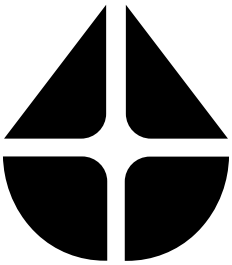
Multi Flow: Inter-block MEV Manipulation Resistant Pump For Current Values



Ben Weintraub
ben@manifestcrypto.org

Brendan Sanderson
brendan@manifestcrypto.org

basin.exchange

Published: August 23, 2023
Modified: August 23, 2023
Whitepaper Version: 1.0.0
Code Version: 1.0.0[1]

"So we sailed on through the narrow straits, crying aloud for fear of Scylla on the one hand while divine Charybdis sucked the sea in terribly on the other."

- Homer, The Odyssey[2]

Abstract

Oracles are a core piece of the decentralized finance tech stack. Non-network-native oracles that require additional trust assumptions beyond the integrity of the network are the only current option for EVM-native protocols in a post-Merge[3] environment because current network-native oracles are not resistant to inter-block maximum extractable value (MEV) manipulation. We propose an inter-block MEV manipulation resistant network-native oracle for arbitrary current data in an EVM for both instantaneous and time-weighted average (TWA) values.

1 github.com/BeanstalkFarms/Basin/blob/master/src/pumps/MultiFlowPump.sol
2 poetryintranslation.com/PITBR/Greek/Odyssey12.php
3 ethereum.org/en/roadmap/merge

Table of Contents

1 Introduction

Oracles are a principle component of complex network-native financial activity: network-native contracts often require oracles to report external data to process transactions. Oracle data can be bifurcated into data that is native to the network of the contracts using the oracle data and data that is not native to the network. Whereas data that is not native to the network must be reported to the network through some external oracle system that requires additional trust assumptions beyond the integrity of the network itself, data that is native to the network can be reported through a network-native oracle that does not. However, there are no current Ethereum-native oracles for Ethereum-native data that offer manipulation resistance in a post-Merge environment.

Prior to the Merge the potential for "risk-free" manipulation resistance was limited to intra-block (*i.e.*, within a single block) manipulation. However, the Merge created the potential for "risk-free" inter-block (*i.e.*, across multiple blocks) manipulation because block proposers are known at the beginning of each epoch. When the same proposer (or a coordinated set of proposers) knows they get to propose multiple consecutive blocks, they can execute arbitrary activity acceptable to the network (*e.g.*, buy as much of an asset as possible) without risk of having any transactions processed that cost them anything (*e.g.*, selling the asset at its now elevated price) until the first block proposed by others. In practice, there is no way to prevent inter-block manipulation from inter-block MEV. Therefore, network-native oracles must be resistant to inter-block manipulation as much as possible. However, all oracles for network-native data that were implemented prior to the Merge lack inter-block manipulation resistance. The occurrence of consecutive blocks being proposed by the same proposer is high enough[4] such that current network-native oracles are unusable for any protocol that requires manipulation resistant oracles.

# of Multi-block Occurences per Year of Ethereum

% Ownership	2 Blocks	3 Blocks	4 Blocks	5 Blocks	7 Blocks	8 Blocks	9 Blocks	10 Blocks	11 Blocks	12 Blocks
0.100000	2.480000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1.000000	254.940000	2.300000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
5.000000	5863.580000	293.660000	15.120000	0.460000	0.020000	0.000000	0.000000	0.000000	0.000000	0.000000
10.000000	20419.580000	2200.600000	217.220000	21.700000	0.300000	0.020000	0.000000	0.000000	0.000000	0.000000
15.000000	38212.320000	6885.560000	1017.480000	148.480000	2.940000	0.460000	0.040000	0.000000	0.000000	0.000000
20.000000	54447.560000	14727.420000	3035.140000	596.860000	22.020000	4.140000	0.700000	0.100000	0.000000	0.020000
25.000000	66664.040000	25405.800000	6837.320000	1690.640000	99.680000	22.740000	5.200000	1.060000	0.420000	0.140000
33.000000	77420.140000	45260.480000	17668.260000	5980.520000	624.700000	199.700000	63.740000	18.960000	6.540000	1.560000

