Yield Collective

March 2024

Weston Nelson
weston@yieldcollective.xyz

A guide to building the most advanced yield optimization protocol in DeFi by leveraging UniswapV3 on scaling networks, machine learning profitability analysis algorithms, and automation.

INTRODUCTION

Yield optimization is the process of maximizing the returns from providing liquidity to decentralized exchanges (DEXs) such as Uniswap. Uniswap is a DEX that allows users to swap any two tokens without intermediaries and earn fees by supplying liquidity to the pools. UniswapV3 is the latest version of Uniswap (V4 to be deployed this year), which introduces several features that make yield optimization more complex and challenging, such as concentrated liquidity, multiple fee tiers, and dynamic price ranges.

To achieve optimal yield, liquidity providers (LPs) need to constantly monitor and adjust their positions according to the market conditions and the performance of their pools. This requires a high level of technical expertise, time, and resources, which are not available to most LPs. Thus, most LPs do not earn significant real returns on their capital in positions and most in fact lose money due to impermanent loss and slippage when rebalancing.

THE SOLUTION

The solution for maximizing net profitability from liquidity provisioning is to build a yield optimization tool unlike any other: one that leverages UniswapV3's capabilities for concentrating liquidity, including profitability analysis algorithms in position selection, and machine learning from acquired data over time to continually refine and iterate in creating better performance. The tool will consist of a bot and set of programs written in Solidity/JavaScript to interact with the Uniswap smart contracts, and a set of programs will automate the process of creating, managing, and rebalancing liquidity positions on UniswapV3, not unlike many existing liquidity solutions of today. The primary difference - and key advantage of Yield as a tool - is the ability to dynamically set price ranges and consistently rebalance positions to always be the top fee-earning position, relative to others, in every pool it enters and then continue to perform this analysis on each potential candidate of a potential position until this opportunity is found. Thus, by forecasting with both pure mathematical approaches for ideal LPing plus the addition of pragmatic market analysis and monitoring (at times discretionary) this leads to a situation where positions are only entered if they are deemed profitable at the outset, and then monitored and adjusted in real-time as they accrue fees and prices change. The bot(s) will use profitability analysis algorithms to evaluate the potential returns and risks of different pools, fee tiers, and price ranges, and suggest the optimal strategies for each LP. The bot will also use machine learning from acquired data to continually improve its performance and intelligence, by learning from the historical and real-time data of the market, the pools, and the LPs.

The tool will enable LPs to achieve higher and more consistent returns, while reducing the complexity, time, and costs involved in yield optimization. To determine if a position will be profitable to the provisioner of liquidity, an algorithm has been created to serve as a rough guide for the protocol mechanisms and functionality.

- **Calculate Your Position in the Pool**:

$$Let\ P\ represent\ your\ proportion\ of\ liquidity\ relative\ to\ the\ total\ liquidity\ in\ the\ pool.$$

- **Estimate Trading Volume Rate**:

$$Let\ V\ represent\ the\ rate\ of\ trading\ volume\ in\ the\ pool\ over\ a\ specific\ time\ interval.$$

- **Estimate Fees Earned**:

$$Let\ F\ represent\ fees\ earned.$$

$$F = \text{P x V x R}$$

R is the fee rate (expressed as a percentage of trading fees earning for provisioning liquidity).

- **Assess Volatility Impact**:

$Let\ L\ represent\ the\ potential\ loss\ due\ to\ volatility\ (impermanent\ loss).$

· **Compare Fees Earned with Volatility Impact**:

$If\ F\ >\ L\ ,\ providing\ liquidity\ may\ be\ profitable.$

· **Now, to put this all together:**

$$Profitability\ =\ \{Profitable\ if\ F\ >\ L$$
$$\{Not\ Profitable\ if\ F \leq L$$

Where:

- **F** are the fees earned.
- ***L*** *i*s the potential loss due to volatility.
- ***R*** is the fee rate (percentage of trading fees earned for providing liquidity).
- ***P*** is your proportion of liquidity relative to the total liquidity pool.
- ***V*** is the rate of trading volume in the pool over a specific time interval

Note: Including your position - and its specific range - in relation to other positions in each pool is the final variable that needs to be added to further enhance your likelihood of profitability.

PROFITABILITY ANALYSIS ALGORITHMS

The profitability analysis algorithms are the core component of the yield optimization tool. They are responsible for calculating the expected profit and loss (P/L) of liquidity positions based on a few various factors, such as the pool size, the fee tier, the price range, the trading volume, the price volatility, and the gas costs (negligible on Layer2 chains, but still important to account for) and then perhaps most important, your band/tick range for the position. The algorithms use mathematical models and simulations to estimate the P/L of different scenarios and compare them to find the optimal strategy for each LP.

