

Lifinity: A Proactive Market Maker with Concentrated Liquidity

The Lifinity team

April 2022

Introduction

Since their inception in 2018, AMMs have grown to become the core infrastructure of DeFi, allowing anyone to trade and market make in a few clicks. However, although TVL and trading volume have shown exponential growth, the underlying technology has seen little improvement. Issues such as capital inefficiency, impermanent loss, and high fees remain unsolved, creating costs that are rarely seen in TradFi for both traders and liquidity providers.

We propose to solve these issues by 1) concentrating liquidity, 2) proactively market making using an oracle, and 3) implementing a rebalancing mechanism. Concentrating liquidity improves capital efficiency for liquidity providers while reducing slippage for traders. By using an oracle as the key pricing mechanism, the market maker is able to reduce impermanent loss, as it does not rely on arbitrageurs to adjust the price. Lastly, the rebalancing mechanism ensures that the pool balance regresses to the bonding curve while generating profit by delaying the rebalancing process.

Concentrated Liquidity

In the constant product model ($x \cdot y = k$), liquidity is provided across the price range from 0 to infinity. In contrast, concentrated liquidity adds depth to a market by providing capital in a limited price range, thereby improving capital efficiency. There are several methods of concentrating liquidity, but we take the simple yet unique approach of applying leverage to the value of k . This is achieved by simply multiplying k by the desired level of concentration c , which determines the amount of liquidity provided:

$$x \cdot y = c \cdot k = K$$

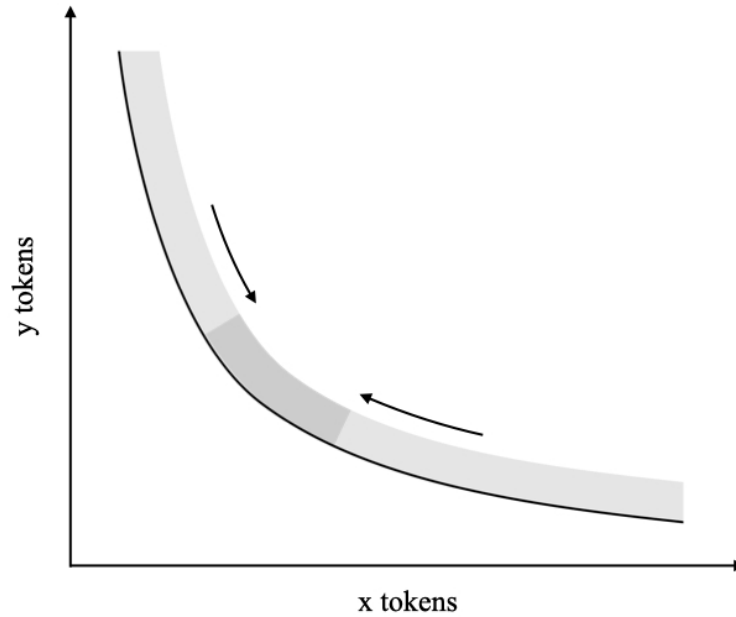


Figure 1: Concentrating liquidity

Proactive Market Making

Although concentrating liquidity improves capital efficiency, it also increases the risk of impermanent loss. To minimize this risk, the protocol proactively market makes using an oracle. This frees the pools from reliance on arbitrageurs to adjust the price and can greatly reduce or even reverse impermanent loss. Furthermore, since oracles on Solana such as the Pyth Network provide price updates every slot (about 0.5 seconds), it is extremely difficult to profitably front run.

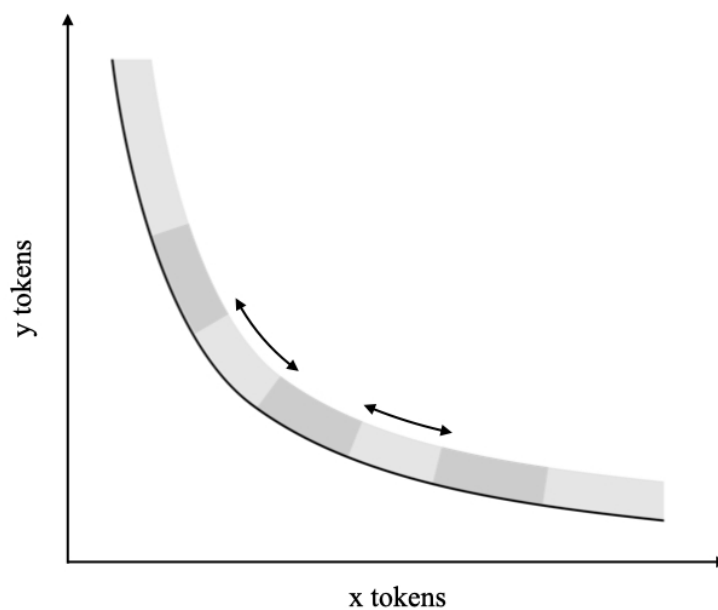


Figure 2: Concentrated liquidity moves along the bonding curve

Why does using an oracle reduce the risk of impermanent loss? On constant product AMMs, every trade that moves the price away from the price at which liquidity was initially deposited results in impermanent loss. Using an oracle helps avoid this because the AMM adjusts to the “correct” price before arbitrageurs can trade against stale prices, allowing LPs to avoid impermanent loss that they otherwise would have incurred.