Figure 1: Inter-Block MEV Occurrences Per Year By Validator Ownership Percent

Basin is a composable decentralized exchange (DEX) architecture native to the EVM.[5] Pumps are the generalized oracle component of Basin. The ability to compose together arbitrary exchange functions and oracles with Basin allows for the combination of existing exchange functions that remain popular for network-native exchanges with new network-native oracles that are inter-block MEV manipulation resistant.

Oracle data can be further classified as current or historical data, and instantaneous or TWA. Whereas there is inherently no network-native oracle for non-network-native data, there are potential network-native oracles for current and historical data of instantaneous and TWA values. Multi Flow supports current instantaneous and TWA values to allow for any network-native protocol to access manipulation resistant current data without additional trust assumptions beyond the integrity of the network and the Multi Flow implementation.

[4] alrevuelta.github.io/posts/ethereum-mev-multiblock

[5] basin.exchange/basin.pdf

2 Previous Work

Multi Flow is the next step in the evolution of EVM-native oracles for current values.

A robust, trustless computer network that supports composability and fungible token standards (*e.g.*, Ethereum) is necessary to host a DEX. A DEX that supports (1) the composition of other components of an exchange with arbitrary oracles (*i.e.*, Basin) and (2) an interface for arbitrary oracles (*i.e.*, *Pumps*) is necessary to support Multi Flow.

Uniswap V2[6] implemented a TWA network-native oracle for assets being traded in a Uniswap V2 liquidity pool by saving a cumulative sum of the price of the pool at the end of each block. Using the end of block price prevents intra-block manipulation but does not provide any inter-block manipulation resistance. The Uniswap V2 oracle lacks support for an inter or intra-block manipulation resistant instantaneous price.

Uniswap V3[7] innovated further on manipulation resistant network-native oracles for TWA values by moving from a simple moving average (SMA) of an arithmetic mean,[8] which weights outliers equally to all other data points, to an SMA of a geometric mean, which weights certain outliers significantly less. In addition to price, the oracle also stores data related to the liquidity in the pool, such that a time and liquidity-weighted average price can be measured. However, the Uniswap V3 oracle also lacks inter-block manipulation resistance for TWA values and supports neither an inter nor intra-block manipulation resistant instantaneous price.

Curve[9] implemented an intra-block manipulation resistant instantaneous price oracle using the price at the end of the last block. The Curve oracle lacks support for an inter-block manipulation resistant instantaneous price.

3 Multi Flow

Multi Flow uses a two-step approach to create a network-native inter-block MEV manipulation resistant oracle from manipulable data. First, Multi Flow "cleans" the data by calculating inter-block MEV manipulation resistant last values. Then, using the manipulation resistant last values, Multi Flow calculates current instantaneous and TWA inter-block MEV manipulation resistant values.

In practice, Multi Flow stores 3 different types of values (x) at the last updated timestamp (l), such that $l, x \in \mathbb{Z}^+$:

- Inter-block MEV manipulation resistant last values $[x_{0,l}^{\text{LAST}}, \cdots, x_{n,l}^{\text{LAST}}]$;
- Inter-block MEV manipulation resistant Geometric EMA values $[x_{0,l}^{\text{EMA}}, \cdots, x_{n,l}^{\text{EMA}}]$; and
- Inter-block MEV manipulation resistant Cumulative Geometric SMA values $[x_{0,l}^{\text{SMA}}, \cdots, x_{n,l}^{\text{SMA}}]$.

