

Analysis of Market Manipulation on the Solana Blockchain with a Focus on Wash Trading

Daniel Wiedenmann

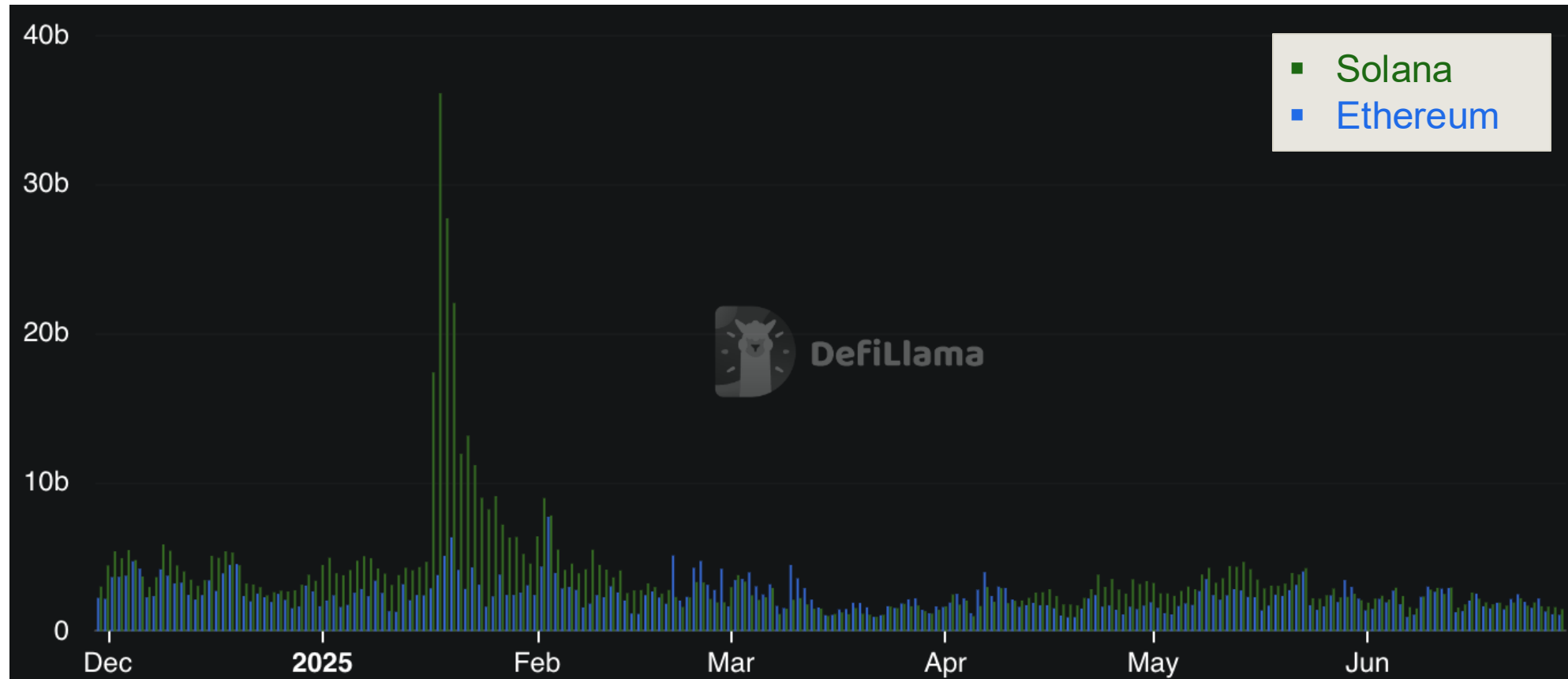
June 30th 2025, Thesis Final Presentation

Chair of Software Engineering for Business Information Systems (sebis)
Department of Computer Science
School of Computation, Information and Technology (CIT)
Technical University of Munich (TUM)
www.matthes.in.tum.de

1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion

Motivation

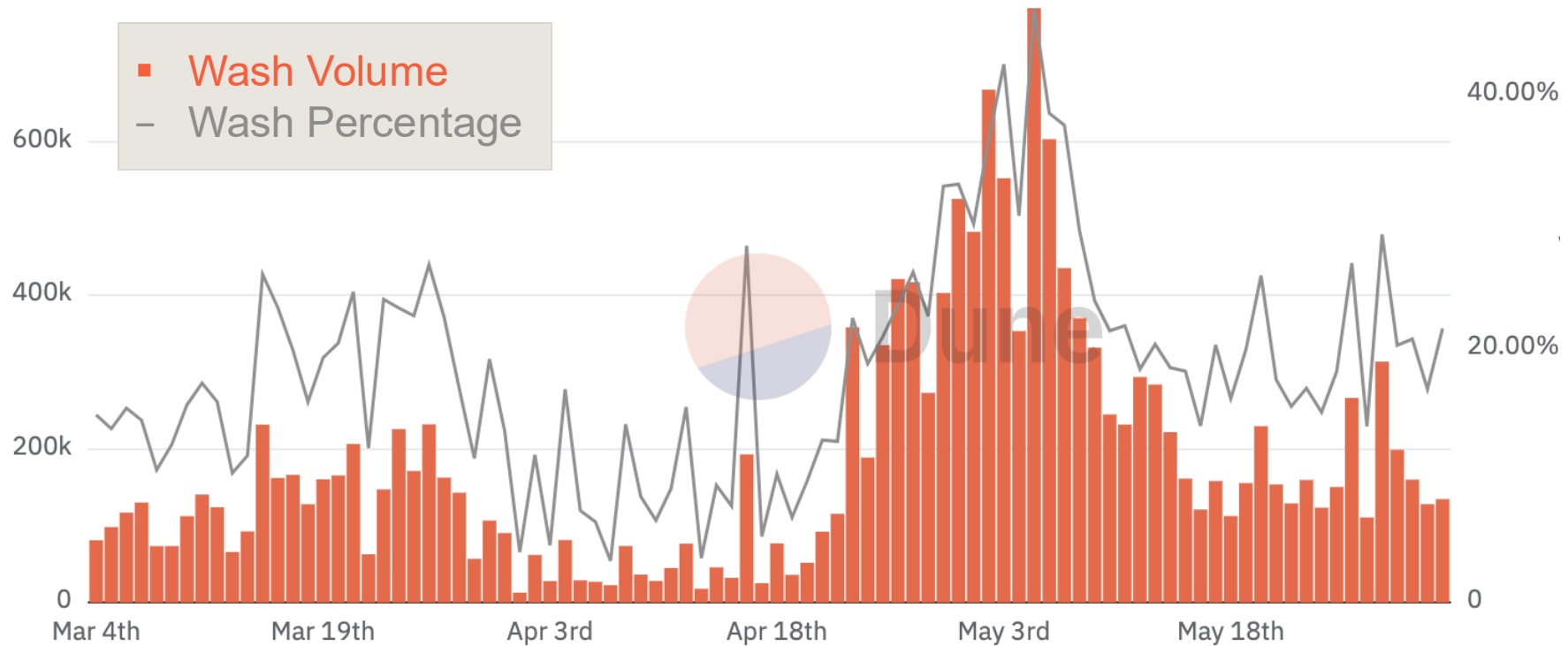
DEX Daily Trading Volume Comparison of Solana and Ethereum in USD



- Solana's DEX trading volume has been outperforming Ethereum's in the last year.
- Compared to other blockchains, Solana stands out for its exceptional speed and minimal transaction costs.

Motivation

Why is Wash trading a problem?



Solana NFT wash trading in USD

- Wash trading is a problem. This is quite well researched for the NFT Market.
- The chart above illustrates the problem of NFT wash trades conducted on the Solana blockchain.
- Wash trading of fungible Token on Solana is almost not researched.

Motivation

What is the reason for wash trading on fungible tokens?

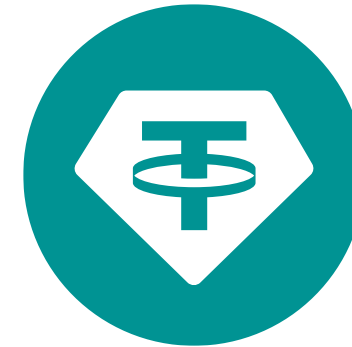
- Wash trading is a market manipulation practice where an entity simultaneously buys and sells the same financial instrument to create a false impression of activity or demand.
- Wash trading of fungible tokens (as shown on the right) is an under-researched area on the Solana blockchain.
- The primary incentive for wash trading coins is to artificially inflate their trading volume, making them appear more attractive.
- Example:
 - A creator launches a new coin on the Solana blockchain that lacks visibility.
 - To gain attention, they artificially boost the coin's trading volume, increasing its chances of being featured on exchange top trading lists.



Solana



Wrapped Ether



USDT



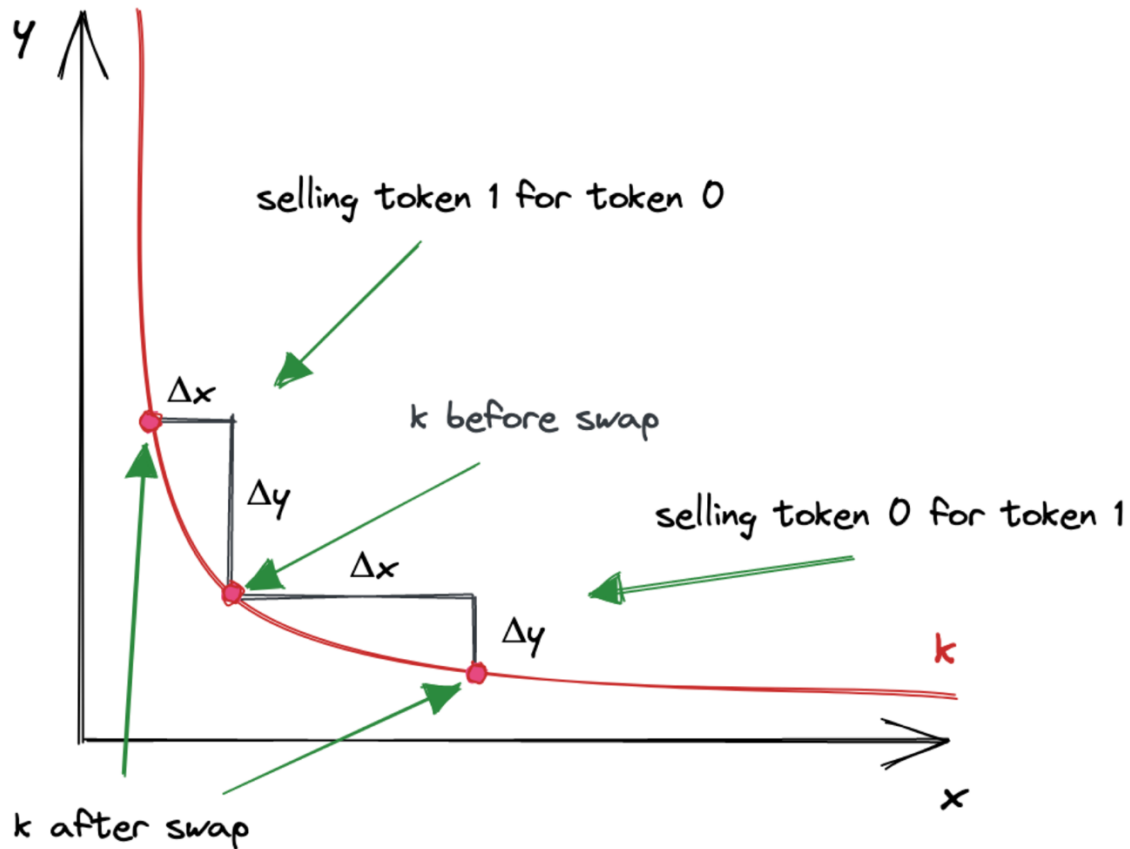
Jupiter



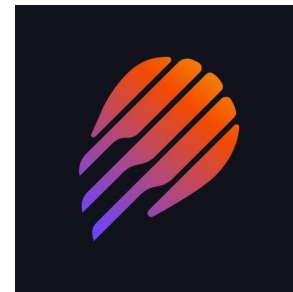
USDC

Decentralized Exchanges

Automated Market Maker



- Automated Market Makers use the product formula: $x \cdot y = k$
 - x = amount of token 0
 - y = amount of token 1
 - k = constant value (liquidity invariant)
- Traders move along the curve: adding token 0 removes token 1, and vice versa.
- Price depends on the token balance ratio in the pool $\Delta y / \Delta x$.



Meteora



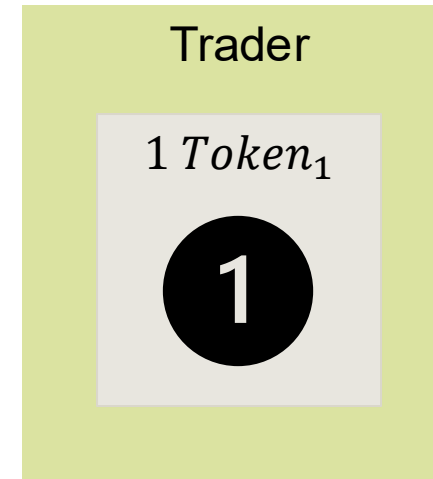
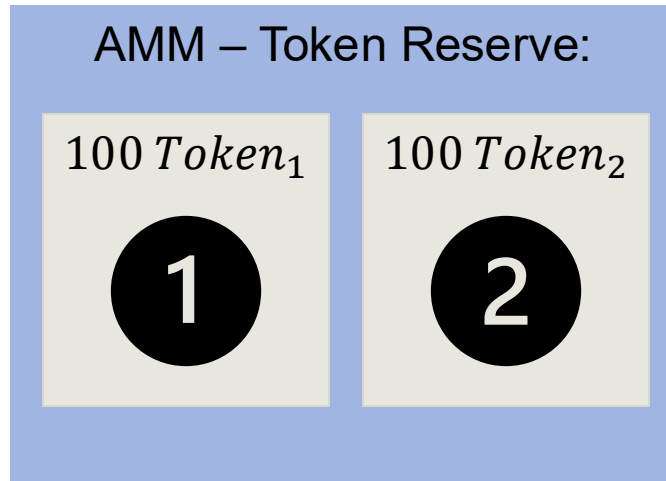
Raydium



