# Model: Pool Centralization From MEV

## Summary

One of the main justifications given by@vbuterin and others for in-protocol PBS is this:

“A pool that is 10x bigger will have 10x more opportunities to extract MEV but it will also be able to spend much more effort on making proprietary optimizations to extract more out of each opportunity”.

This is an early version of a model to find out how much of a problem this kind of validator centralization is, and to find out whether MEV auctions are an effective solution.

It’s a work in progress and a call for peer review, not a final publication. I’m looking for input on any problems with my logic, model, assumptions, parameters etc. Thanks all!

## Model

Run/edit the model code in your browser:Ethereum Validator Centralization Model | C# Online Compiler | .NET Fiddle

Here are my results: ValidatorCentralizationResults - Google Sheets

First Thoughts:

1. Validator centralization from MEV is low in all cases, even over 50 years
2. MEV auctions (eg: MEV-Boost/PBS) make it worse
3. Toxic MEV mitigation makes it better
4. Builder/Order flow centralization (which is incentivized by MEV auctions withouttoxic MEV mitigation) makes things much worse

[

Validator Centralization Model Results

880×538 25.2 KB

](https://ethresear.ch/uploads/default/original/2X/a/aa2cdc02f0c88d1925d1cd0f0c4dc657b4ee2c0b.png)

- If a pool is significantly better at extracting than anyone else, they will earn relatively more to reinvest in staking when using mev auctions than without, which worsens validator centralization
- Builder centralization is incentivized by mev auctions, especially private order flow from users trying to avoid toxic mev, which worsens centralization further (ie: builder centralization causes validator centralization).
- I’m aware that the builder centralization modeling is inadequate, I just added it as a fixed amount of mev for now- actually there will be a feedback effect.

## Description

- Initial pool sizes are taken from current figures (see poolStartingAllocations

0.3, 0.15, 0.14, 0.09, 0.06, 0.03, 0.23)

- The mev extraction efficiency for each pool is estimated based on pool size (1, 0.9, 0.85, 0.8, 0.75, 0.75, 0.7), and doesn’t drop below 70% due to MEV extraction still being possible via GPA even on gas price only validators (poolMevEfficiency

, 1 = all possible mev extracted, 0 = no mev extracted)

- Select preset tests to run (test = 0,1,...

)

- Estimated mev per block is taken fromMEV in Eth2 by Alex Obadia
- Staking rewards per block are estimated fromBeacon Chain Validator Rewards by pintail
- Define whether you want to use mev auctions or not (useMevAuctions)

- Define how much mev has been mitigated as a percentage (mevMitigated, 1 = no mev mitigated, 0.5 = half mev mitigated). It is set to 40% in the mitigation sim to represent preventing sandwich attacks via threshold encryption of the mempool
- Iterates blocks over a 50 year period choosing block producers proportional to their stake
- Adds staking rewards and mev per block:
- No mev auctions: the producer extracts mev at their own level of efficiency
- With mev auctions: the producer gets the winning bid, the extractor wins the difference between the mev they can extract and the nearest competitor bid
- No mev auctions: the producer extracts mev at their own level of efficiency
- With mev auctions: the producer gets the winning bid, the extractor wins the difference between the mev they can extract and the nearest competitor bid
- Assumes all rewards and mev are reinvested into staking