$$

$$a_4 := a_1\, L_k\sqrt{p}^{\pm} \tag{141}$$

$$a_5 := -(a_2 + a_3 + a_4)/2 \tag{142}$$

$$a_6 := a_2\, a_3 \tag{143}$$

then the solution to (139) can be expressed as

$$\Delta t_{in} = a_5 + \sqrt{(a_5)^2 - a_6} \tag{144}$$

We may take this as our maximum incoming amount M:

$$M := a_5 + \sqrt{(a_5)^2 - a_6} \tag{145}$$

With that, we have then reproduced all of our familiar formulas, but from a more general approach.