Orca

Decentralized Exchanges

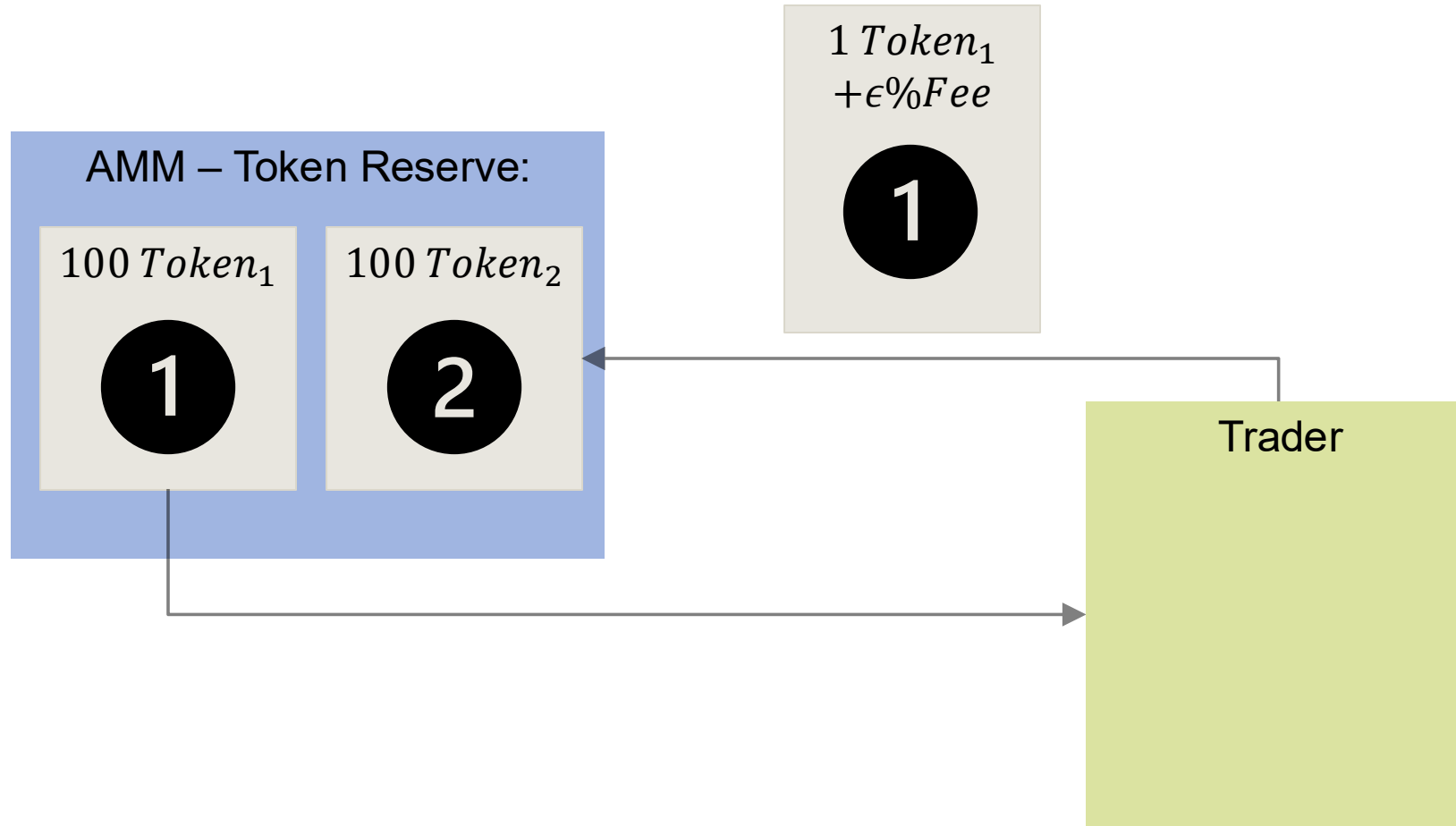
Example of a Trade on an Automated Market Maker (AMM)



- $100 \text{ Token}_1 * 100 \text{ Token}_2 = 10000$
- Price of each Token:
 - $\text{Token}_1 = \text{Token}_2 = 1$

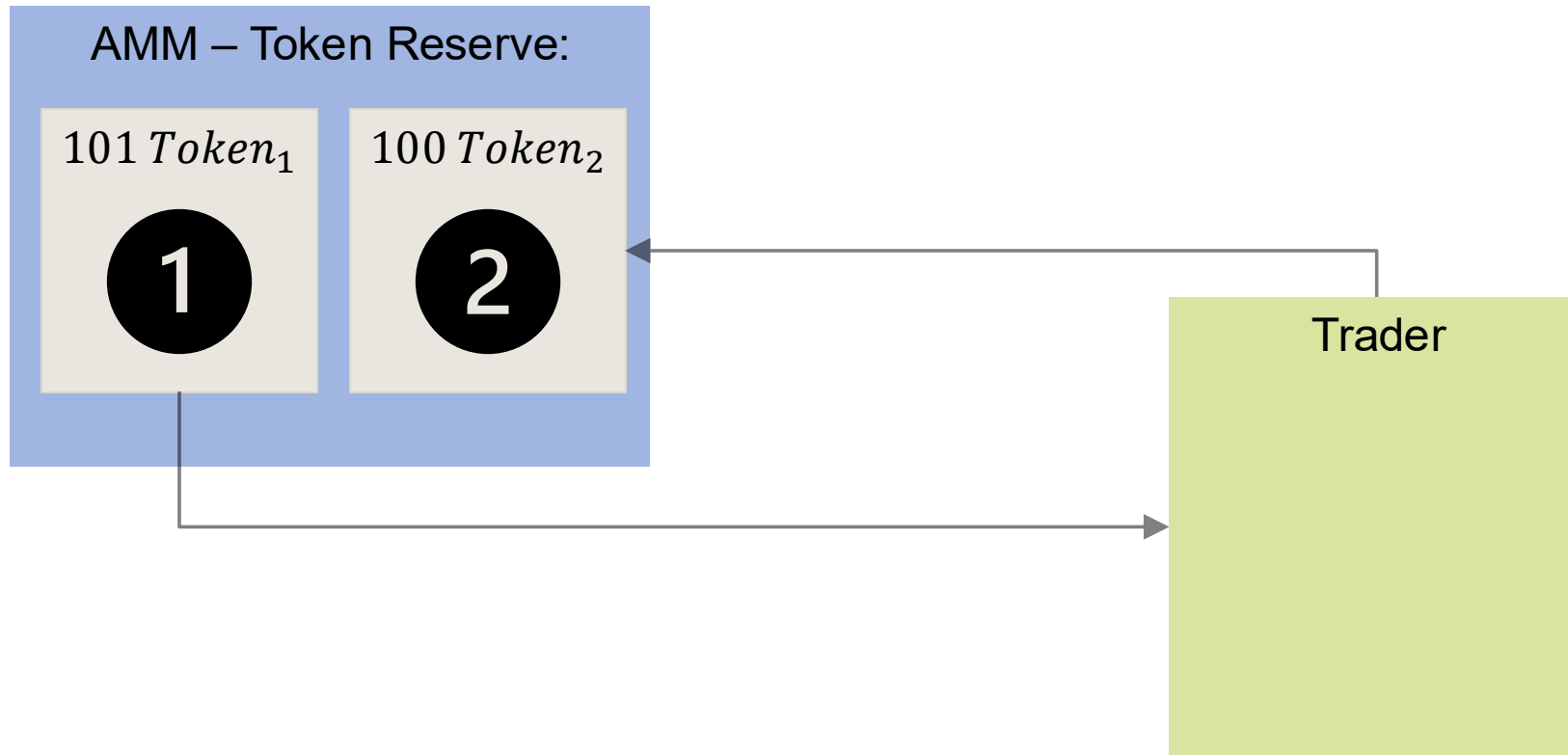
Decentralized Exchanges

Example of a Trade on an Automated Market Maker (AMM)



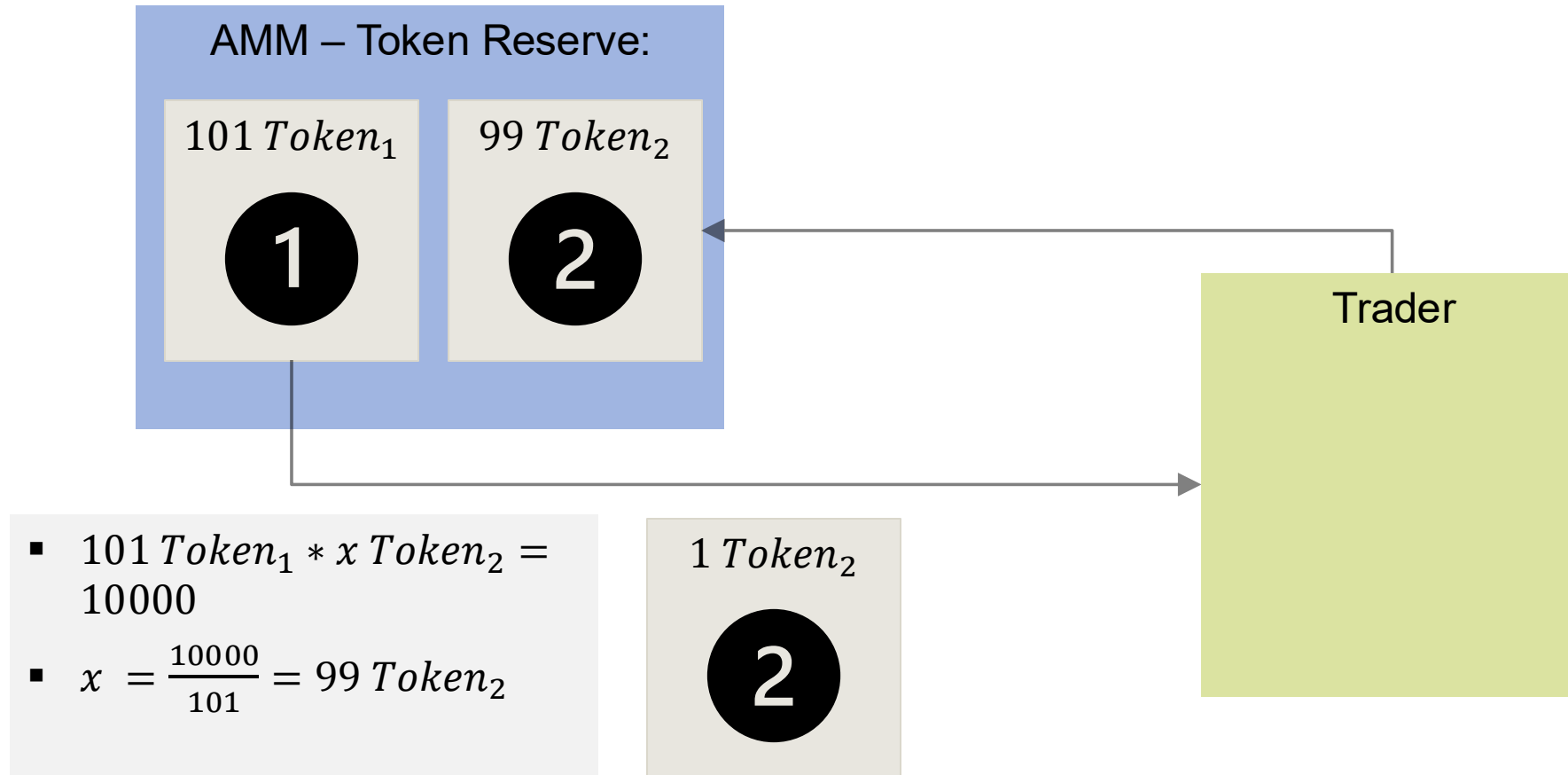
Decentralized Exchanges

Example of a Trade on an Automated Market Maker (AMM)



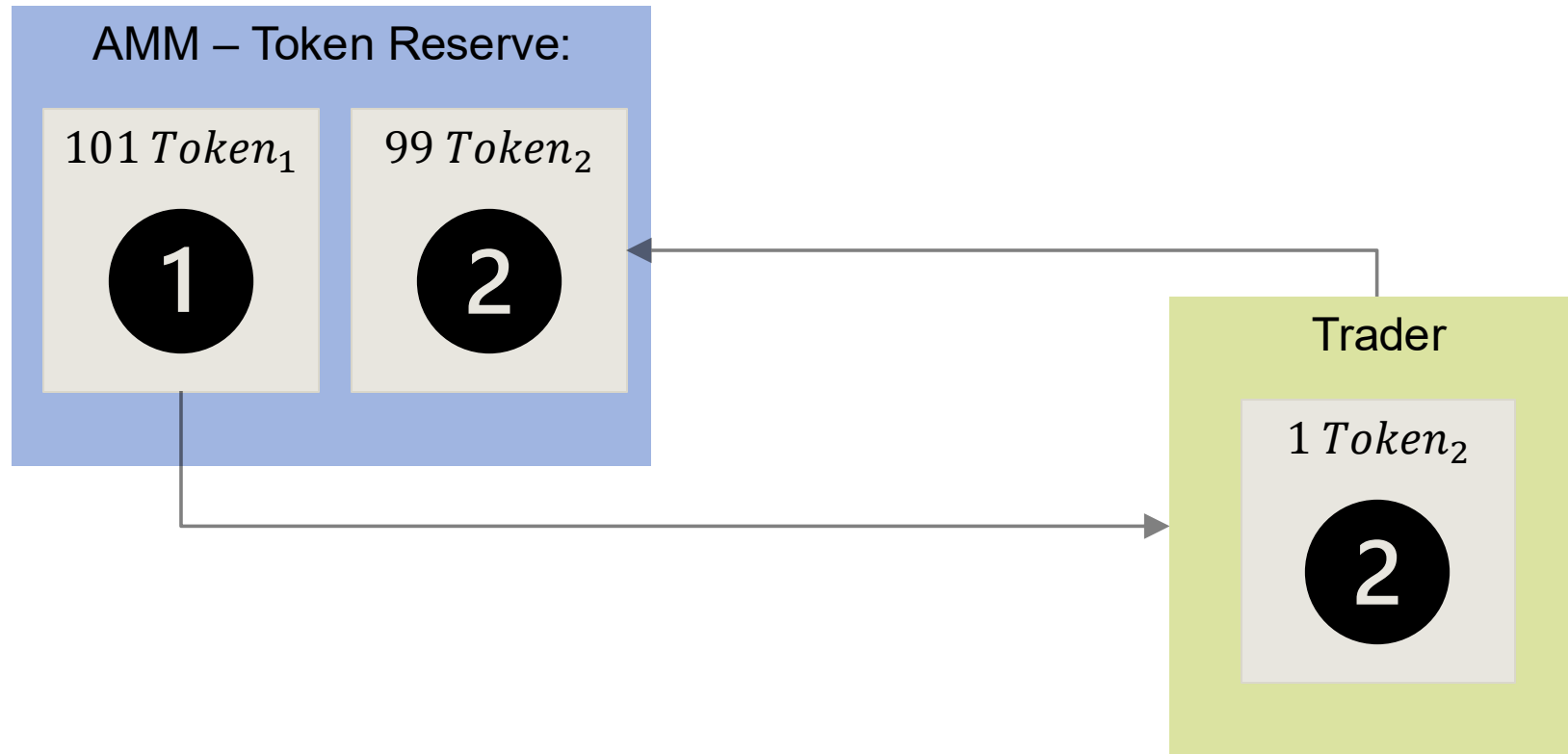
Decentralized Exchanges

Example of a Trade on an Automated Market Maker (AMM)



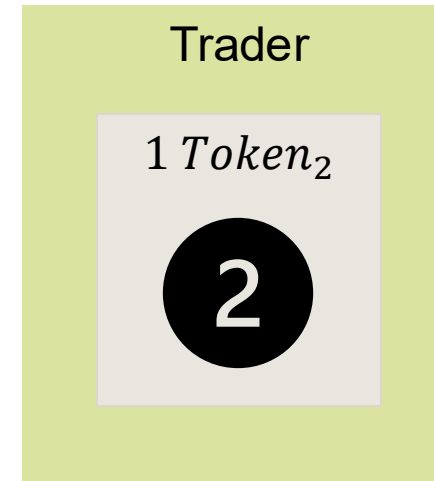
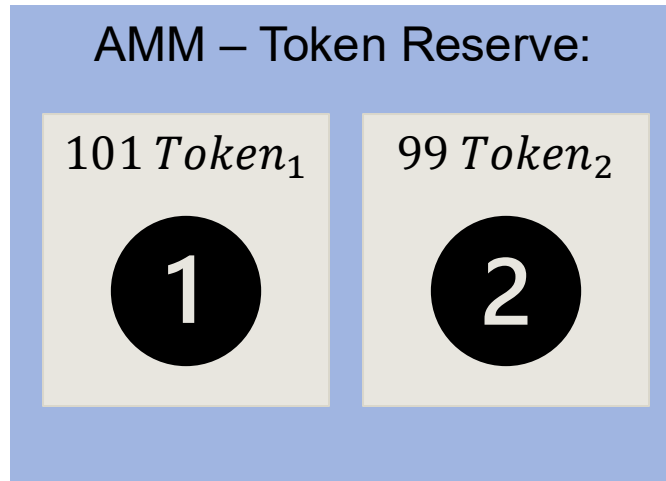
Decentralized Exchanges

