

A photograph of four individuals in a laboratory or office environment. A man in a dark suit and white shirt is leaning over a table, looking down at some papers. To his right, a woman in a light blue shirt is also looking down at the table. In the foreground, another man in a light blue shirt is partially visible, looking towards the right. On the far right, a woman with curly hair is looking towards the center. The background shows shelves with various items and equipment.

# SIMULATED AMMs

---

## Under Chaotic Scenarios

*Alessandro Tricca  
Jonathan Cerullo Uyi*

# SIMULATOR ARCHITECTURE

---

## Uniswap v2 - Constant Product Market Maker

*Robust pioneer AMM, but with high slippage on large trades*

### Formula:

$$x \cdot y = k$$

## Constant Sum Market Maker

*A linear model optimized for equal-value assets like stablecoins, but catastrophically unstable for volatile pairs*

### Formula:

$$x + y = k$$

## Curve StableSwap

*A hybrid approach blending both formulas to achieve low slippage near price parity while maintaining stability*

### Formula (simplified):

$$k = (x \cdot y)^\alpha \cdot (x + y)^{1-\alpha}, \alpha = \frac{A}{A+1}$$

# MARKET AGENTS

---

## Retail Trader

*Performs random buy/sell actions based on the available wallet balance*

## Smart Trader

*Avoids trades with excessive slippage*

## Arbitrageur

*Acts as the market's stabilizer*

## Liquidity Provider

*Introduces fragility to the system*

## Whale Trader

*Capable of moving the market with a single action*

# SIMULATION DYNAMICS

```
AMM SIMULATION MENU
Current AMM: UNISWAP
=====
1. Go forward 1 step
2. Go forward N steps
3. Apply market shock
4. Current status
5. Show graphs
6. Save results
7. See statistics
8. Whale dump
9. Whale pump
0. Exit
=====
Choice: |
```

Main CLI



## The Main Interface (main.py)

Provides a Command Line Interface (CLI) that gives the user full control on the market via a simple text menu.

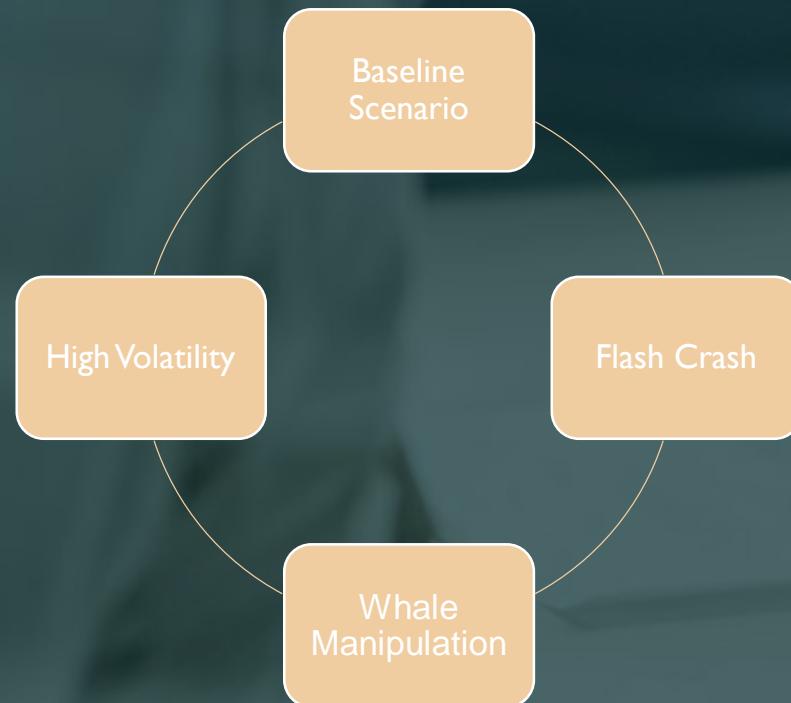


## The Simulation Class (run\_simulation.py)

Initialize the markets and execute the full simulation lifecycle, while recording data for analysis.