The second log of each value is stored. To properly read each value from Multi Flow, it must be converted back to its original value (y), such that $y \in \mathbb{Z}^+$. In the case of the TWA inter-block MEV manipulation resistant value, two values of x (*i.e.*, from the beginning and end of the period over which the TWA is being measured (t_0 and t_1, respectively) such that $t_0,\ t_1 \in \mathbb{Z}^+$) must be used. Therefore, Multi Flow supports 3 different types of values:

6 uniswap.org/whitepaper.pdf
7 uniswap.org/whitepaper-v3.pdf
8 uniswap.org/whitepaper.pdf
9 etherscan.io/token/0xc9c32cd16bf7efb85ff14e0c8603cc90f6f2ee49

- Inter-block MEV manipulation resistant last values $[y_{0,l}^{\text{LAST}}, \cdots, y_{n,l}^{\text{LAST}}]$;
- Inter-block MEV manipulation resistant Geometric EMA values $[y_{0,l}^{\text{EMA}}, \cdots, y_{n,l}^{\text{EMA}}]$; and
- Inter-block MEV manipulation resistant TWA Geometric SMA values $[y_{0,t_0,t_1}^{\text{SMA}}, \cdots, y_{n,t_0,t_1}^{\text{SMA}}]$.

Multi Flow supports reading values at both l and the current timestamp t, such that $l \leq t$, $t \in \mathbb{Z}^+$.

3.1 Inter-Block MEV Manipulation Resistant Last Values

In a post-Merge inter-block MEV susceptible environment there is no way for network-native oracles to know whether the data at the beginning or end of a particular transaction or block is being manipulated or not. In practice this means that oracles must assume the potential for manipulation of every value. Therefore, the fundamental question for manipulation resistant oracles in an inter-block MEV environment is around the treatment of statistical outliers which are certain to exist with no way for the oracle to know if it is the result of honest network activity or manipulation.

Multi Flow calculates the inter-block MEV manipulation resistant last values by using a cap on the maximum percent change of the values acceptable per block. The cap limits the extent of manipulation over a given number of blocks and in doing so "cleans" the data for postprocessing into current instantaneous and TWA oracle values.

For a given maximum increase permitted of a value per block (γ_+), maximum decrease permitted of a value per block (γ_-), block time of the given EVM in seconds (β), such that $\gamma_+,\ \gamma_-,\ \beta, \in \mathbb{Z}^+$, and the current values ($[x_{0,t}, \cdots, x_{n,t}]$), Multi Flow updates $[x_{0,l}^{\text{LAST}}, \cdots, x_{n,l}^{\text{LAST}}]$ at timestamp t as:

$$x_{i,t}^{\text{LAST}} = \begin{cases} \min\left(x_{i,t}, x_{i,l}^{\text{LAST}}(1+\gamma_+)^{\frac{t-l}{\beta}}\right) & x_{i,t} > x_{i,l}^{\text{LAST}} \\ \max\left(x_{i,t}, x_{i,l}^{\text{LAST}}(1-\gamma_-)^{\frac{t-l}{\beta}}\right) & \text{otherwise} \end{cases}$$

Because exponentiation is expensive in the EVM, in practice $[x_{0,l}^{\text{LAST}}, \cdots, x_{n,l}^{\text{LAST}}]$ is updated at t as:

$$x_{i,t}^{\text{LAST}} = \begin{cases} \min\left(log_2(x_{i,t}), x_{i,l}^{\text{LAST}} + log_2(1+\gamma_+)\frac{t-l}{\beta}\right) & log_2(x_{i,t}) > x_{i,l}^{\text{LAST}} \\ \max\left(log_2(x_{i,t}), x_{i,l}^{\text{LAST}} + log_2(1-\gamma_-)\frac{t-l}{\beta}\right) & \text{otherwise} \end{cases}$$

Therefore, the multi-block MEV manipulation resistant last values can be read as:

$$y_{i,l}^{\text{LAST}} = 2^{x_{i,l}^{\text{LAST}}}$$

3.2 Inter-Block MEV Manipulation Resistant Instantaneous Values

Multi Flow uses an exponential moving average (EMA) of the inter-block MEV manipulation resistant last values to calculate inter-block MEV manipulation resistant current instantaneous values. EMAs are useful for aggregating a time-series of data into a single value and are preferable to SMAs for instantaneous value because (1) they weight recent data more than older data and (2) instances where the direction in the deviations of the new last value and EMA value from the old ones are different are much more limited in frequency and scale (*i.e.*, their changes are directionally the same more often). An EMA of the inter-block MEV manipulation resistant last values is preferable to the inter-block MEV manipulation resistant last values for reading an instantaneous value because it requires manipulation over extended periods of time to significantly manipulate the value.