Example of a Trade on an Automated Market Maker (AMM)



Decentralized Exchanges

Example of a Trade on an Automated Market Maker (AMM)



- $101 \text{ Token}_1 * 99 \text{ Token}_2 = 10000$
- Price of each Token:
 - $1 \text{ Token}_1 = \frac{99}{101} = 0.98 \text{ Token}_2$
 - $1 \text{ Token}_2 = \frac{101}{99} = 1.02 \text{ Token}_1$

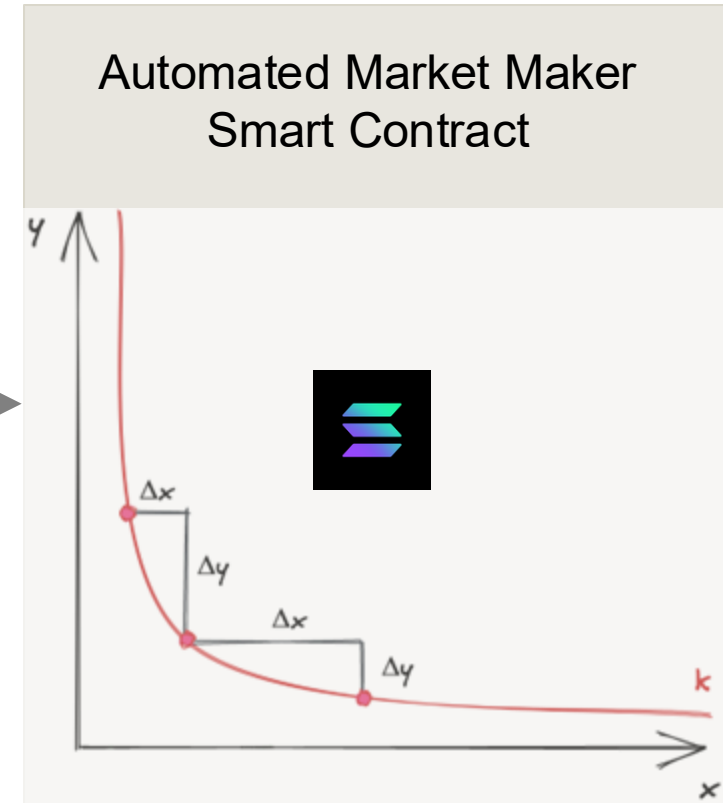
1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion

Problem Statement

How is Wash trading done?



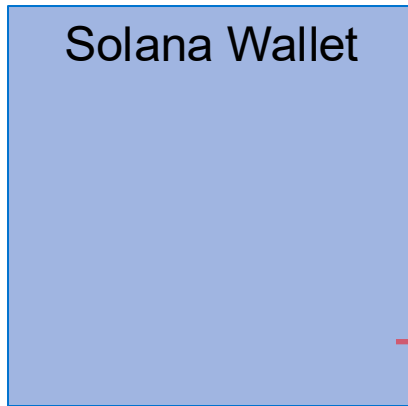
Wants to swap USDC for SOL



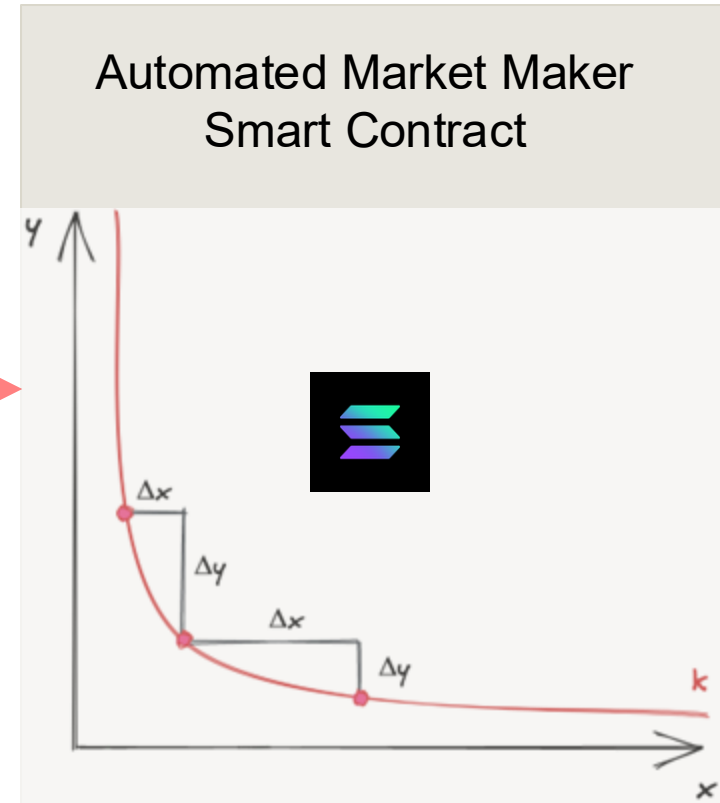
Problem Statement

How is Wash trading done?

Alice



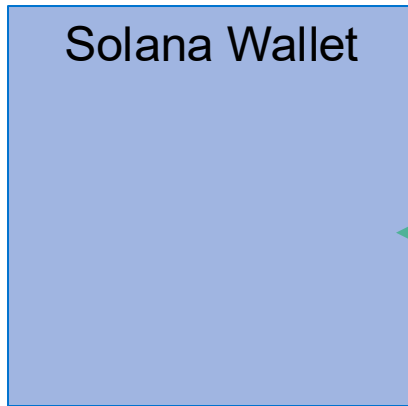
Sends USDC



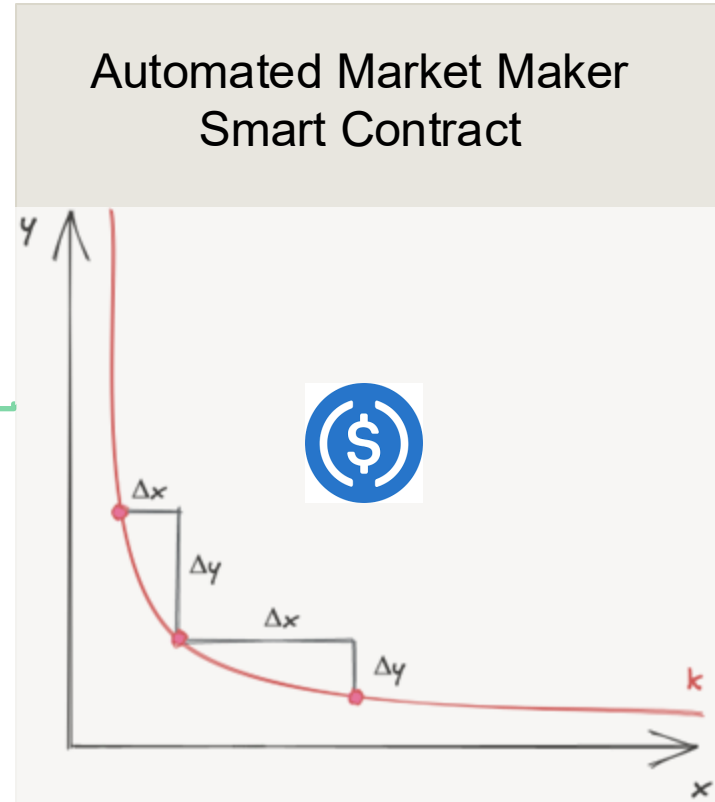
Problem Statement

How is Wash trading done?

Alice

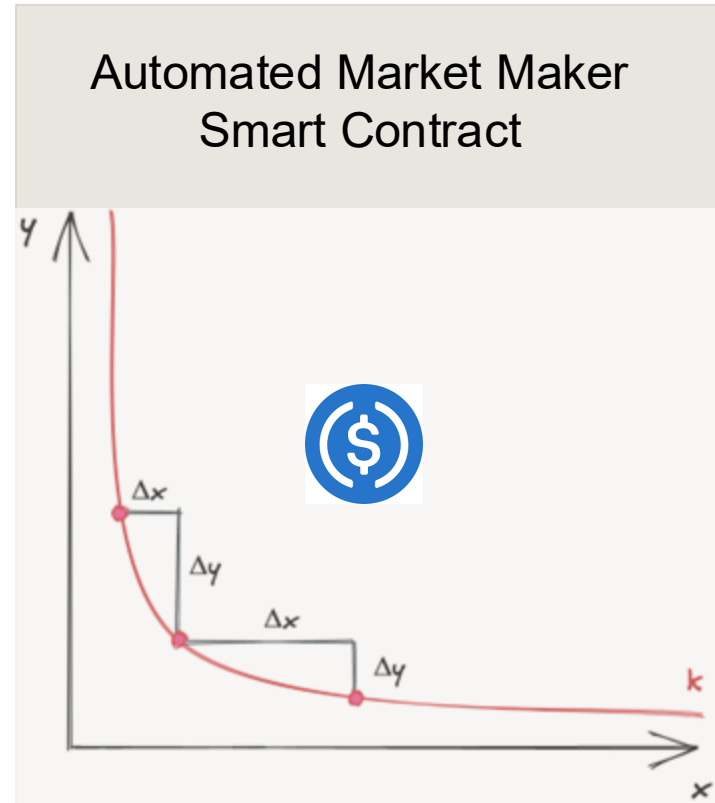
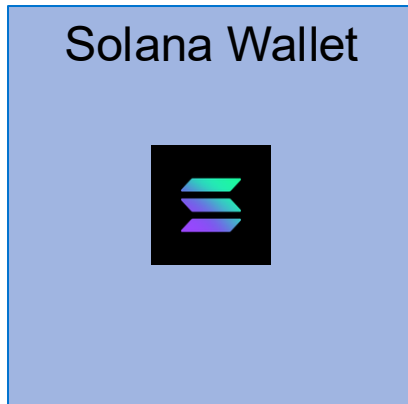


Receives SOL



Problem Statement

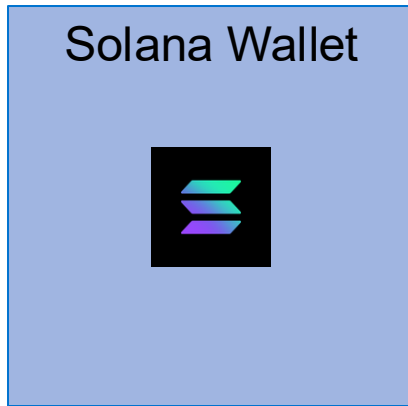
How is Wash trading done?



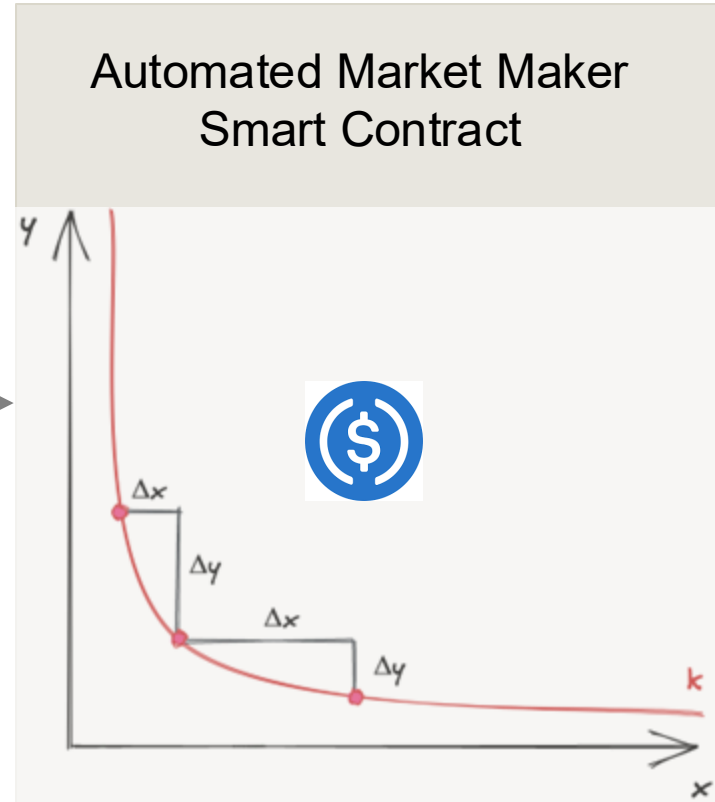
Problem Statement

How is Wash trading done?

Alice



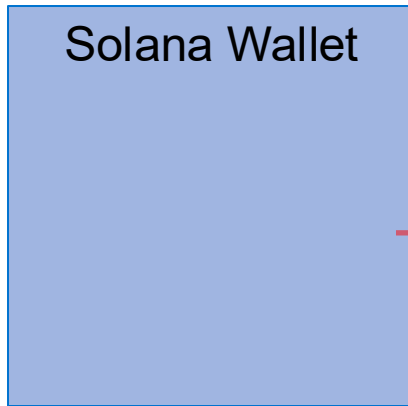
Wants to swap SOL for USDC



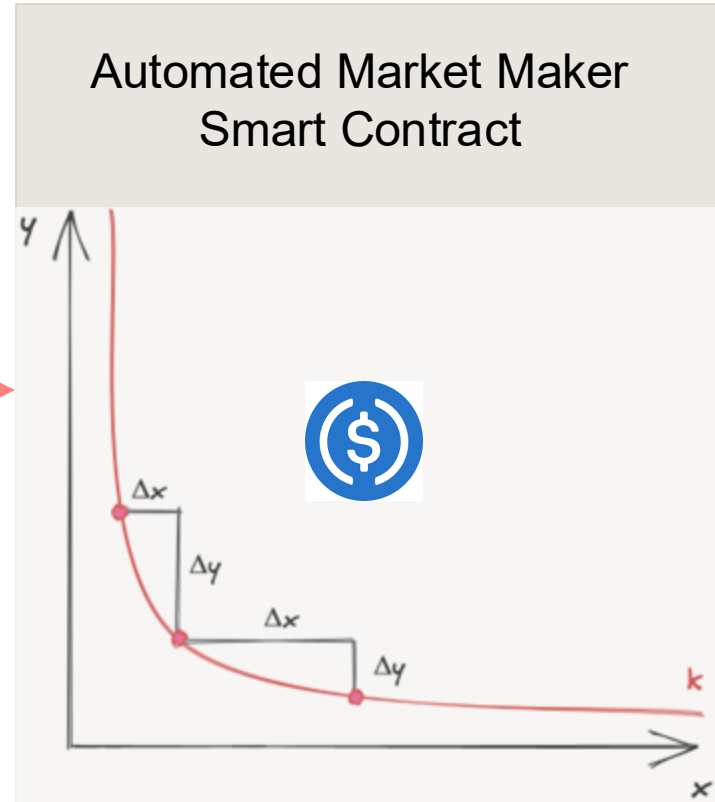
Problem Statement

How is Wash trading done?