# EXPERIMENTAL DESIGN

We designed four distinct stress-testing scenarios to evaluate our AMMs



## Baseline Scenario

Simulates normal market condition

## Flash Crash

Simulates a sudden 50% price drop

## Whale Manipulation

Executes large-scale pump-and-dump trades

## High Volatility

Simulates prolonged market chaos by applying sustained extreme volatility

# METRICS

---



## Price Gap

- Divergence between AMM and market price

## Price Stability

- Standard deviation of AMM price

## Arbitrageur Profit

- Total gains from price corrections

## LP Panic Events

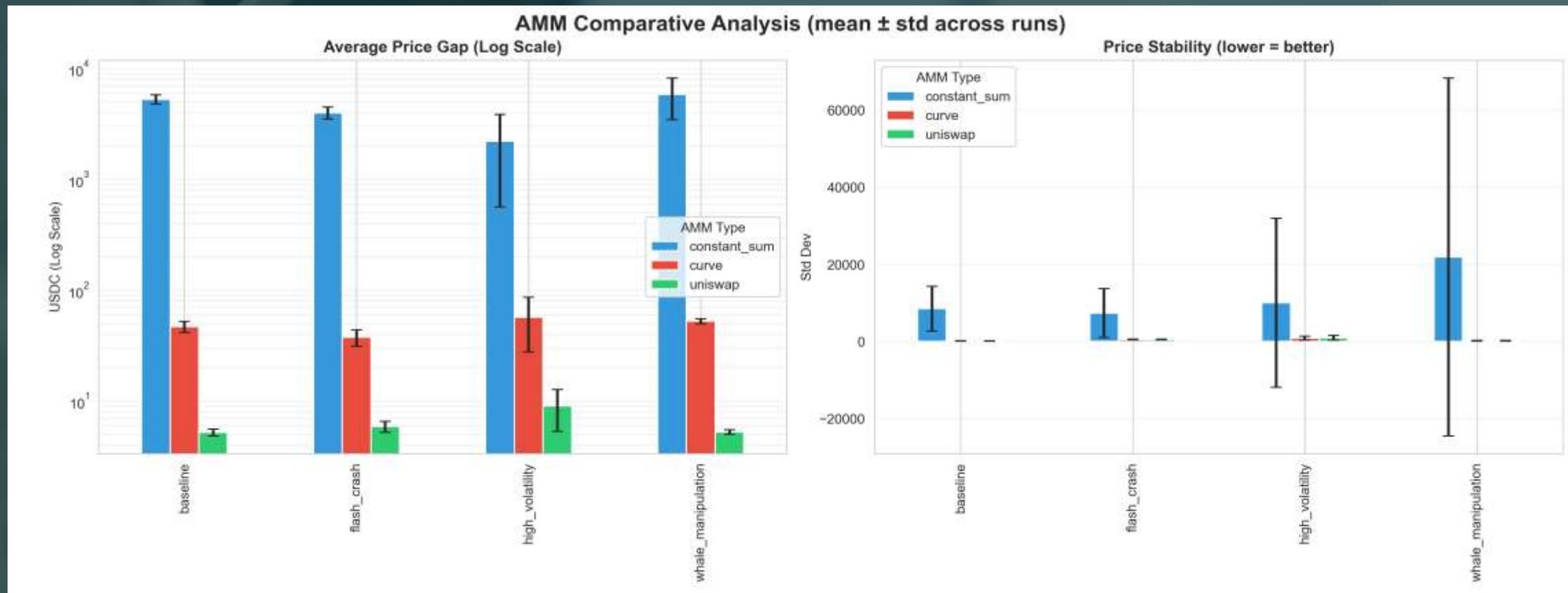
- Frequency of liquidity withdrawals

## k Growth

- Fee accumulation rate (pool health)