MATHEMATICAL MODELS

The mathematical models are based on the formulas and parameters of UniswapV3, which are described in detail in their documentation at https://docs.uniswap.org. The main formulas and parameters are:

- The invariant formula, which determines the relationship between the reserves of the two tokens in a pool and the price of the swap.
- The fee formula, which determines the amount of fees collected by the LPs for each swap.
- The liquidity formula, which determines the amount of liquidity that an LP can provide to a pool within a specific price range.
- The tick formula, which determines the discrete price points (ticks) that define the boundaries of the price ranges.
- The fee tier parameter, which determines the percentage of fees collected by the LPs for each pool.
- The gas cost parameter, which determines the amount of gas required to perform transactions on UniswapV3. Layer2 scaling solutions provide a key new opportunity at this moment in time.

The mathematical models use these formulas and parameters to calculate the P/L of a liquidity position, given the initial and final states of the pool and the position. The P/L consists of two components: the fee income and the impermanent loss. The fee income is the amount of fees earned by the LP for providing liquidity to the pool. The impermanent loss is the amount of potential profit that the LP loses due to the price change/internal ratio changes occurring of the tokens in the pool while each position is live. The P/L can be expressed as follows:
P/L = Fee Income - Impermanent Loss

SIMULATIONS

The simulations are based on the historical and real-time data of the market and the pools, which are acquired by the tool from various sources, such as undefined, undefined, and undefined. The data include the price, volume, volatility, and gas costs of the tokens and the pools. The simulations use the data to generate realistic and probabilistic scenarios of the market and the pool behavior and apply the mathematical models to calculate the P/L of each scenario. The simulations can be classified into two types: backtesting and forecasting.

- Backtesting is the process of testing the performance of a strategy based on the historical data of the market and the pool. It can be used to evaluate the past performance of a strategy and compare it with other strategies or benchmarks.
- Forecasting is the process of predicting the performance of a strategy based on the real-time data of the market and the pool. It can be used to estimate the future performance of a strategy and adjust it according to the changing market conditions.

EXAMPLES

To illustrate the profitability analysis algorithms, we present some examples of the P/L calculations based on the simulations. We assume that the LP has 10 ETH and 100,000 USDC and wants to provide liquidity to the ETH/USDC pool on UniswapV3. We also assume that the current price of ETH is 1000 USDC, and the gas cost is 50 Gwei. We compare the P/L of three different strategies: (1) providing liquidity to the entire price range with the 0.3% fee tier, (2) providing liquidity to the price range of [900, 1100] with the 0.3% fee tier, and (3) providing liquidity to the price range of [900, 1100] with the 1% fee tier.

Strategy	Initial Liquidity	Initial Price	Final Price	Trading Volume	Fee Income	Impermanent Loss	P/L
(1)	$5,000	$1000	$1200	$10,000,000	$30,000	-$50,000	-$20,000
(2)	$10,000	$1000	$1200	$5,000,000	$15,000	-$25,000	-$10,000
(3)	$10,000	$1000	$1200	$5,000,000	$50,000	-$25,000	$25,000

The table shows the P/L of each strategy, based on the backtesting simulation with the historical data of the market and the pool. The data show that the price of ETH increased from 1000 USDC to 1200 USDC, and the trading volume of the pool was 10,000,000 USDC. The table also shows the initial liquidity, the initial price, the final price, the fee income, and the impermanent loss of each strategy.

The results show that strategy (3) has the highest P/L, followed by strategy (2), and strategy (1) has the lowest P/L. This is because strategy (3) has the highest fee income, due to the higher fee tier, and the lowest impermanent loss, due to the concentrated liquidity. Strategy (2) has a lower fee income than strategy (1), due to the lower trading volume, but a lower impermanent loss, due to the concentrated liquidity. Strategy (1) has the lowest P/L, due to the low fee income and the high impermanent loss, caused by the wide price range and the low fee tier.

MACHINE LEARNING FROM ACQUIRED DATA

The machine learning from acquired data is the advanced component of the yield optimization tool. It is responsible for enhancing the performance and intelligence of the bot and the programs, by learning from the historical and real-time data of the market, the pools, and the LPs. The machine learning uses various techniques and methods, such as supervised learning, unsupervised learning, reinforcement learning, and deep learning, to analyze the data and generate insights, predictions, and recommendations.