Alice

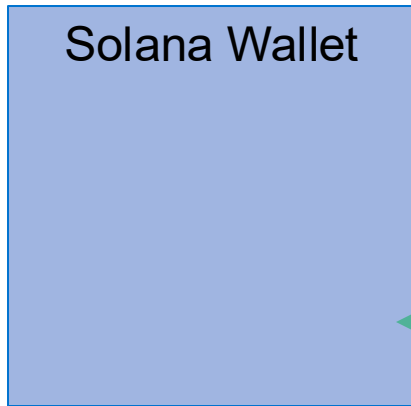


Sends SOL

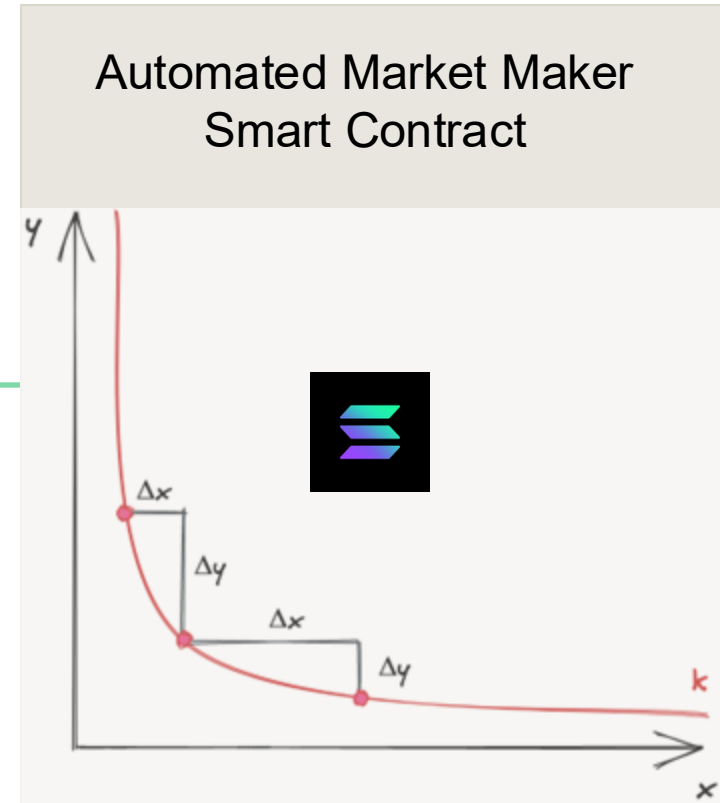


Problem Statement

How is Wash trading done?

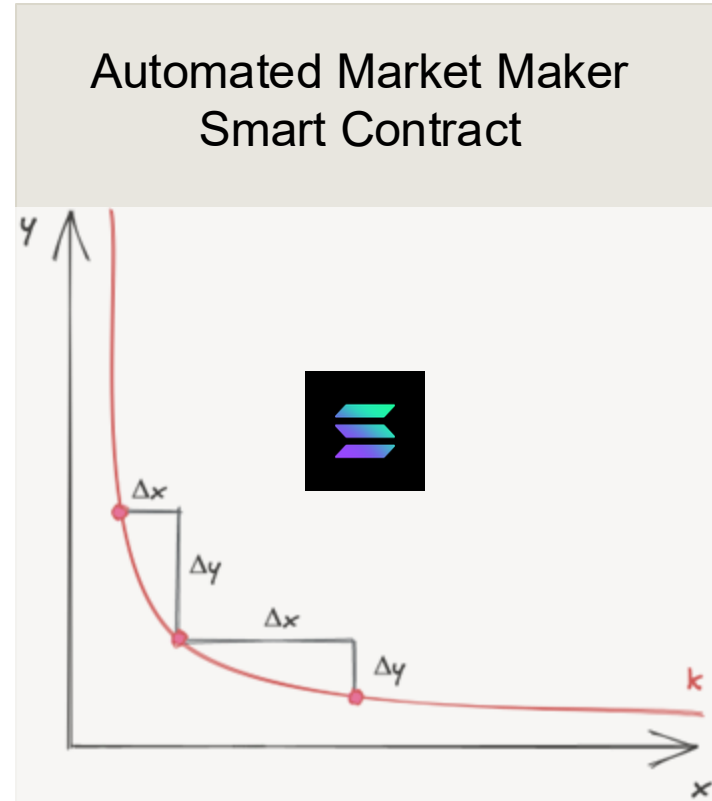


Receives USDC



Problem Statement

How is Wash trading done?



Problem Statement

How is Wash trading done?

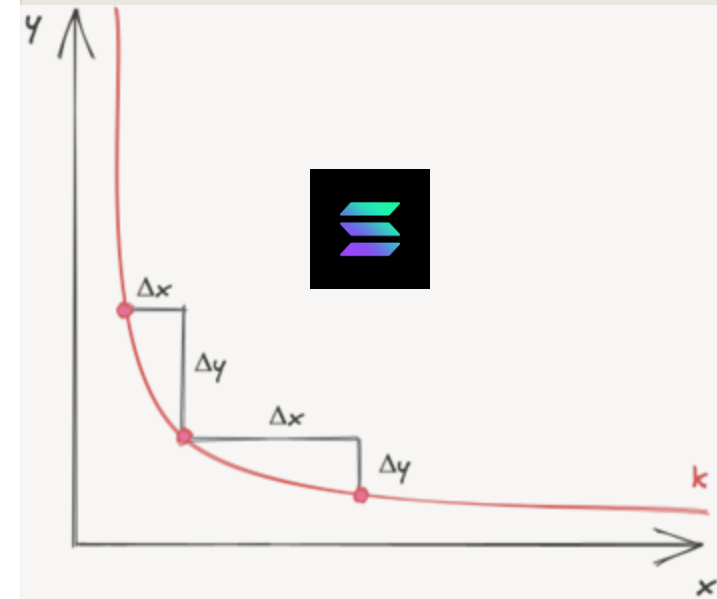
Trading Volume increased !



Alice



Automated Market Maker
Smart Contract



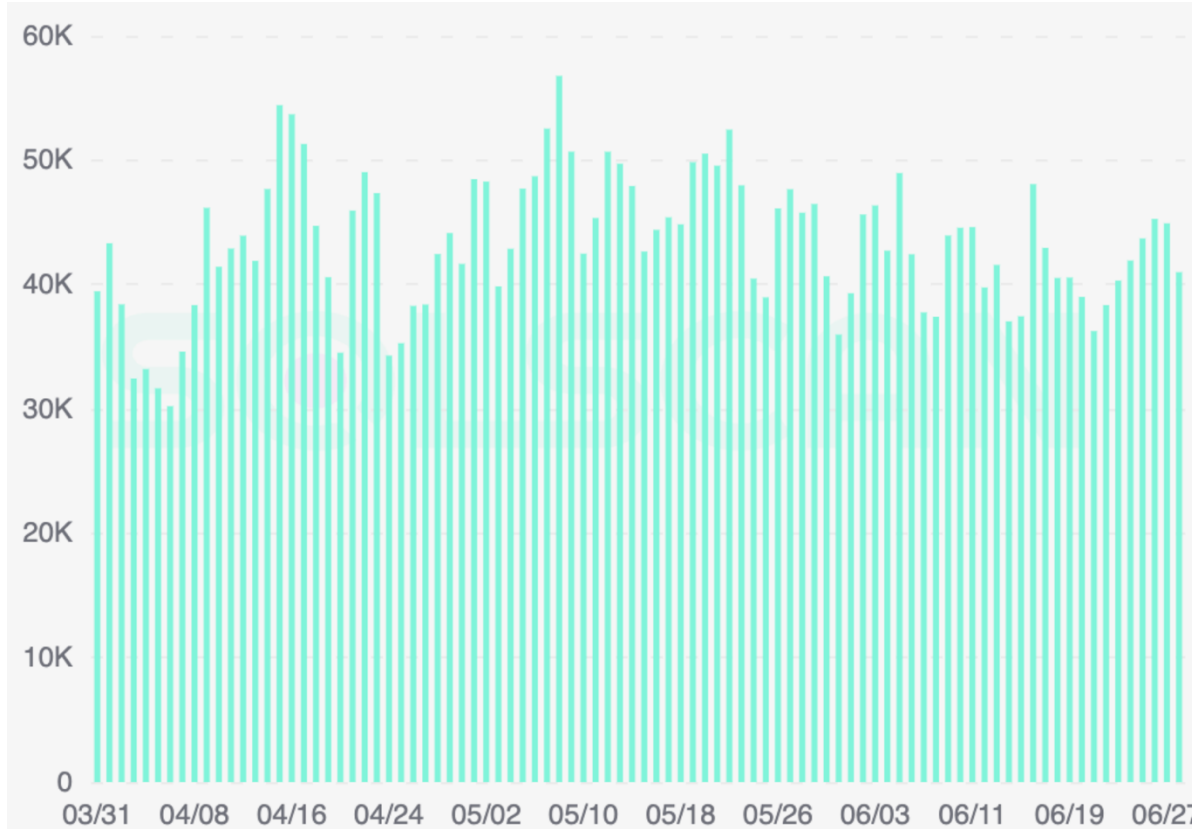
1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion

RQ 1: Assessment of current wash trading research on Solana

RQ 2: Development of a framework or a methodology to detect wash trades

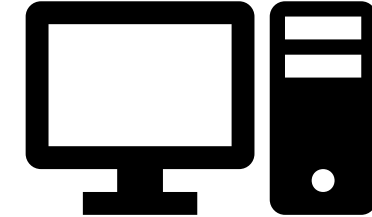
RQ 3: Quantification of wash trades on Solana

1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion



Daily SPL token launches of the last three months

- Focused on tokens with high wash trading potential, especially smaller ones where visibility boosts have more impact.
- Selected 15 tokens with ~1,500 transactions and 5 tokens with 15,000 to 1.5 million transactions.

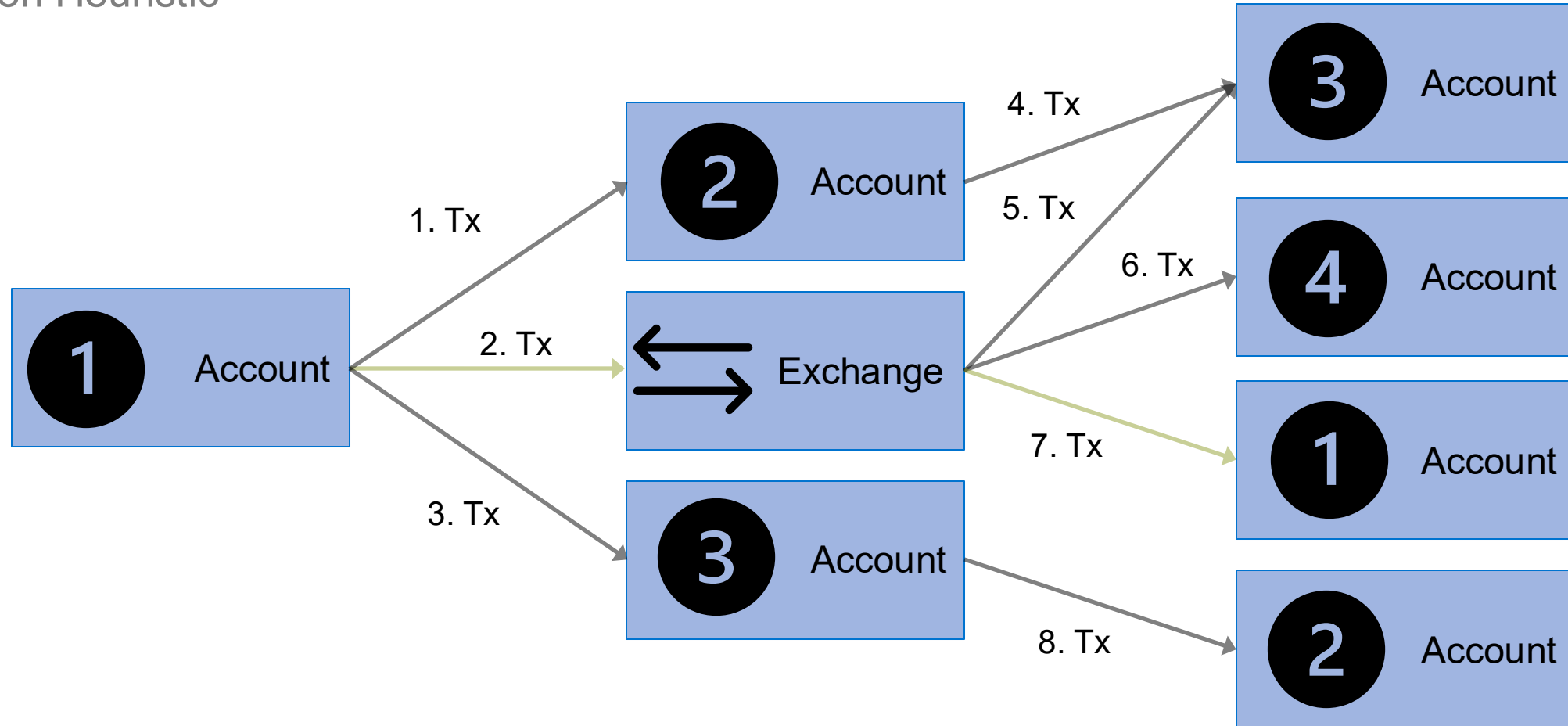


```
{  
  "info": {  
    ...,  
    "destination": "SPL-Market",  
    "source": "Maker-Address",  
    "amount": 1.0}  
  },  
  "type": "transfer"  
}
```



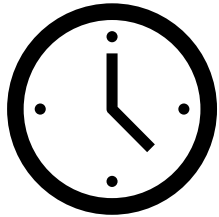
```
{  
  ...,  
  "type": "Sell",  
  "amount": 1.0,  
  "vault": "SPL-Market"  
  ...  
}
```

- Collected transactions from RPC providers (nodes dedicated to serving transaction data).
- Cleaned and structured trades by identifying direction and removing failed transactions and liquidity operations.



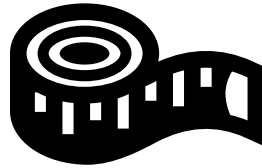
- Uses path-based depth-first search. It iterates through all outgoing edges (transactions) of a starting node (first account) and its underlying destinations until a transaction builds a cycle by returning to the starting node.
- Early stopping and constraints improve performance.

Transaction Time Window



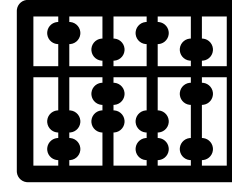
- The transactions within a wash trading cycle need to stay within a certain time.
- **Very strict:** 3min.
- **Strict:** 18min.
- **Lenient:** 1h.

Transaction Sequence



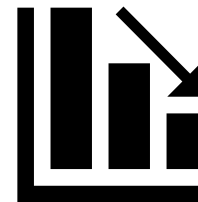
- Each transaction within a cycle should only have a maximum number of transactions between them.
- **Very strict:** 0 Tx.
- **Strict:** 5 Tx.
- **Lenient:** 20 Tx.

Amount Consistency



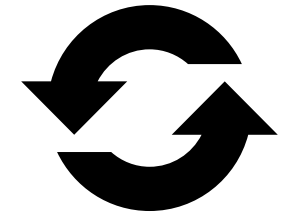
- The amounts match each other up to a small threshold.

Negativity



- The wash cycles are negative.