For a given exponential decay parameter (α), such that $\alpha \in (0, 1)$, $[x_{0,l}^{\text{EMA}}, \cdots, x_{n,l}^{\text{EMA}}]$ and $[x_{0,t}^{\text{LAST}}, \cdots, x_{n,t}^{\text{LAST}}]$, Multi Flow updates $[x_{0,t}^{\text{EMA}}, \cdots, x_{n,t}^{\text{EMA}}]$ at timestamp t as:

$$x_{i,t}^{\text{EMA}} = \alpha^{t-l} x_{i,l}^{\text{EMA}} + (1 - \alpha^{t-l}) x_{i,t}^{\text{LAST}}$$

Therefore, the inter-block MEV manipulation resistant current instantaneous values can be read as:

$$y_{i,l}^{\text{EMA}} = 2^{x_{i,l}^{\text{EMA}}}$$

3.3 Inter-Block MEV manipulation resistant TWA Values

Multi Flow uses a cumulative sum of the inter-block MEV manipulation resistant last values to calculate inter-block MEV manipulation resistant SMA values. SMAs are useful for aggregating a time-series of data into a single value and are preferable to EMAs for TWA value because they weight recent data the same as older data.

For a given $[x_{0,l}^{\text{SMA}}, \cdots, x_{n,l}^{\text{SMA}}]$ and $[x_{0,t}^{\text{LAST}}, \cdots, x_{n,t}^{\text{LAST}}]$, Multi Flow updates $[x_{0,t}^{\text{SMA}}, \cdots, x_{n,t}^{\text{SMA}}]$ at timestamp t as:

$$x_{i,t}^{\text{SMA}} = x_{i,l}^{\text{SMA}} + (t - l) x_{i,t}^{\text{LAST}}$$

Therefore, the inter-block MEV manipulation resistant TWA values can be calculated from x_{i,t_0}^{SMA} and x_{i,t_1}^{SMA} as:

$$y_{i,t_0,t_1}^{\text{SMA}} = 2^{\frac{x_{i,t_1}^{\text{SMA}} - x_{i,l}^{\text{SMA}}}{t_1 - t_0}}$$

Because only the latest values of $[x_{0,l}^{\text{SMA}}, \cdots, x_{n,l}^{\text{SMA}}]$ are stored by Multi Flow, protocols must save values at the beginning of the period over which the TWA is being measured.

3.4 Governance

Multi Flow is non-upgradable. Therefore, Multi Flow does not support governance.

4 Risks

There are risks associated with Multi Flow. This is not an exhaustive list.

The Multi Flow code base is novel. It has not been tested in the "real world" prior to its initial deployment. The open source nature of Multi Flow means that others can take advantage of any bugs, flaws or deficiencies in it. While Multi Flow has been audited[10,11] it is no guarantee of security.

Multi Flow does not prevent manipulation of network-native values. Instead, it minimizes the effect of manipulation on oracle values. Therefore, Multi Flow is manipulation resistant, not manipulation-preventing.

5 Future Work

Manipulation resistant network-native oracles are a work in progress. The following are some potential improvements that can be incorporated into future iterations of manipulation resistant network-native oracles:

[10] basin.exchange/06-16-23-halborn-report
[11] basin.exchange/06-16-23-cyfrin-report

- Support for historical values.
- Support for additional data (*e.g.*, volatility).
- The ability to turn off when manipulation is detected.
- The ability to reset if it breaks.