Validator Economics: Variable min validator deposit size

EF Academic Grant ID: FY23-1030

DRAFT MODEL, CHALLENGES & POTENTIAL MITIGATIONS
(MAXEB - EIP-7521)

Sandra Johnson
Consensys Software R&D

January 29, 2024

1 Overview

This document builds on the previous document, *Milestone 1(EB): Review & analyse EIP-7521*, addressing each of the identified challenges and applying example scenarios to the expected behaviour cited in blog posts and analysis done to date.

In summary, as mentioned in the previous document, the two key criteria we need to assess for this proposal, and indeed for any proposed strategy to reduce, or cap, the validator set size, are the risks and benefits introduced by that proposal.

With respect to EIP-7521:

- **Risks**: Are risks are being introduced by this proposal? Are there adverse affects on some categories of stakers, or the Ethereum ecosystem?
- **Benefits**: Do all staker categories benefit equally from EIP-7521? Apart from the expected reduction in network load, are there other benefits to the health of the ecosystem?

We need to bear in mind that regardless of the risks and benefits, EIP-7521 relies on stakers' willingness to consolidate at least some of their validators. Therefore the messaging around the benefits of consolidation and the perception that the benefits to stakers and the ecosystem definitely outweigh any perceived risks needs to be compelling.

The impact on staking pools, their operational business models, and software changes required to take advantage of the increased maximum will need to be discussed with stakeholders effective balance (EB).

2 Challenges & Additional analysis requirements

In addition to the advantages and disadvantages highlighted in the *ethresear.ch* post, *Increase the MAX_EFFECTIVE_BALANCE – a modest proposal*, by Neuder et al [36], we propose three topics that may benefit from further in-depth analysis and working examples:

- Validator duty selection
- Rewards & penalties
- Ethereum ecosystem & protocol

Each of the three topics will be discussed in more detail.

2.1 Validator duty selection

In this section we review each of the selection processes to confirm or challenge the anticipated behaviour.

1. *Proposer selection*: demonstrate that proposer selection still works as designed with partial and full consolidation of validators.
2. *Sync committee selection*: demonstrate that sync committee selection will work as designed even when there is a large increase in MaxEB.
3. *Slot committees*: Demonstrate that not taking validator weight into account does not have any adverse downstream effects. D'Amato and Neuder state that even if one "attestation is majority adversarial", then the worst that can happen is that the attacker will execute a local reorg [15]. Explore in more detail, e.g. use an example or two, to ascertain the effect of variations in slot committee balances with the inclusion, or omission, of fully consolidated stakers.
4. *Aggregator selection*: Does the selection of more than one virtual validator of a consolidated validator change the probability distribution of sub-committees? Could the probability of fewer than 16 distinct aggregators increase?
   Virtual validators: D'Amato and Neuder's conjecture is that treating consolidated validators as several 'virtual' validators yields two main benefits [15]. Demonstrate the correctness of these statements.:
   - The total number of aggregators is unchanged, because one consolidated validator would account for only one aggregator.
   - The probability of at least one honest validator is equivalent to the current probability, because the introduction of virtual validators means that the probability of selecting an honest aggregator is the same as it is currently. The total number of honest validators prior to EIP-7251 is the same as the number of virtual and unconsolidated honest validators.

2.2 Rewards & penalties

1. *Proposer selection probability*: Demonstrate that this probability remains unchanged and will not unduly disadvantage small-scale solo stakers [36].

2. *Whistleblower reward*: Currently the proposer receives both the whistleblower reward and the proposer reward. Moreover, the whistleblower reward is proportional to the slashed validator's effective balance. How is this reward influenced by the increased maximum effective balance?

3. *Slashing penalties*: Verify the suitability of the suggested changes to slashing penalties (initial and correlation penalties), as well as those penalties remaining unchanged (attestation and inactivity leak penalties), proposed by Neuder and Monnot [37] using working examples, or other approaches.
   Moreover, based on the severity of the changed slashing penalties, what situation constitutes an attack on the chain? How do we assess whether the penalties are sufficient to deter adverse behaviour? Are the proposed changes too lenient for large stakers?

2.3 Ethereum ecosystem & protocol

It is important that there no adverse impacts on the health of the Ethereum ecosystem, and a lot of analysis has already been undertaken to ensure that it remains secure and healthy. As a consequence of EIP-7521, as with all EIPs, changes to the protocol would be required, which the initiators of the proposal have endeavoured to minimise.