Repetition

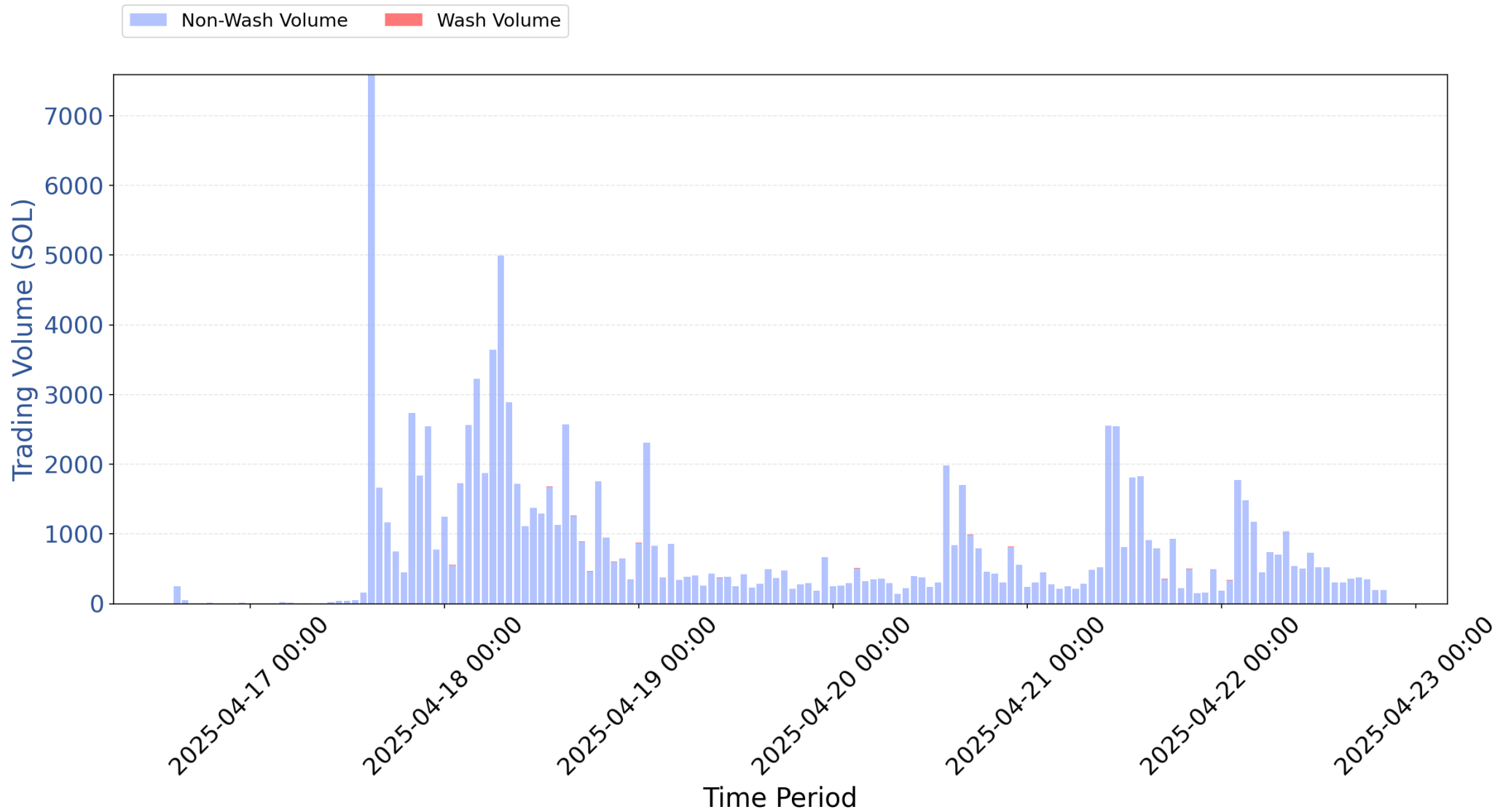


- The accounts conducting wash trades, should have at least two cycles.

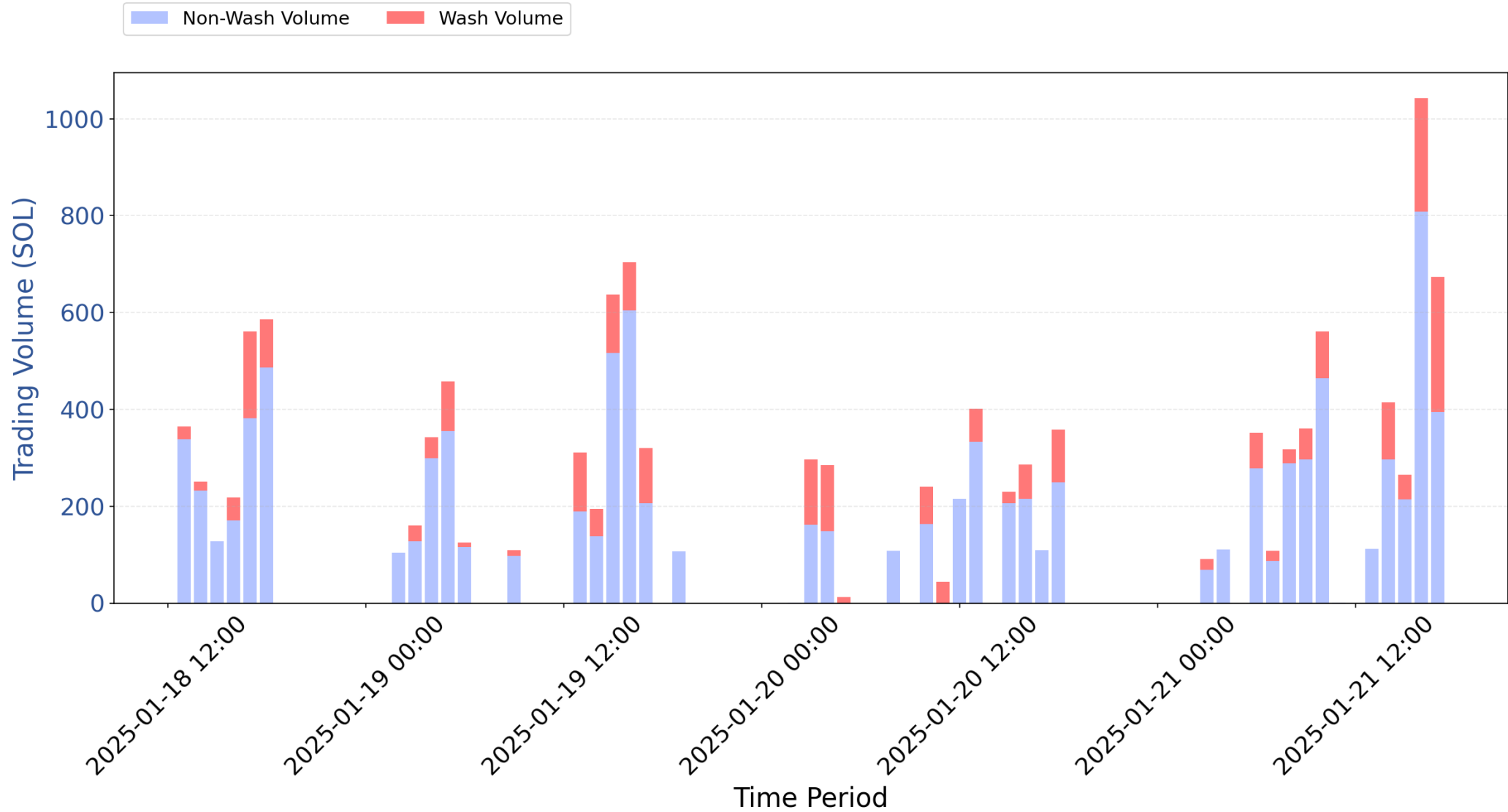
1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion

Results

Wash Trading of Token Over Time — Gorecats-Token (Strict)

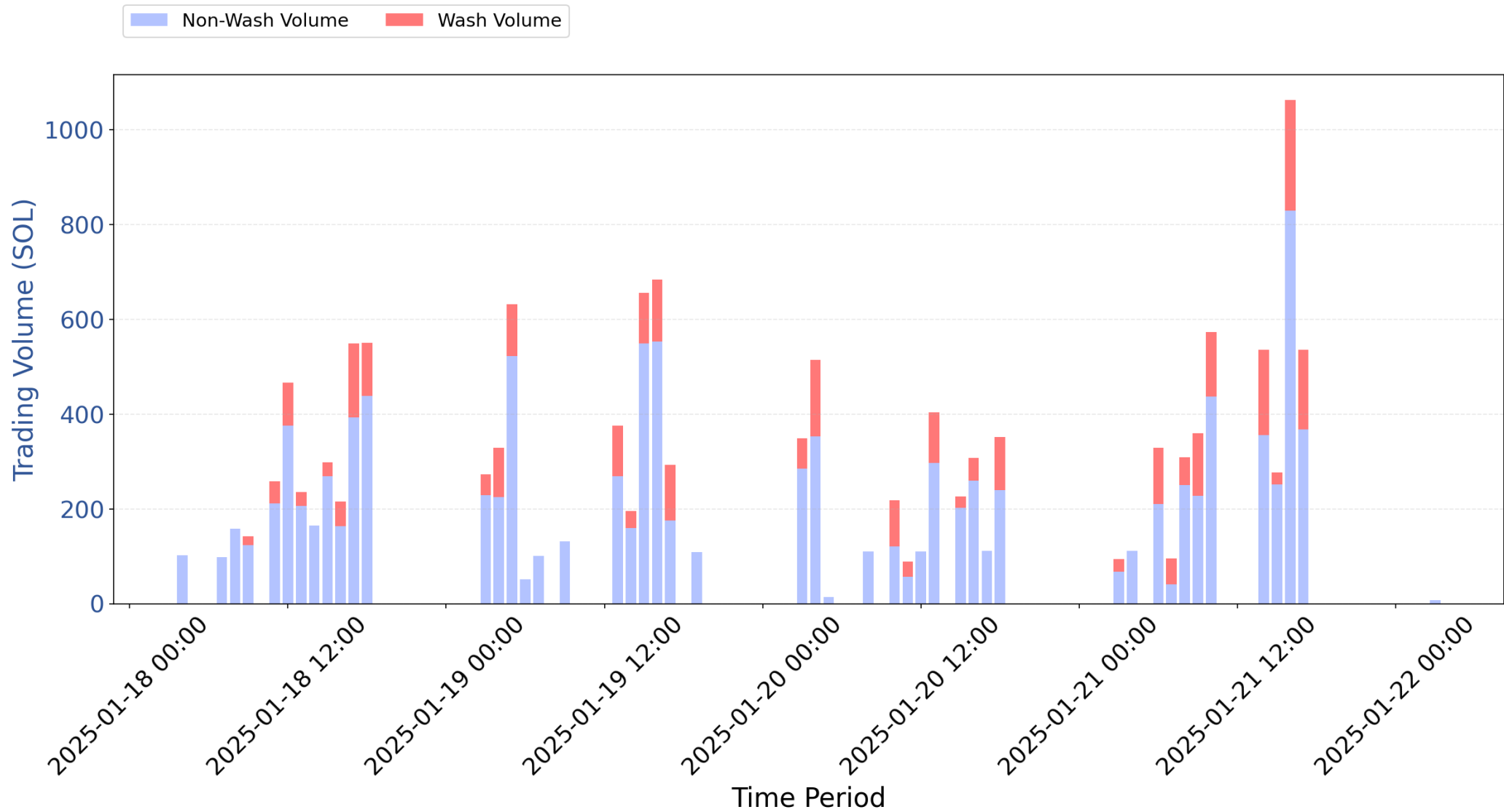


Wash Trading of Token Over Time — HC-Token (Very Strict)



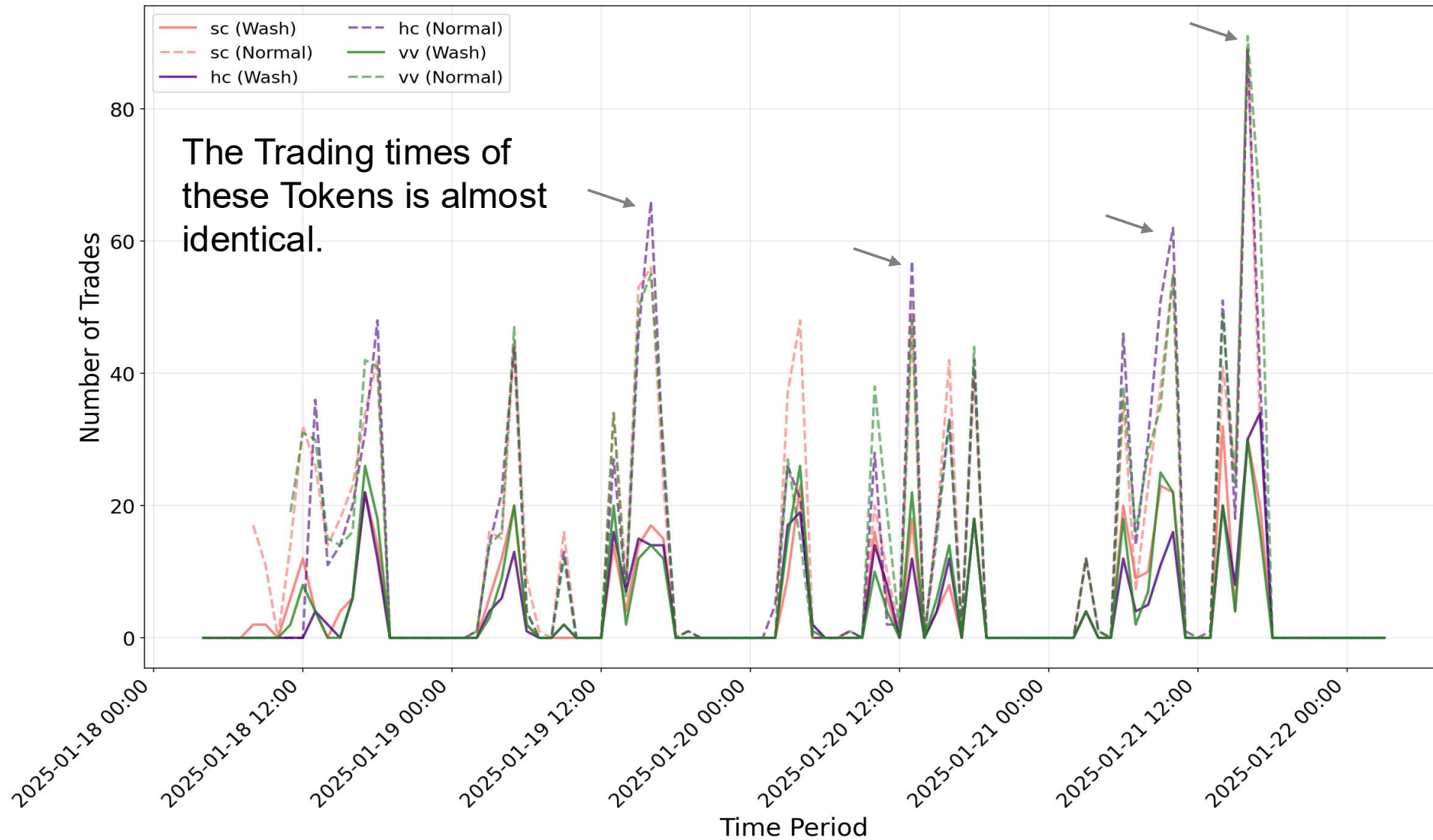
Results

Wash Trading of Token Over Time — SC-Token (Very Strict)