INSIGHTS

The insights are the results of the data analysis, which provide useful information and knowledge about the market and the pool behavior, the profitability and risk factors, and the LP preferences and behavior. The insights can be used to improve the understanding and decision making of the bot and the programs, as well as the LPs. Some examples of the insights are:

- The correlation and causation between the price, volume, volatility, and gas costs of the tokens and the pools.
- The distribution and frequency of the price changes and the trading volume of the tokens and the pools.
- The optimal and suboptimal price ranges and fee tiers for different pools and market conditions.
- The trade-offs and risks involved in providing liquidity to different pools and strategies.
- The performance and behavior patterns of the LPs and the strategies.
- The feedback and satisfaction of the LPs and the users of the tool.

PREDICTIONS

The predictions are the outputs of the data modeling, which provide probabilistic estimates and forecasts of the future outcomes and events of the market and the pool behavior, the profitability and risk factors, and the LP preferences and behavior. The predictions can be used to optimize and adjust the strategies and actions of the bot and the programs, as well as the LPs. Some examples of the predictions are:

- The expected price, volume, volatility, and gas costs of the tokens and the pools in the next hour, day, week, or month.
- The expected P/L of different strategies and positions in the next hour, day, week, or month.
- The expected demand and supply of liquidity for different pools and fee tiers in the next hour, day, week, or month.
- The expected behavior and actions of the LPs and the users of the tool in the next hour, day, week, or month.
- The expected impact and effect of the bot and the programs on the market and the pool behavior, the profitability and risk factors, and the LP preferences and behavior in the next hour, day, week, or month.

RECOMMENDATIONS

The recommendations are the suggestions and advice of the data optimization, which provide optimal and feasible solutions and alternatives for the market and the pool behavior, the profitability and risk factors, and the LP preferences and behavior. The recommendations can be used to guide and assist the bot and the programs, as well as the LPs, in choosing and implementing the best strategies and actions. Some examples of the recommendations are:

- The best pool, fee tier, and price range to provide liquidity to, based on the current and expected market conditions and the LP's goals and preferences.
- The best time and frequency to create, manage, and rebalance liquidity positions, based on the current and expected market conditions and the LP's goals and preferences.
- The best way to allocate and diversify the LP's assets and liquidity among different pools and strategies, based on the current and expected market conditions and the LP's goals and preferences.
- The best way to use and interact with the bot and the programs, based on the current and expected market conditions and the LP's goals and preferences.
- The best way to improve and enhance the performance and intelligence of the bot and the programs, based on the current and expected market conditions and the LP's goals and preferences.

CONCLUSION

Yield optimization is a lucrative but challenging opportunity for LPs on UniswapV3. To overcome the challenges, LPs need a yield optimization tool that leverages UniswapV3, profitability analysis algorithms, and machine learning from acquired data. The tool will automate and optimize the process of creating, managing, and rebalancing liquidity positions on UniswapV3, and enhance the performance and intelligence of the bot and the programs. The tool will provide LPs with a competitive edge and a superior user experience, while contributing to the growth and innovation of the Uniswap ecosystem.

REFERENCES

[1] Hayden Adams, Noah Zinsmeister, and Dan Robinson. 2020. Uniswap v2 Core. Retrieved Feb 24, 2021, from https://uniswap.org/whitepaper.pdf
[2] Guillermo Angeris and Tarun Chitra. 2020. Improved Price Oracles: Constant Function Market Makers. In Proceedings of the 2nd ACM Conference on Advances in Financial Technologies (AFT ’20). Association for Computing Machinery, New York, NY,

United States, 80–91. https://doi.org/10.1145/3419614.3423251
[3] Michael Egorov. 2019. StableSwap - Efficient Mechanism for Stablecoin Liquidity. Retrieved Feb 24, 2021, from https://www.curve.fi/stableswap-paper.pdf
[4] Allan Niemerg, Dan Robinson, and Lev Livnev. 2020. YieldSpace: An Automated Liquidity Provider for Fixed Yield Tokens. Retrieved Feb 24, 2021, from https://yield.is/YieldSpace.pdf
[5] Abraham Othman. 2012. Automated Market Making: Theory and Practice. Ph.D. Dissertation. Carnegie Mellon University

DISCLAIMER

This paper is for general information purposes only. It does not constitute investment advice or a recommendation or solicitation to buy or sell any investment and should not be used in the evaluation of the merits of making any investment decision. It should not be relied upon for accounting, legal or tax advice or investment recommendations. This paper reflects current opinions of the author and is not made on behalf any other parties and does not necessarily reflect the opinions of individuals associated with the author or tooling used. The opinions reflected herein are subject to change without being updated.