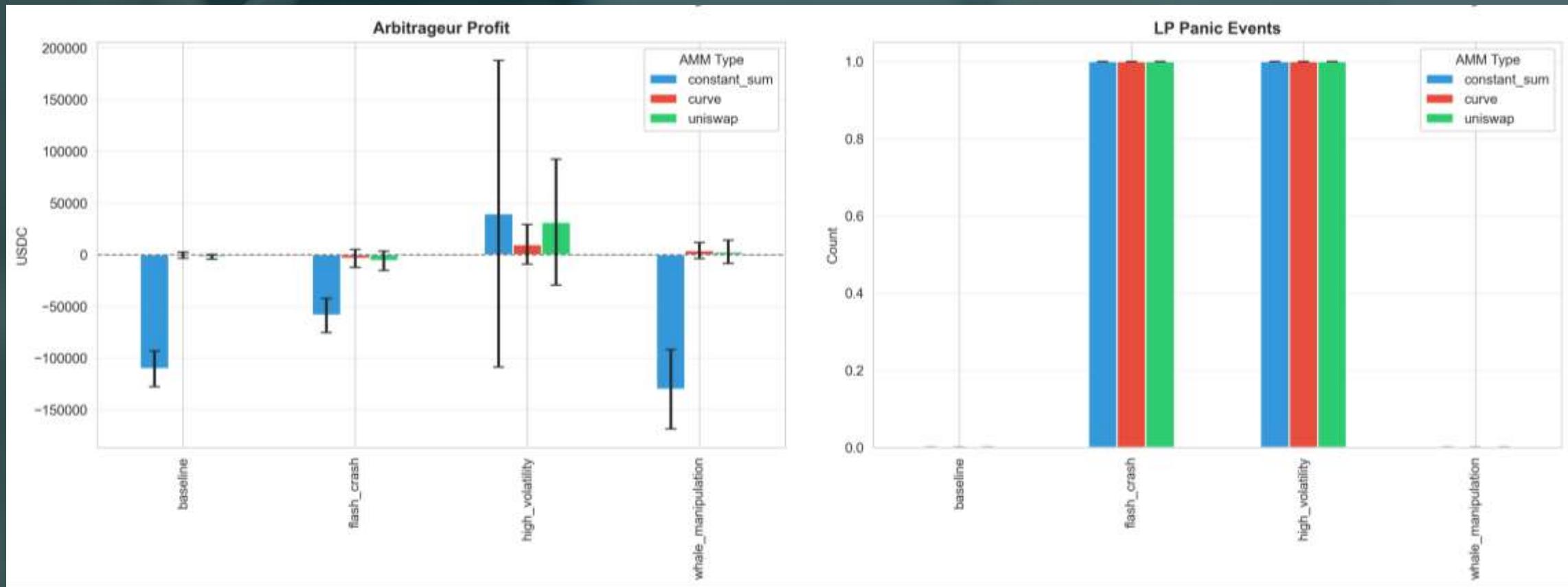
# RESULTS

## AMM Comparative Analysis



# RESULTS

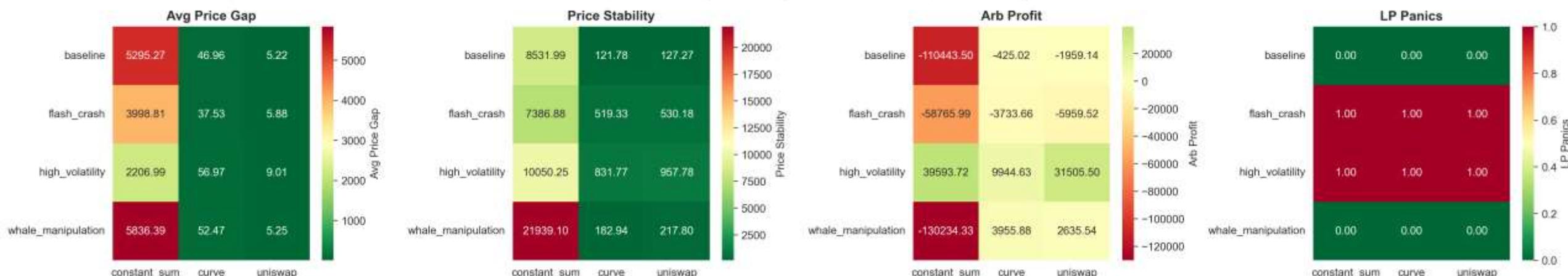
## AMM Comparative Analysis



# RESULTS

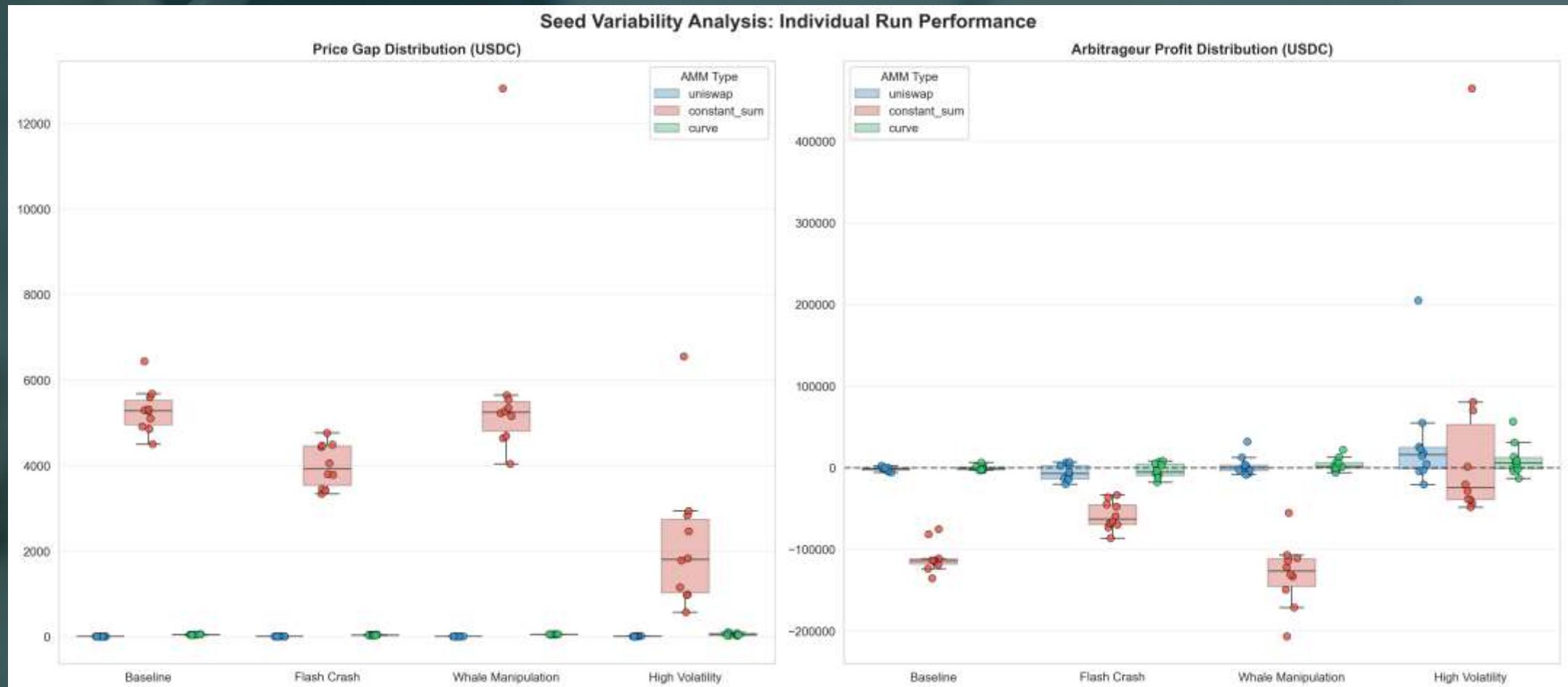
## Performance Heatmap

Performance Heatmaps: Uniswap vs Constant Sum (mean values)



# RESULTS

## Variability Analysis



# CONCLUSIONS

---

## Uniswap v2

- Best for Volatile Pairs
- Lowest price gaps across all scenarios
- Consistent arbitrage opportunities

## Curve Stableswap

- Moderate Efficiency
- Price gaps 9 $\times$  worse than Uniswap, but 113 $\times$  better than CSMM
- Consistent arbitrage opportunities

## Constant Sum

- Catastrophic Failure
- Price gaps 1014 $\times$  worse than Uniswap
- High instability
- Inconsistent arbitrage opportunities

# Thank you!

---

-  *Alessandro Tricca*
-  *Jonathan Cerullo Uyi*