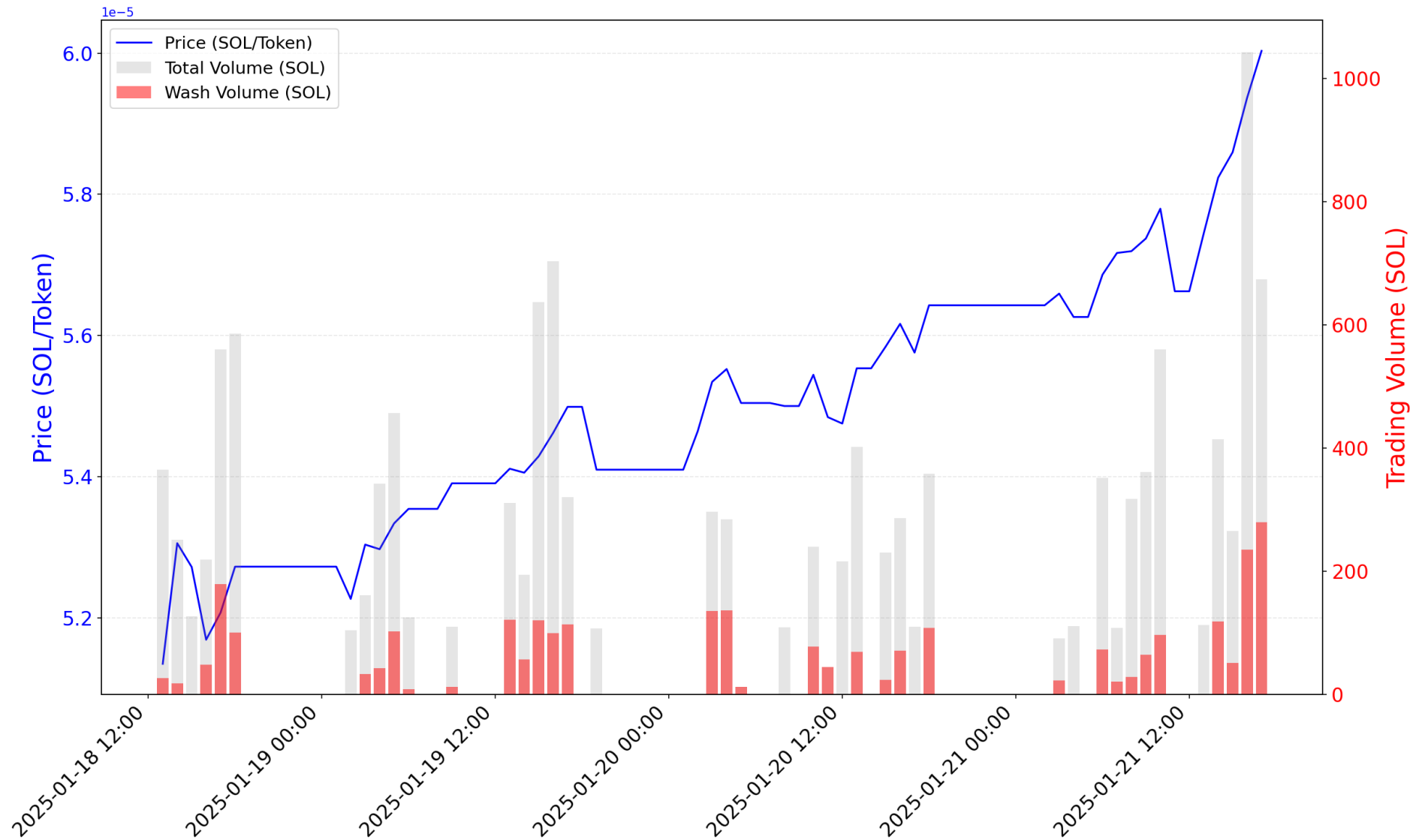


Results

Wash Trading of Token Over Time — SC, HC, VV-Token (Very Strict)

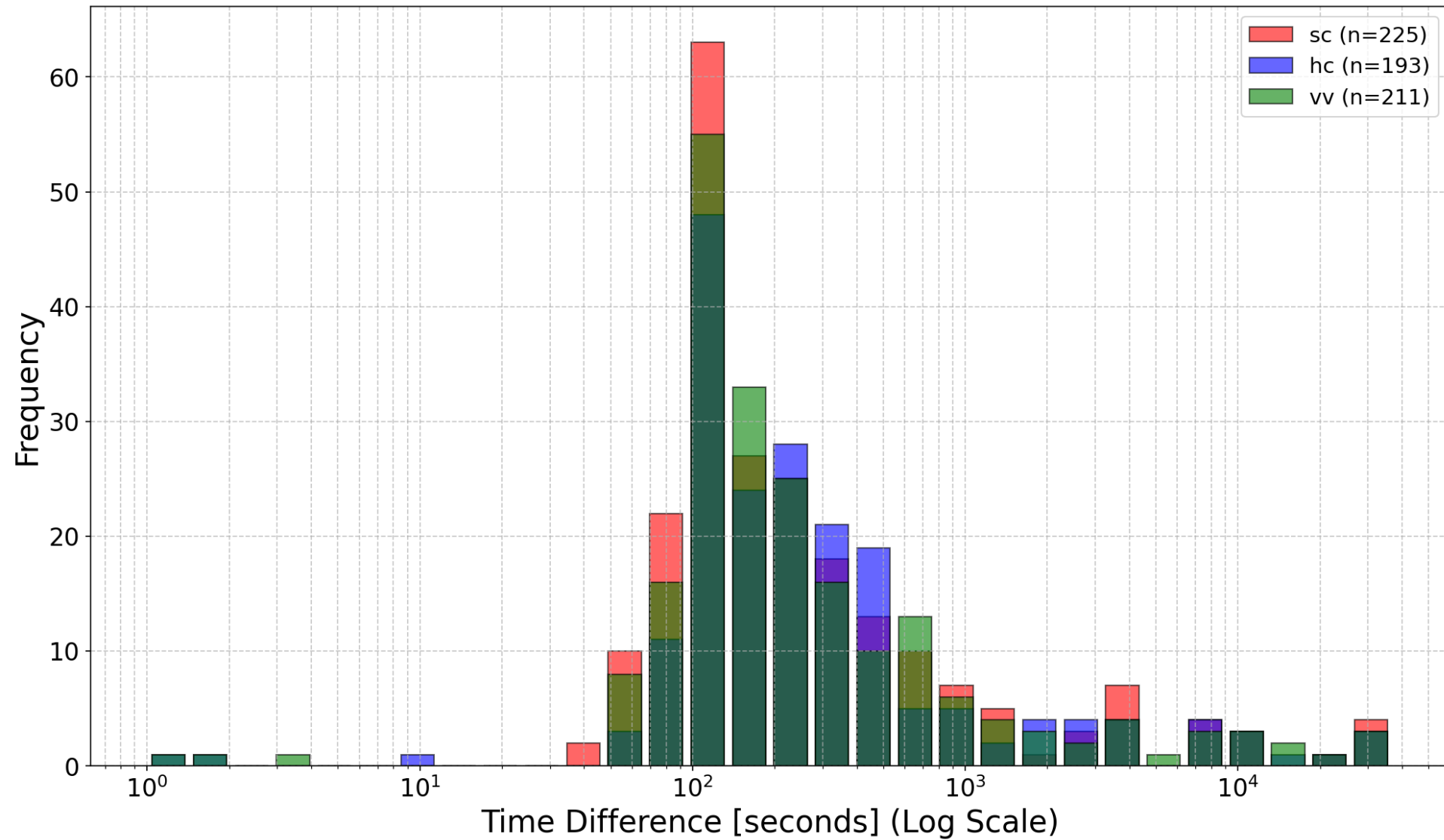


Price Development of Token's Lifetime — HC (Very Strict)



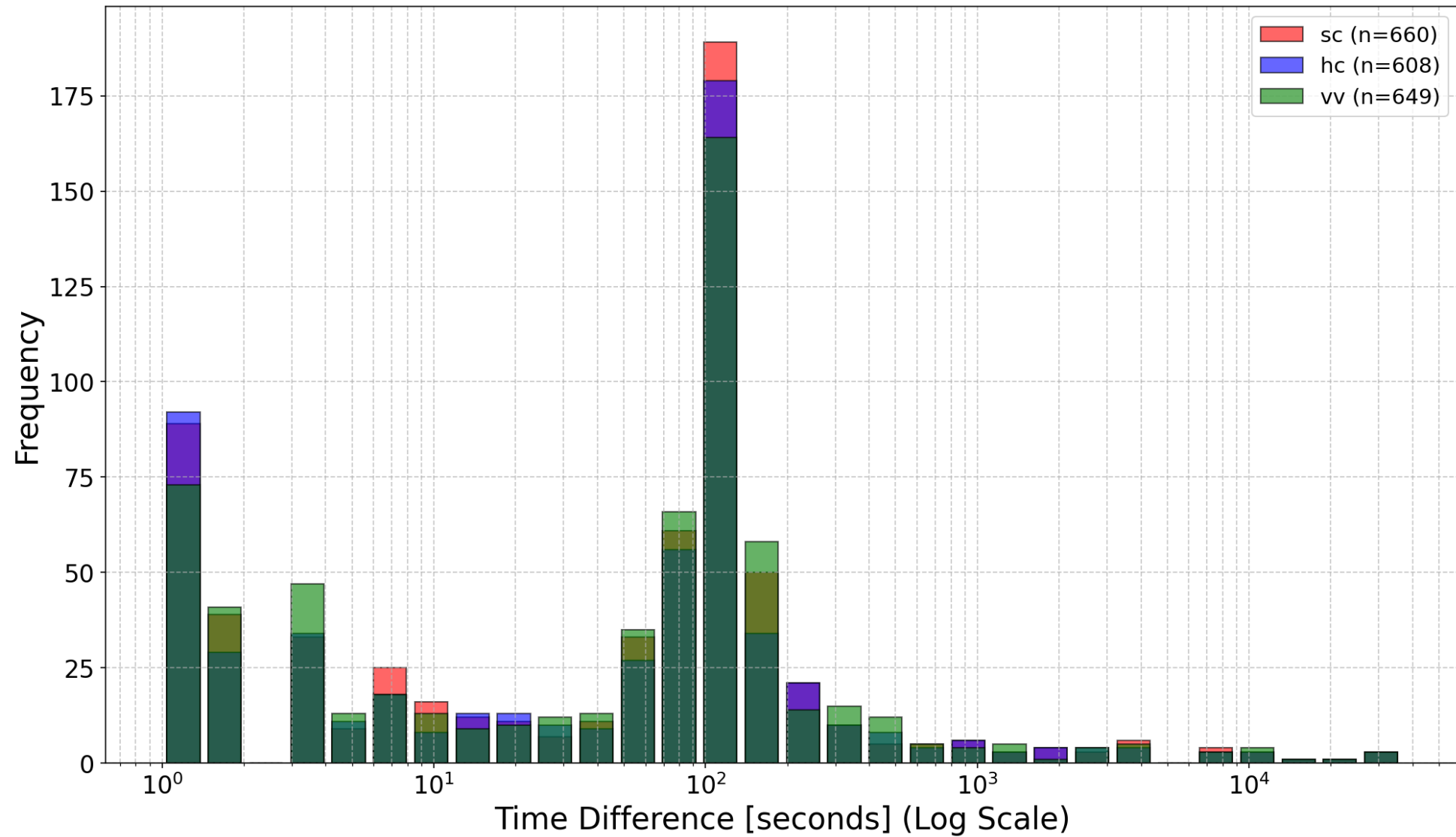
Results

SC, HC and VV-Token Wash Trading Pattern (Very Strict)

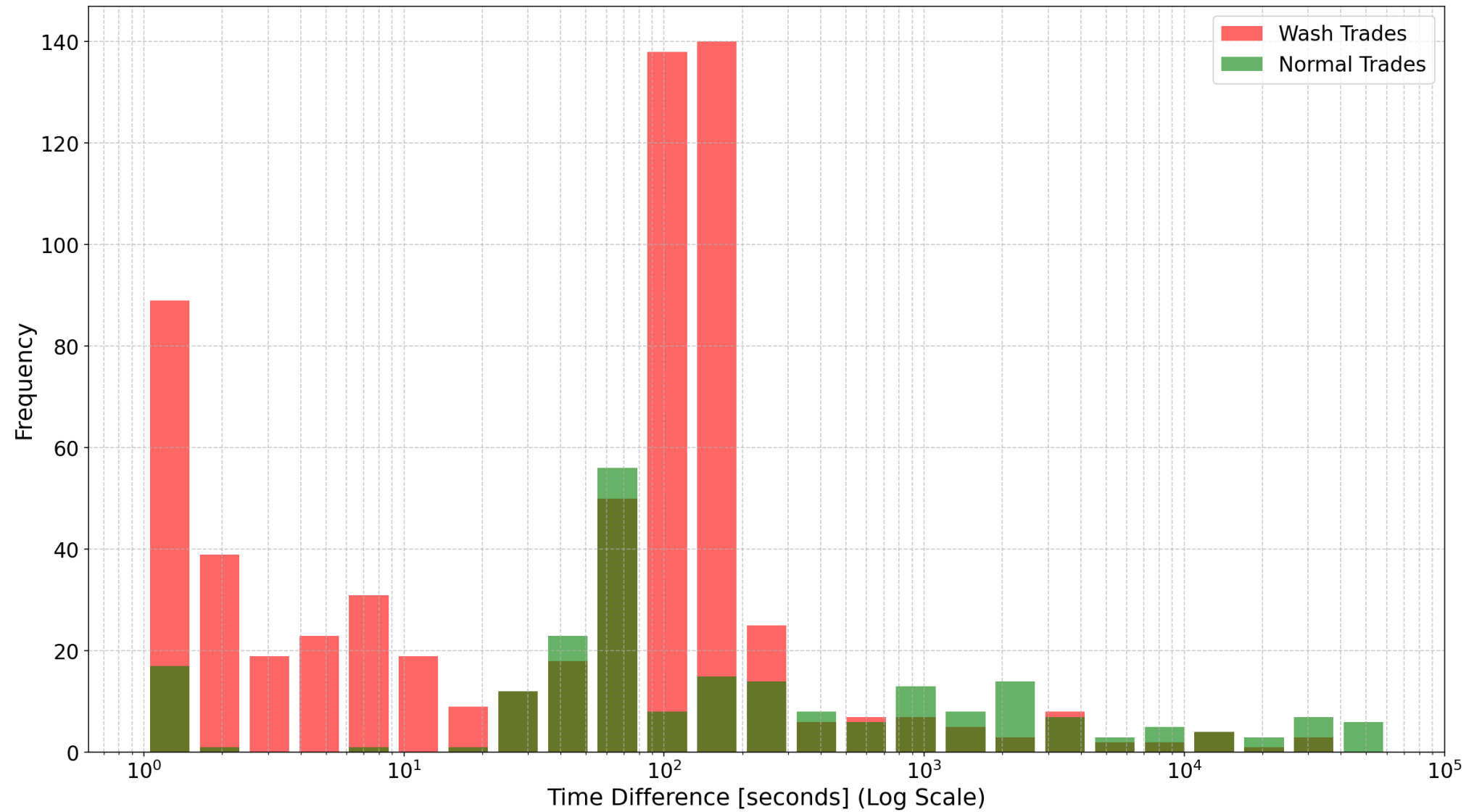


Results

SC, HC and VV-Token Wash Trading Pattern (Strict)