Rebalancing Mechanism

Although this is sufficient for the protocol to efficiently market make, it lacks a mechanism to keep the pool balanced, which increases the risk of impermanent loss. To minimize this risk while maximizing profit, the protocol rebalances its pools by adjusting their liquidity.

This adjustment of liquidity occurs on every trade and/or change in oracle price according to the following formulae:

$$K_{\text{adjusted}} = K \cdot (x \cdot p / y)^z \quad \text{to increase liquidity for } x$$

$$K_{\text{adjusted}} = K \cdot (y / (x \cdot p))^z \quad \text{to decrease liquidity for } x$$

where K is the total amount of liquidity, p is the price of x provided by the oracle, and z is the parameter that determines the magnitude of the adjustment.

For example, when x comprises less than 50% of the total value of the pooled assets, the market maker will decrease liquidity for buyers of x and increase liquidity for sellers of x in order to regain balance. This will incentivize traders to sell against the pool while discouraging them from buying, ensuring that the pool balance regresses to the bonding curve.

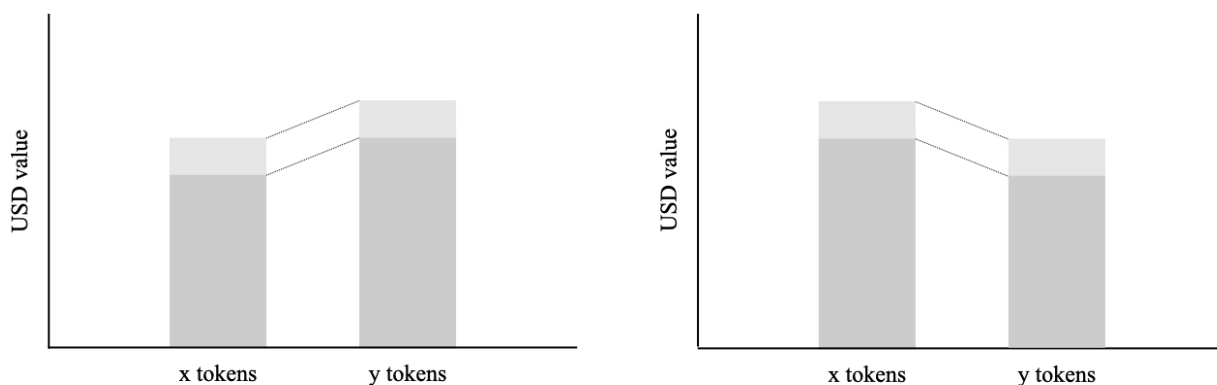


Figure 3: Rebalancing

An additional advantage of implementing a rebalancing mechanism is that the protocol is able to concentrate liquidity beyond what is reasonable in other concentrated liquidity protocols because it makes it more difficult for the price to go out of the position’s price range.

Rebalancing Scenarios

Let's consider the possible scenarios that can occur in practice. The pool balance only changes when users trade with the pool or the oracle price changes. The combination of the two can generate a profit or a loss for liquidity providers. Below are the four possible scenarios and how the protocol handles them:

- 1) The price of x *increases* when x comprises *more* than 50% of the pool
- 2) The price of x *decreases* when x comprises *less* than 50% of the pool
- 3) The price of x *increases* when x comprises *less* than 50% of the pool
- 4) The price of x *decreases* when x comprises *more* than 50% of the pool

1 & 2 are scenarios in favor of liquidity providers since they generate a profit or reverse the impermanent loss that constant product market makers would incur. The rebalancing mechanism effectively buys the dip when prices fall and takes profit when prices rise. These are the most likely scenarios since the oracle price tends to move faster than the price on other DEXs and arbitrageurs cannot profitably trade against our pools.

Conversely, 3 & 4 are scenarios that work against liquidity providers. They are less likely to occur since the oracle price usually moves ahead of other DEXs. In the case that it does happen, however, the protocol will adjust its liquidity to minimize further impermanent loss.

Conclusion

Although liquidity concentration has heretofore implied greater risk of impermanent loss, Lifinity has developed a novel market making mechanism that concentrates liquidity even as it reduces impermanent loss. Further, since liquidity is adjusted automatically according to an oracle, liquidity providers do not need to actively manage their positions. Lifinity thus improves both liquidity for traders and the profitability of market making for liquidity providers.