1. *Worst case: Full consolidation*: D'Amato and Neuder identify full consolidation as the worst case scenario when we consider two groups of validators: honest and adversarial validators. Several reasons are given for this conclusion [15]. Using examples or other approaches we need to check the veracity of these statements, viz. that full consolidation:

   - increases the probability that one or more committees will be majority dishonest
   - increases the variance of adversarial balances
   - increases the variance of the distribution of adversarial weight over committees
   - due to the above, it follows that it is more likely that there will be "a positive deviation from the expected adversarial weight in a committee". Apparently although there may also be "negative deviations" the adversarial party is not concerned because it merely aims to control at least one committee.
   - On the other hand, full consolidation of honest validators maximises the risk of negative deviations from the expected committee weight.
   - Honest validators benefit from spreading weight evenly across the committees because it minimises the risk of negative deviations.

2. *Discouragement attacks*: Investigate potential griefing, or discouragement, attacks.

3. *Consolidating validators*: There are many challenges around exiting and activation of validators for consolidation. Therefore, as Asgaonkar points out, in-protocol consolidation needs to be in place [5].

4. *Withdrawals*: The automated withdrawal sweep will now skip over consolidated validators with an effective balance below 2, 048 ETH. Therefore, there may be reductions in time to do a full sweep as well as a reduced network load because fewer withdrawals will be initiated.
   The capability of being able to initiate partial withdrawals for a specified amount appears to be a requirement of large staking pools such as Lido.

5. *Centralisation forces*: Centralisation needs to be assessed within the context of this grant and the proposed increase in maximum effective balance, especially with respect to large stakers and staking pools.

6. *Bayesian network model*: Build a Bayesian network (BN) to capture current factors and interactions, including distributions around factors such as uptake of consolidation, current distribution of different categories of stakers, etc.
   Use the BN model to gain insight into questions such as:

   - What does the landscape look like if there is no consolidation, but some or all validators decide to compound their rewards? Is this a good or desirable outcome?
   - Create some example scenarios that include solo, partial and fully consolidated stakers and how different compositions translate to the selection of validators for specific duties and ideally the probability of dishonest committees.

In summary it is important to ensure that small scale solo-stakers are not unduly disadvantaged with the introduction of a 2,048 ETH MaxEB, in terms of proposer selection etc. which may disincentivise the commitment to running a validator. What is being compromised to enable consolidation? Are there other aspects we have not considered? What are the negatives for solo stakers who are considered to be the backbone of Ethereum? Does MaxEB swing it too much in favour of large stakers? One important advantage for solo stakers is that any rewards earned can be autocompounded and if they have insufficient ETH to activate another validator, they can instead top up their validator(s) with amounts less than 32 ETH, e.g. 5 ETH.

3 Analysis - TO DO

3.1 Validator duty selection

The selection of validators to be part of committees or to be selected as a proposer needs to be done in an unpredictable, or random, way to ensure that the selection cannot be manipulated by bad actors.

However, in a blockchain system, all nodes need to come to consensus, and the key "randomness lever" is the seed used in the calculation that would have the same outcome for all nodes. Therefore the seed needs to be unpredictable [9].

In the Consensus Layer there are a few different approaches to achieve randomness used in selection of validators for duties:

- Aggregators are selected via a verifiable random function (VRF) lottery [17].

Proposer

Endianness, the order of bytes in the binary representation of a number, is not commonly of interest, but in the case of index shuffling and proposer selection, the RANDAO, and serialising with SSZ, the endianness does matter. Initially big-endian was used, i.e. the first byte is the most-significant byte, but this has now changed mainly to little-endian, i.e. the first byte is the least-significant byte [17].

The *swap-or-not-shuffle* technique [25] is used to shuffle the validator indices in preparation for the selection of a block proposer. This is done in *compute_shuffled_index*.

The computation to determine the proposer for the next block is done in *compute_proposer_index*:

```
def compute_proposer_index(state: BeaconState, indices: Sequence[ValidatorIndex], seed:
    Bytes32) -> ValidatorIndex:
    """
    Return from ``indices`` a random index sampled by effective balance.
    """
    assert len(indices) > 0
    MAX_RANDOM_BYTE = 2**8 - 1
    i = uint64(0)
    total = uint64(len(indices))
    while True:
        candidate_index = indices[compute_shuffled_index(i % total, total, seed)]
        random_byte = hash(seed + uint_to_bytes(uint64(i // 32)))[i % 32]
        effective_balance = state.validators[candidate_index].effective_balance
        if effective_balance * MAX_RANDOM_BYTE >= MAX_EFFECTIVE_BALANCE * random_byte:
            return candidate_index
        i += 1
```

Therefore, we iterate through the shuffled indices, starting with the first entry and then check whether it passes the selection criteria. If it doesn't, then the next candidate index goes through the same checks.

The validator's effective balance is multiplied by 255 and then compared to the product of the generated *random_byte*, which we assume is uniformly randomly distributed across 0 to 255, and the *MAX_EFFECTIVE_BALANCE* which is now 2,048 ETH.