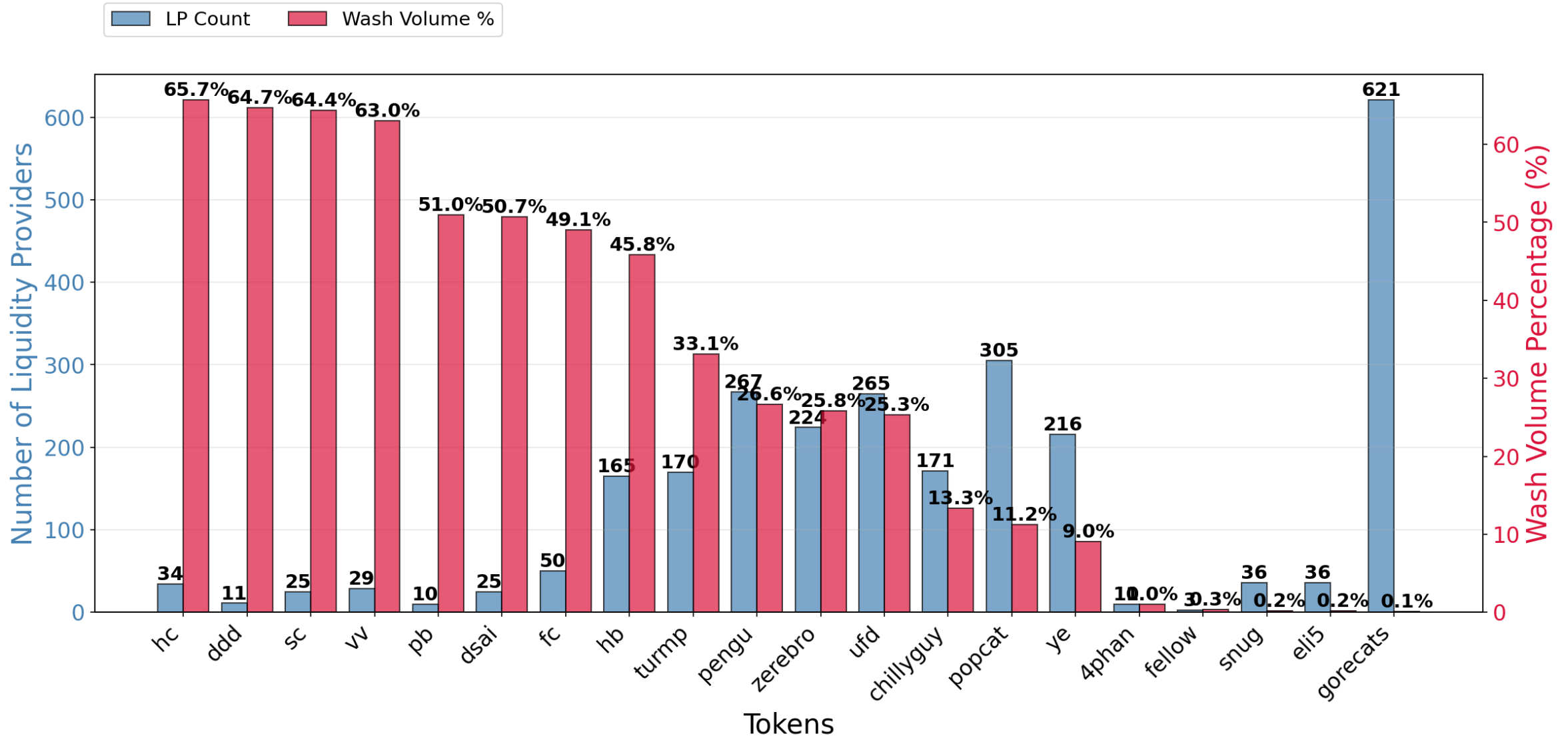


SC Wash vs. Normal Trading Pattern (Strict)



Results

Liquidity Provider Analysis (strict)



1. Background and Motivation
2. Problem Statement
3. Research Questions
4. Methodology
5. Results
6. Conclusion

Findings

- There is wash trading on Solana.
- Smaller tokens show the potential for more wash trading.
- Trade timing behavior is the best possibility to identify and validate wash trading behavior.
- There is a tendency between the number of Liquidity Providers and the percentage of wash trading of a token.

Future Work

- Implement a net amount matching algorithm to capture more complex or indirect forms of manipulation, especially cycles involving partial fills.
- Address subtler relationships between wallet addresses, such as shared funding sources or programmatic similarities, beyond explicit interactions.
- Analyze liquidity provider dynamics by accounting for temporal variation and identifying which entities provide liquidity at the time wash trading is observed.
- Large Scale analysis.



Daniel Wiedenmann

daniel.wiedenmann@tum.de

Technical University of Munich (TUM)
TUM School of CIT
Department of Computer Science (CS)
Chair of Software Engineering for Business
Information Systems (sebis)

Boltzmannstraße 3
85748 Garching bei München

+49.89.289.17132
matthes@in.tum.de
www.matthes.in.tum.de



Comparison DEX Volume:

<https://defillama.com/dexs/chains>

Solana NFT washtrade:

<https://dune.com/tianjinfan/solana-nft-wash-trading>

AMM Graph:

https://uniswapv3book.com/milestone_0/constant-function-market-maker.html

Meteora logo:

<https://www.meteora.ag>

Orca logo:

<https://www.orca.so/>

Raydium logo:

<https://raydium.io/swap/?inputMint=sol&outputMint=4k3Dyjzvzp8eMZWUXbBCjEvwSkkk59S5iCNLY3QrkX6R>

Newly launched SPL Token:

<https://solscan.io/analytics#tokens>

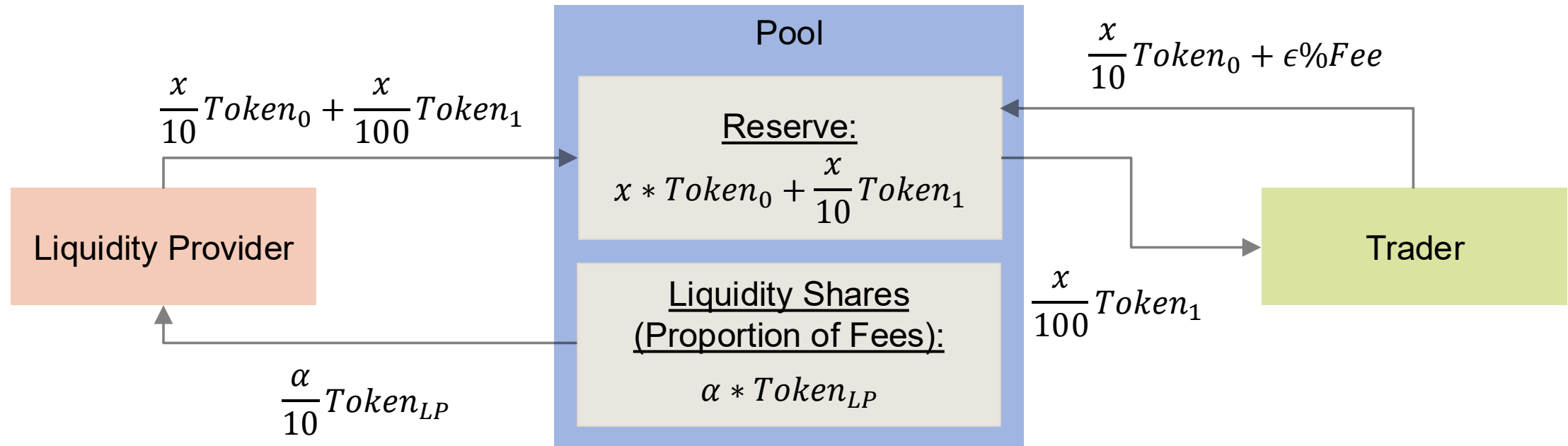
Ethereum vs. Solana:

