Exploring Dynamic Fee Mechanisms in Automated Market Makers: A Pool-Internal Data Approach

Carl Schmidt and Yanis Berkani

Automated Market Makers (AMMs) enable permissionless liquidity provision and trading without traditional order books Adams et al. (2021). Current implementations use static fee tiers that cannot capture opportunities to optimize liquidity attraction and fee revenue in a changing market environment.

Recent research has explored dynamic fee mechanisms to address these limitations. Milionis et al. (2022) demonstrated that loss-versus-rebalancing creates fundamental challenges for liquidity providers, while Cartea et al. (2023) showed that optimal liquidity provision requires careful consideration of predictable loss dynamics. Existing approaches often rely on external oracles or complex mathematical models that may be impractical for solidity implementation.

We explore a dynamic fee algorithm that uses pool-internal data, leveraging volume-to-TVL ratios as the signal for fee adjustments. This approach offers potential advantages: (1) no dependence on external oracles, (2) competitiveness regardless of market activity, and (3) computational simplicity.

Market Analysis and Data

Our analysis utilizes historical on-chain AMM data from November 2023 to September 2025, focusing on larger-cap tokens to provide broader market insights. For this paper, we will illustrate results using ETH/USDC across multiple fee tiers (0.05%, 0.30%, 1.00%). Figure 1 shows the market-demanded fee over time, revealing clear patterns that suggest opportunities for dynamic optimization.

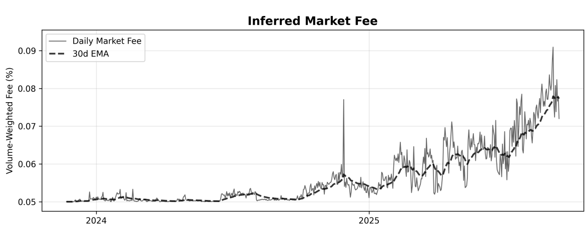

Fig. 1. Volume-weighted market fee for ETH/USDC showing temporal variation in market-demanded fees.

Our focus is on the most liquid pools and fee structures. In ETH/USDC pools on Uniswap v2 and v3 on Base, the 0.05% tier dominates both trading volume (77.2% market share) and TVL. The 0.30% and 1.00% tiers maintain 21.9% and 0.7% volume share, respectively.

Algorithm Design

The tested dynamic fee algorithm operates on three core principles: (1) volume-to-TVL ratio monitoring, (2) exponential moving average target tracking, and (3) bounded fee adjustments. On pool initialization the parameters of the algorithm are defined by following a pool-type dependent setup, which follows historical fee and ratio data. Additionally, the EMA is preloaded with data compiled from leading pools with identical profiles.

The algorithm works by calculating the current volume-to-TVL ratio and comparing it against a dynamic target ratio that evolves via exponential moving average. When the current ratio deviates significantly (meaning is above tolerance) from the target, the algorithm adjusts fees to restore balance: if volume is too high relative to TVL (indicating underpricing), fees increase; if volume is too low (indicating overpricing), fees decrease. This happens randomly once every 24-27h. We are currently investigating whether pool-type dependent minimum periods should be used.

The core algorithm can be expressed mathematically as:

$$\mathrm{fee}_{t+1} = \mathrm{fee}_t + \delta \cdot \mathrm{sign}(\mathrm{ratio}_t - \mathrm{target}_t) \quad [1]$$

$$\mathrm{target}_{t+1} = \alpha \cdot \mathrm{ratio}_t + (1-\alpha) \cdot \mathrm{target}_t \quad [2]$$

$$\delta = \min\left(\frac{|\mathrm{ratio}_t - \mathrm{target}_t|}{\mathrm{target}_t} \cdot \mathrm{slope}, \mathrm{max\_step} \right) \quad [3]$$

Where the 30-day lookback period provides responsive yet stable target ratio evolution, and fee adjustments incorporate asymmetric multipliers enabling faster downward corrections (2.0x) than upward adjustments (1.0x). The production parameter configuration detailed in Table 1 reflects optimization for practical deployment as laid out in the following section.

Table 1. Algorithm Parameters

Parameter	Value	Description
Min Fee	0.99%	Lower bound
Max Fee	100.01%	Upper bound
Lookback Period	30 days	EMA window
Ratio Tolerance	5.0%	Deviation threshold
Base Max Fee Delta	50 bps	Base adjustment step
Linear Slope	1.0	Proportional response
Upper Side Factor	1.0	Fee increase multiplier
Lower Side Factor	2.0	Fee decrease multiplier
Min Period	1 day	Update frequency

Standard Pool Type refers to blue-chip pairs with high liquidity, like ETH/USDC and WBTC/USDC.