Therefore the probability of a validator being the proposer if their index was selected from

the list can be calculated as follows:

$$P(proposer) = P(EB * 255 \geqslant MaxEB * r) \;\; where \;\; r \sim U(0, 255), \;\; r = random\_byte, EB = validator\ effective\ balanc$$
$$\therefore P(proposer) = P\left(r \leqslant \frac{255 * EB}{MaxEB}\right)$$
$$\therefore if\ EB = MaxEB \implies P(proposer) = 1 \tag{1}$$

Therefore, if the effective balance of the candidate validator equalled the maximum effective balance, then the validator becomes the proposer with probability 1.

This creates some interesting scenarios when the maximum effective balance is increased to 2,048 ETH. As before, a fully consolidated validator will become a proposer with a probability of 1 if the validator's index was randomly selected as the next candidate.

Given the candidate index belongs to a solo staker with an effective balance of 32 ETH, then

$$P(proposer\ \ check\ \ passed) = P\left(r \leqslant \frac{255 * 32}{2048}\right) = P(r \leqslant 3.98) = \left(\frac{3.98 - 0}{255}\right) = 0.016$$
$$\therefore P(proposer\ \ check\ \ passed) \equiv \left(\frac{32}{2048}\right) = 0.016$$

$\therefore$ *Given the candidate index belongs to a staker with a partially consolidated validator (2 * 32 ETH) EB , then*

$$P(proposer\ \ check\ \ passed) = \left(\frac{64}{2048}\right) = 0.031$$

$\therefore$ *Given the candidate index belongs to a staker with a partially consolidated validator (5 * 32 ETH) EB , then*

$$P(proposer\ \ check\ \ passed) = \left(\frac{160}{2048}\right) = 0.078$$

In other words, the probability of passing the check for selection to propose the next block varies from 0.016 for an unconsolidated validator to 1 for a fully consolidated validator, if that validator's index is selected as the next candidate.

Assuming a validator set size of 716,800, then the probability of any validator being chosen as the candidate index for the next block proposer is:
$P(candidate) = \frac{1}{(Active\ \ validator\ \ set\ \ size)} = \frac{1}{716{,}800} = 0.000001395$, or 0.0001395 %.

Previously, providing a validator maintained its effect balance at 32 ETH, once its index was selected as the next candidate, it would have passed the proposer selection test with certainty (probability of 1, i.e. 100%).

Putting it another way:

Given EB=32 ETH, then currently

$P(candidate\&proposer) = P(candidate) * P(proposer) = 0.000001395 * 1 = 0.000001395$

After MaxEB = 2048 ETH, this changes to:

$P(candidate\&proposer) = 0.000001395 * 0.016 = 0.00000002232$

Let us look at some example scenarios, assuming an active validator set of 716,800 validators, i.e. a total deposit size of $716,800 * 32ETH \approx 22.94\ M\ ETH$:

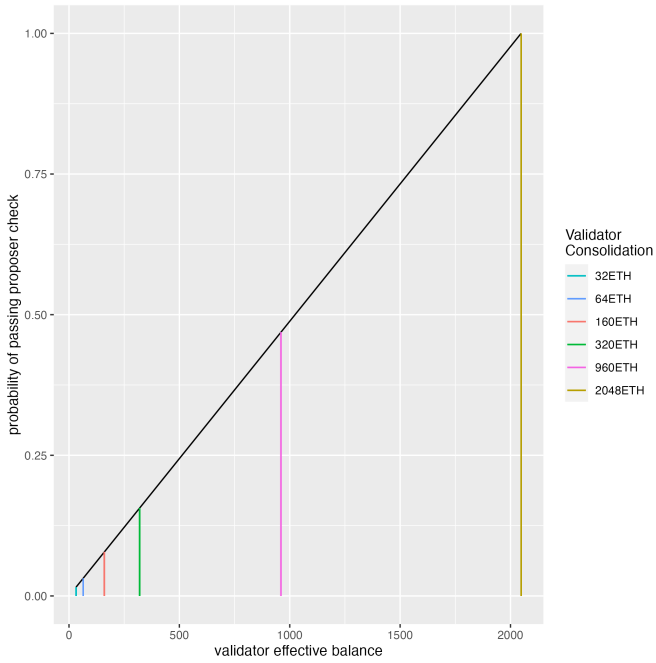


Figure 1: Probability of passing the proposer eligibility check for a candidate validator with an EB ranging from 32 to 2,048 ETH

1. Validator set comprises only validators with an effective balance of 32 ETH.
2. Validator set comprises only validators that are fully consolidated, i.e. each validator has an effective balance of 2,048 ETH.
3. Validator set comprises a combination of validators: solo stakers, partially consolidated stakers and fully consolidated stakers.