<https://cryptomus.com/blog/ethereum-vs-solana-a-complete-comparison>

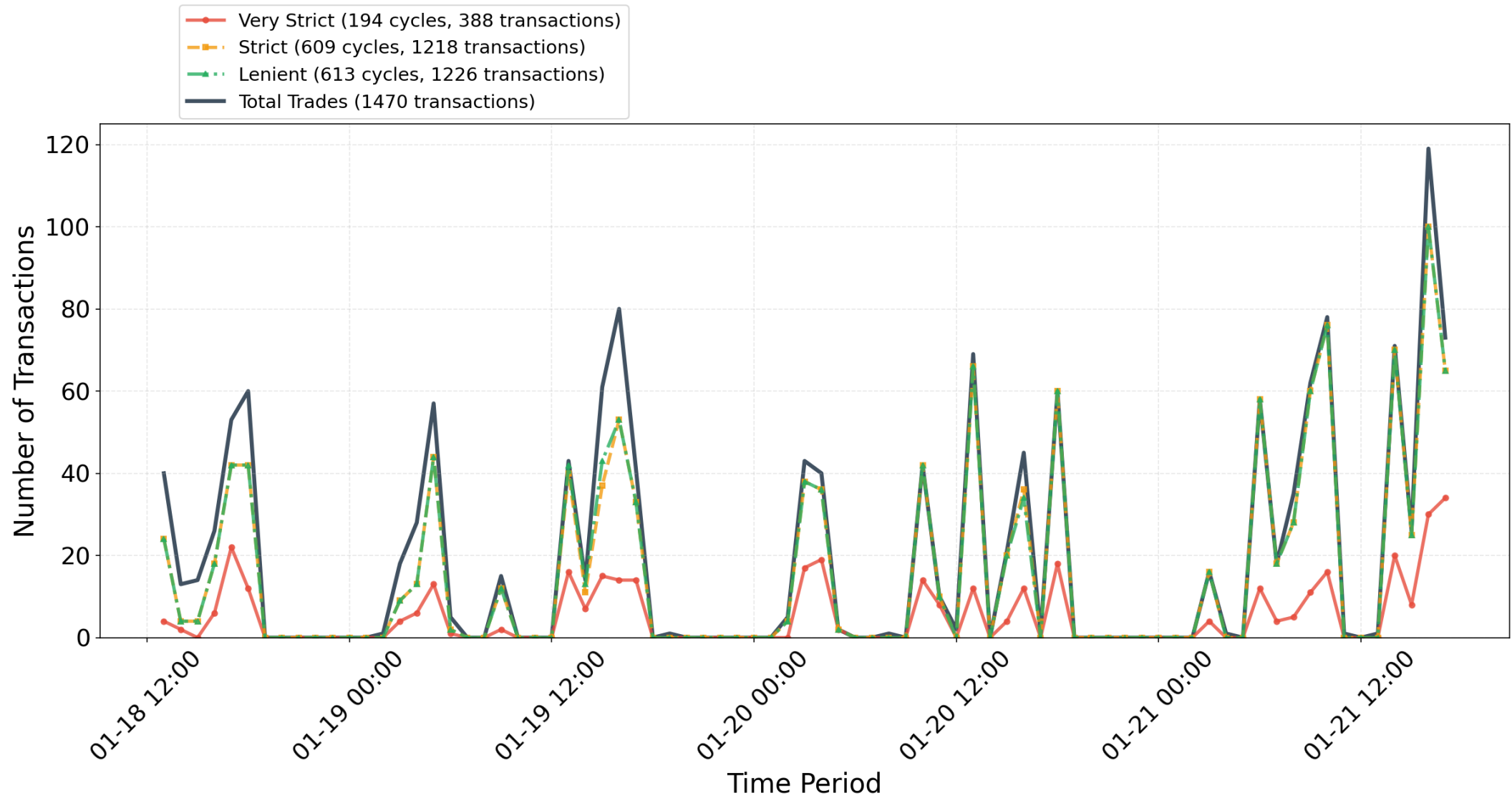
Backlog

Decentralized Exchanges

Role of Liquidity Providers in Automated Market Maker

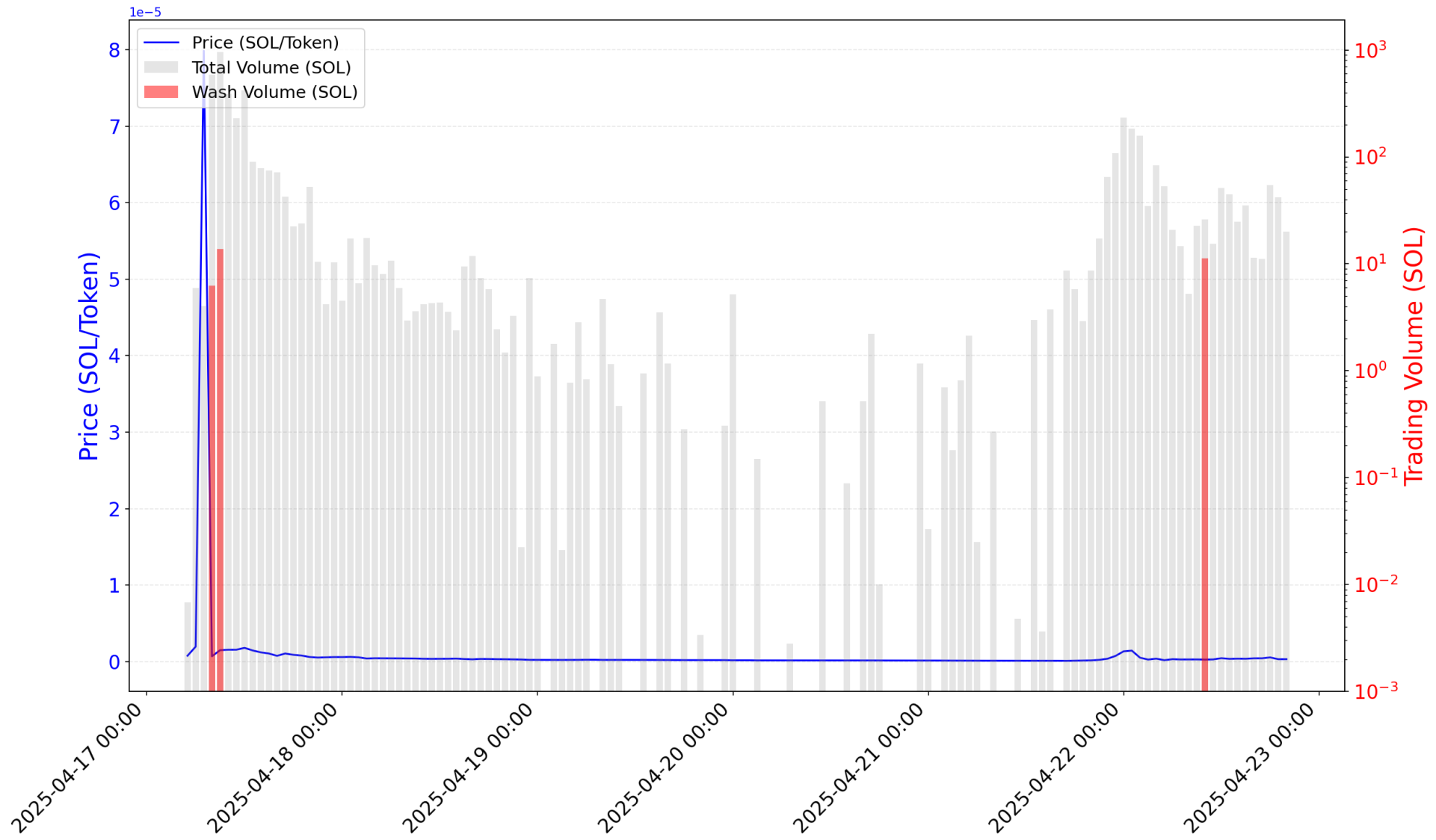


Wash Trading of Token Over Time — HC Comparison of Different Strictness



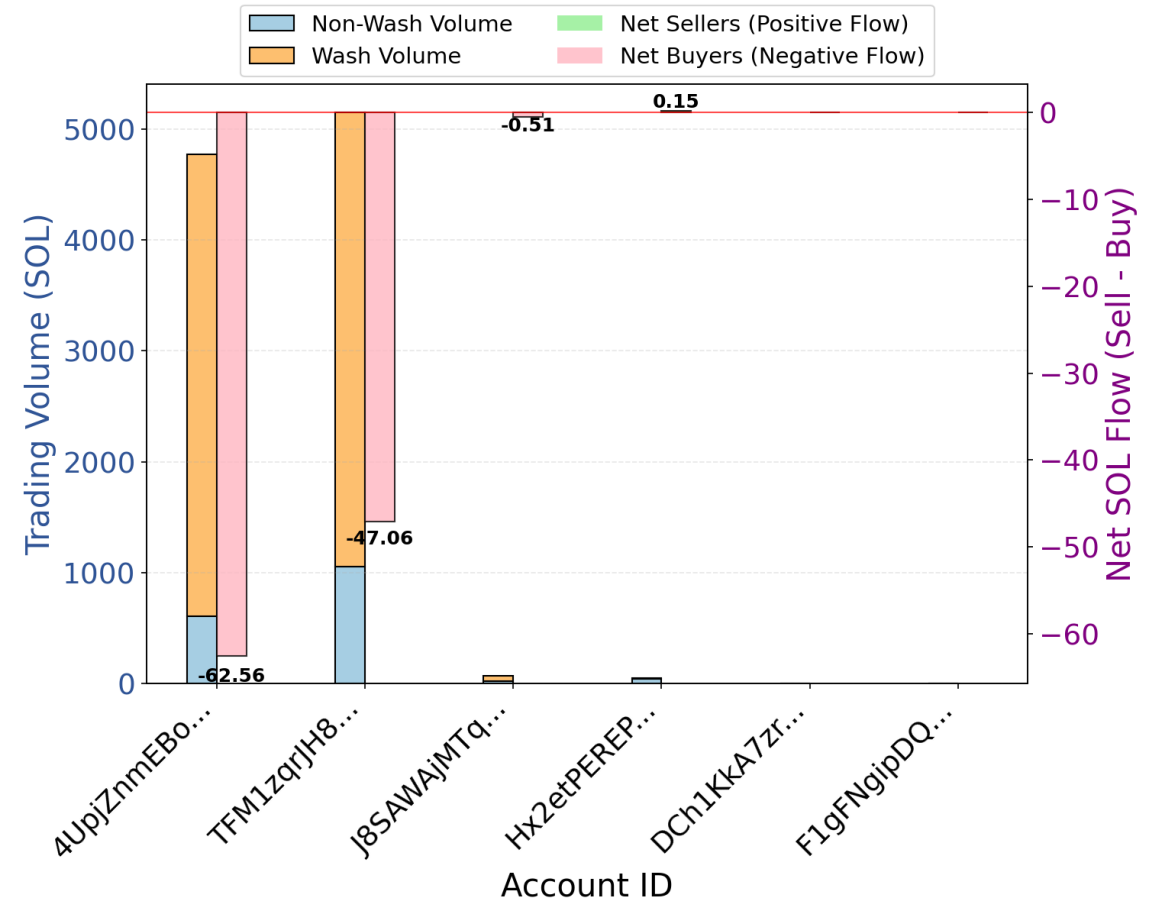
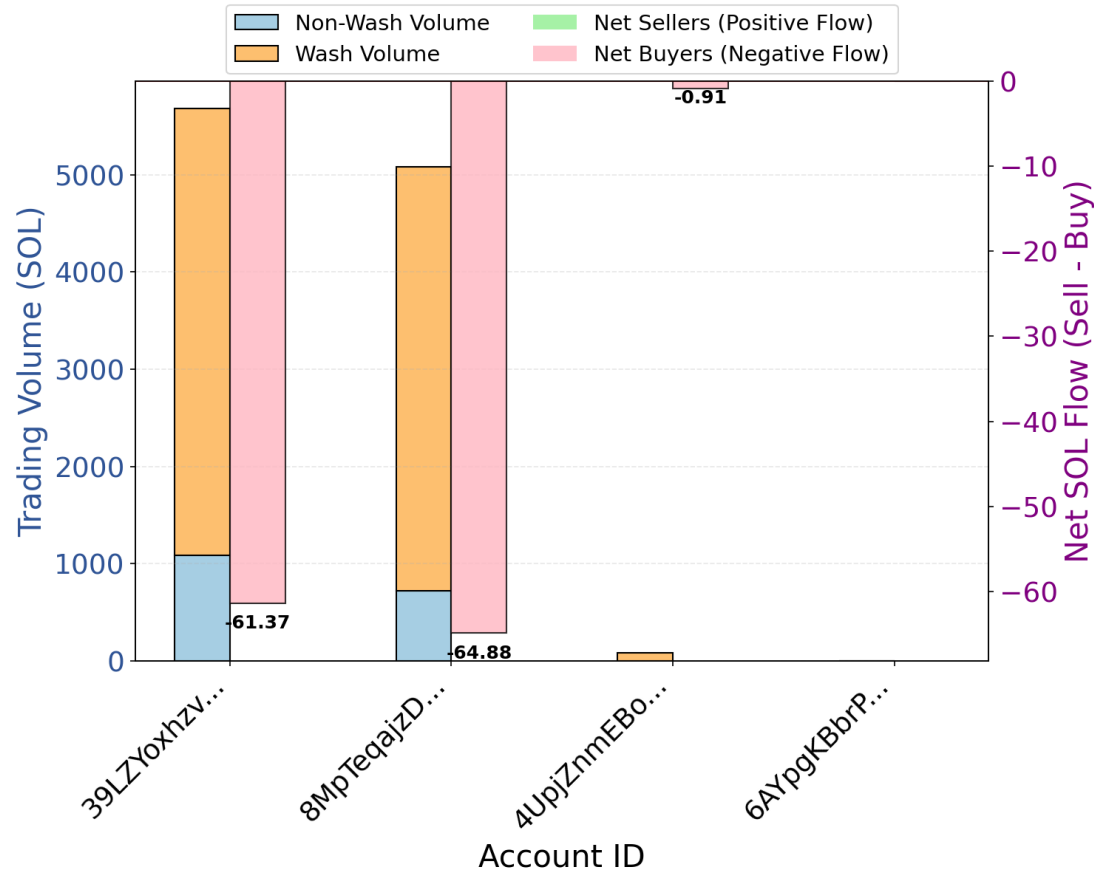
Results

Price Development of Token's Lifetime — 4phan (Very Strict)



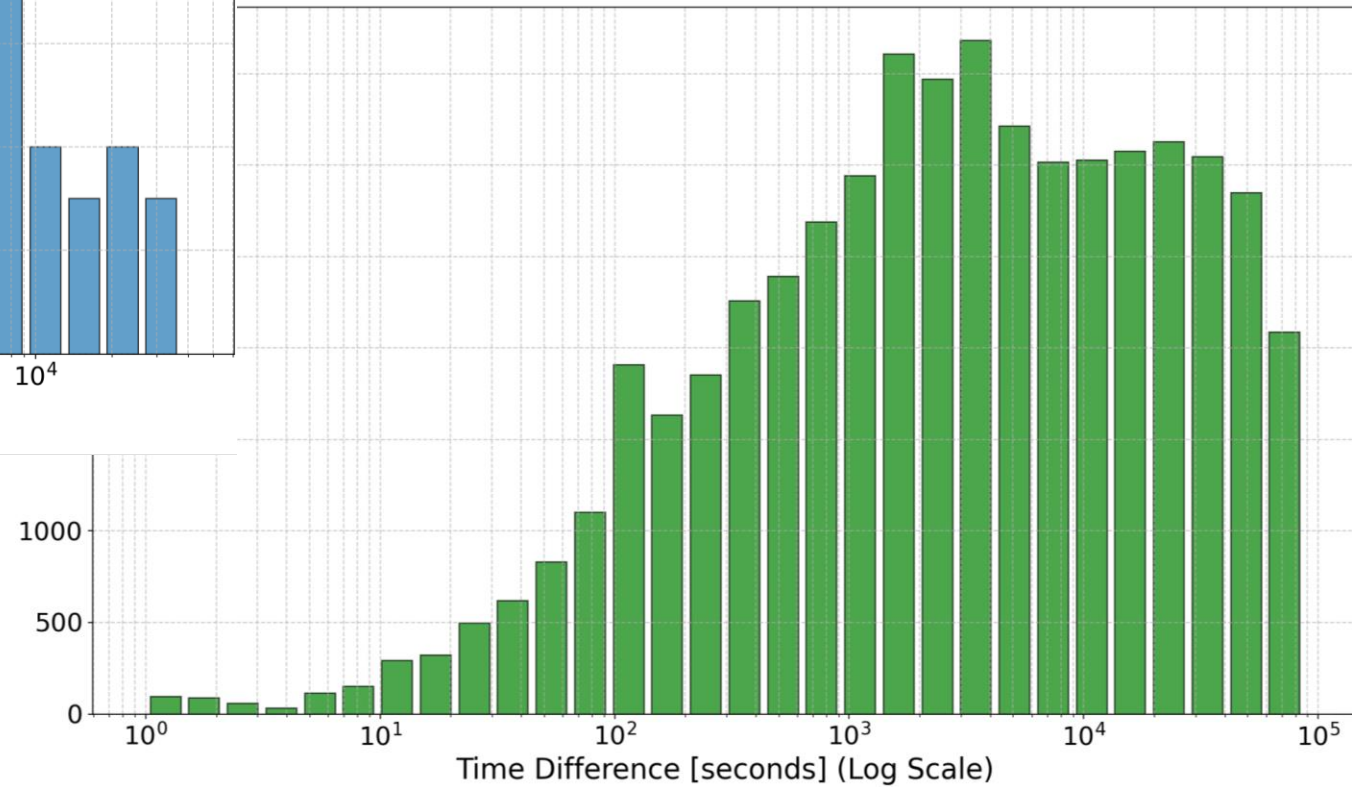
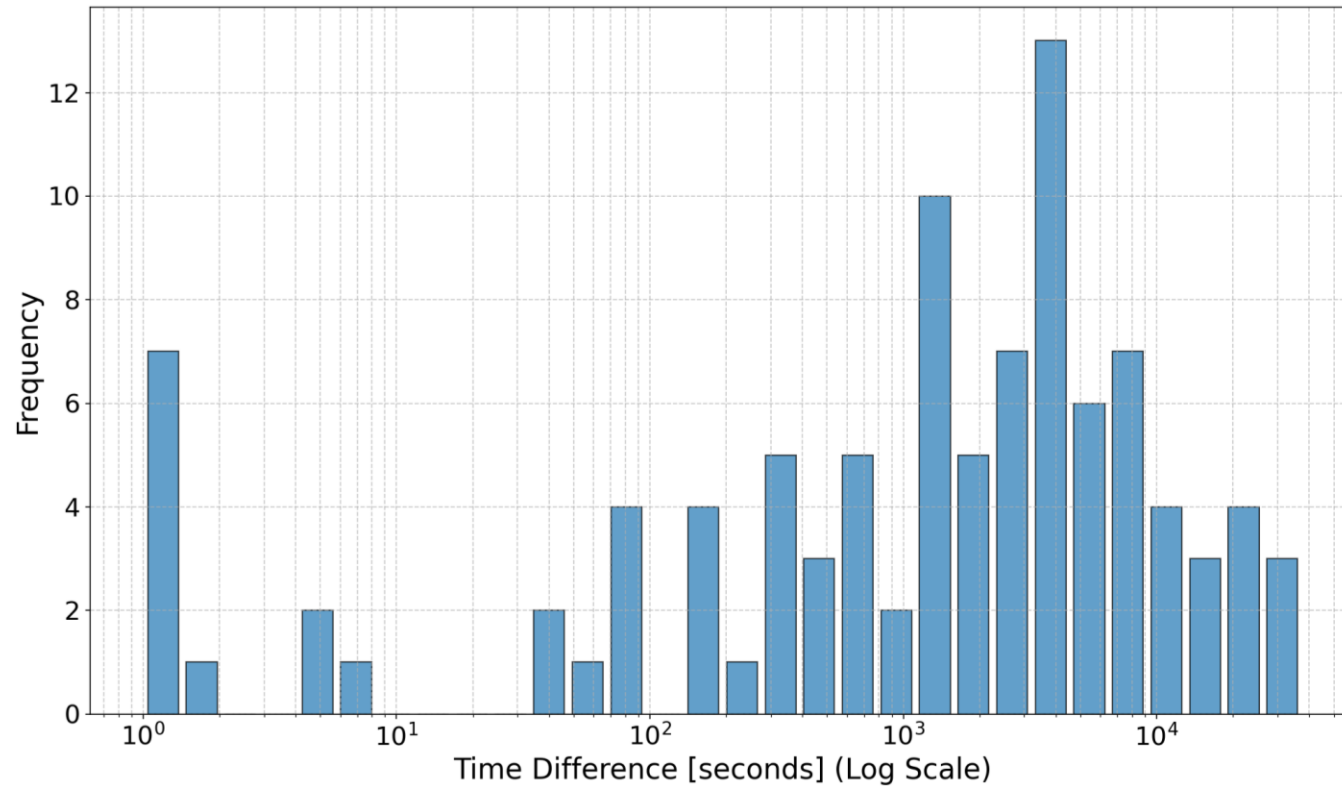
Results

Wash Trading Accounts



Results

Gorecats Wash vs. Normal Trading Pattern (Strict)



Algorithm 4 Outer Loop for Wash Trade Cycle Detection

Require: Directed graph $G = (V, E)$ of transactions; $\Delta window$ (sequence window), $\Delta amount$ (transaction amount tolerance), max_depth , $N_{intermediaries}$ (exchange accounts)

Ensure: List of detected wash-trade cycles

```
1:  $cycles\_out \leftarrow []$ 
2: for each component  $C$  in SCCs do
3:   if IS_SINGLE_NODE( $C$ ) then
4:     continue
5:   end if
6:    $G_C \leftarrow G$  induced by  $C$ 
7:    $start\_nodes \leftarrow$  REMOVE_INTERMEDIARIES( $N_{intermediaries}$ )
8:   for each  $s$  in  $start\_nodes$  do
9:      $cycles \leftarrow$  FIND_CYCLES_WITH_FILTERING( $G_C, s, \Delta window, \Delta amount, max\_depth$ )
10:     $cycles\_out \leftarrow cycles\_out + [cycles]$ 
11:  end for
12: end for
13: return  $cycles\_out$ 
```

Simplified algorithm for cycle detection 1/2

Algorithm 7 *find_cycles_with_filtering*($G, start, \Delta window, \Delta amount, max_depth$)

Require: Graph G , start node $start$, time window $\Delta window$, amount threshold $\Delta amount$, maximum depth max_depth

Ensure: List of valid cycles

```
1: Initialize:  $cycles \leftarrow [], used \leftarrow \{\}$ 
2: function DFS( $node, path$ )
3:   if  $|path| > max\_depth$  then
4:     return
5:   end if
6:   for each outgoing  $edge$  from  $node$  in  $G$  do
7:     if IS_SELF_LOOP( $edge, path$ ) then
8:       continue
9:     end if
10:     $new\_path \leftarrow path + [edge]$ 
11:    if not WINDOW_CHECK( $new\_path, \Delta window$ ) then
12:      continue
13:    end if
```

Simplified algorithm for cycle detection 2/2

```
14:     if not AMOUNT_CHECK(new_path,  $\Delta$ amount) then
15:         continue
16:     end if
17:     if CYCLE_FOUND(new_path, start) and not DUPLICATE(new_path, used) then
18:         ADD_TO_CYCLES(new_path, cycles, used)
19:     else
20:         DFS(edge.target, new_path)
21:     end if
22: end for
23: end function
24: DFS(start, [start])
25: return cycles
```