Significance Statement

Dynamic fee algorithms allow the adjustment of trading fees based on market conditions, potentially improving capital efficiency. We explore a pool-internal approach using volume-to-TVL ratios for fee adjustments.

Parameter selection was done by optimizing across multiple dimensions: lookback periods (7-30 days), ratio tolerances (1%-10%), fee deltas (25-200 basis points), asymmetric multipliers (1.0-3.0), and linear slope coefficients (0.5-2.0). The optimization leveraged multi-asset validation across pairs of the investigated pool-type with cross-validation to ensure stability and competitiveness across market conditions. The framework optimized a score based on average TVL, LP revenue generation, and fee volatility. The final parameter set reflects deployment constraints and the goal of not overfitting to historical data.

Simulation Framework

To evaluate the dynamic fee algorithm, we developed an agent simulation framework with behavioural models for both Liquidity Providers and Traders. Our approach models traders as cost-minimizing agents that consider fees and slippage, while liquidity providers maximize revenue with some capital stickiness reflecting switching costs and a 'set-and-forget' mentality.

The behavioural models demonstrate strong predictive performance on historical data, as shown in Table 2. The R^2 scores indicate how well each model explains observed market behaviour, with both models showing high explanatory power for standard pools.

Table 2. Behavioural Model R^2 Values

Pool	LP Model R^2	Trader Model R^2
Standard Pool Index	98.6%	99.6%

R^2 scores measure the quality of fit between predicted and actual behaviour

We acknowledge the inherent complexity of modelling real-world behaviour. While the models are predictive in backtesting scenarios, the simplified behaviour cannot capture market inefficiencies, institutional behaviour, or sophisticated trading strategies.

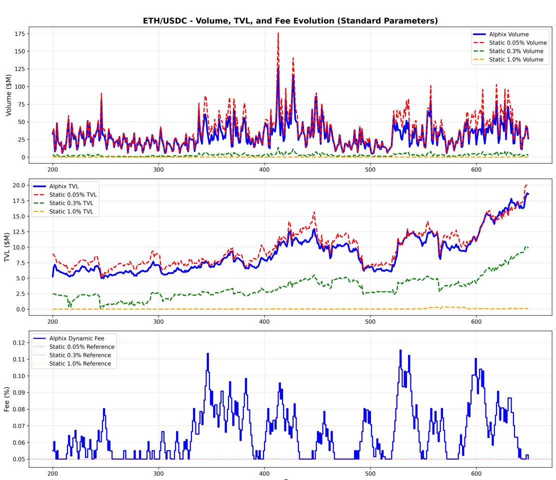

Fig. 2. Simulation framework results showing dynamic interactions between volume allocation, TVL flows, and fee adjustments over time. The findings reveal

Figure 2 shows the performance of volume and TVL as well as the dynamic fee evolution over the simulation time horizon of 664 days using the Standard Pool parameters and prior 30d of historical data to initialize the algorithm.

The results suggest potential for dynamic fee optimization, though the extensive model refinement required highlights the challenge of accurately predicting real-world performance.

Conclusion

Our dynamic fee algorithm offers advantages in addressing static AMM limitations. The pool-internal approach reduces external dependencies at no cost to smart contract complexity. However, our simulation development process revealed implementation challenges that require careful consideration before launching a live product.

Implementation considerations extend beyond algorithmic design to include gas optimization, potential manipulation vectors, and integration complexity. Our simulation work highlighted the sensitivity of outcomes to behavioural modelling assumptions.

The computational framework developed provides a foundation for continued research, though substantial validation remains necessary before practical deployment.

References

Adams, H., Zinsmeister, N., Salem, M., Keefer, R., and Robinson, D. (2021). Uniswap v3 core. *Uniswap Labs*.

Cartea, Á., Drissi, F., and Monga, M. (2023). Decentralised finance and automated market making: Predictable loss and optimal liquidity provision. *SIAM Journal on Financial Mathematics*.

Milionis, J., Moallemi, C. C., Roughgarden, T., and Zhang, A. L. (2022). Automated market making and loss-versus-rebalancing. *arXiv:2208.06046*.