Interesting questions:

- What is the probability that no proposer is selected for a slot when no consolidation has yet happened? This situation could occur during implementation time, before stakers merge some or all of their validators.

Scenario 1
In figure 2 on page 8 we generated 716,800 random values from *U(0,255)*

Scenario 2
In this scenario we have full consolidation by all stakers, so probabilities and mechanisms for selection remain unchanged, i.e. when a candidate validator's balance = maxEB, then they will be selected with certainty, i.e. a probability of 1.

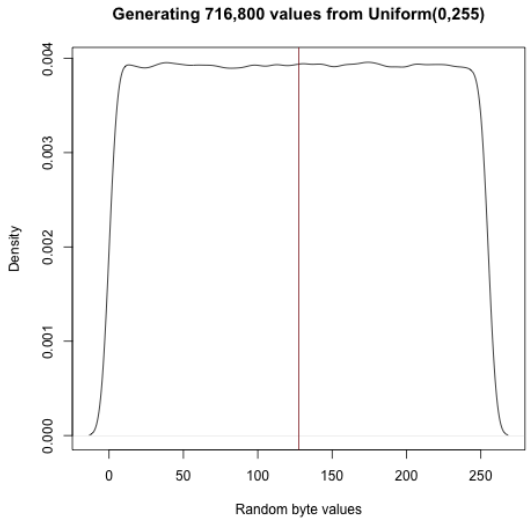
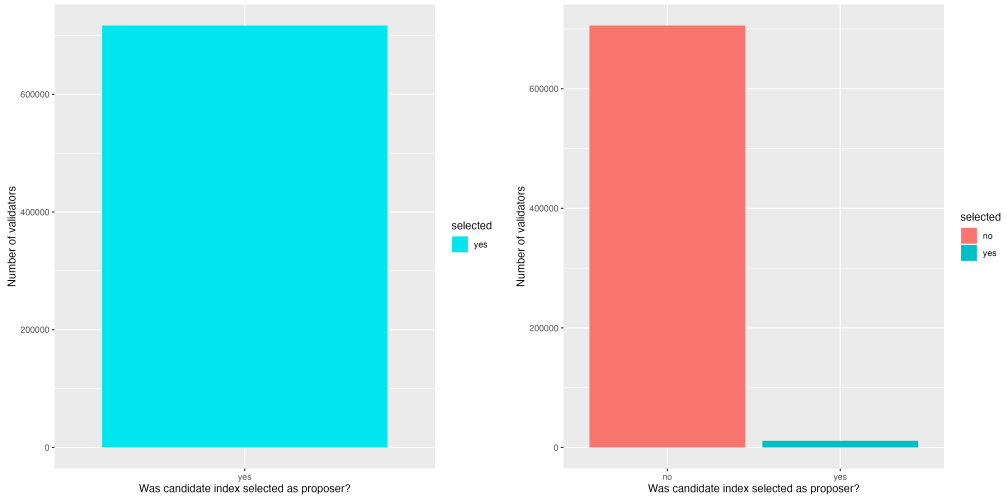


Figure 2: Distribution of 716,800 random bytes generated from U(0,255)



(a) (b)

Figure 3: Proportion of validators with random byte that passes proposer check if all validators have 32 ETH effective balance, and (a) maxEB = 32 ETH, (b) maxEB = 2,048 ETH

Scenario 3
A more interesting scenario is when we look at a variety of effective balances. As an example, let us assume that the validator set of 716,800 validators is made up of a combination of consolidation options as shown in Table 1 on page 9. Based on this chosen configuration, the total validator set size reduces to 297,248:

Number of single validators $= \frac{716{,}800*0.30}{1} = 215,040$
Number of 64ETH validators $= \frac{716{,}800*0.20}{2} = 71,680$
Number of 320ETH validators $= \frac{716{,}800*0.10}{10} = 7,168$
Number of 2,048ETH validators $= \frac{716{,}800*0.30}{64} = 3,360$
Adjusted validator set after consolidation$= 215,040 + 71,680 + 7,168 + 3,360 = 297,248$

We calculate the probability that a validator is selected and passes the check for proposer eligibility, as the product of the probability of being the candidate index and the probability of passing the proposer check, since these two events are independent, i.e. $P(A \cap B) = P(A/B)P(B) = P(A)P(B)$. Using the logic from equation 1 and the example calculations that followed, we populated the table accordingly.

Table 1: Active validator set composition for Scenario 3

Staker Consolidation	**% of total deposit**	**Effective balance**	**P(candidate)**	**P(proposer check = Y)**	**P(candidate & selected)**
Single	30%	32 ETH	$\frac{215{,}040}{297{,}248} = 0.723$	0.016	$0.723 * 0.016 = 0.0116$
Partial (2-fold)	20%	64 ETH	$\frac{71{,}680}{297{,}248} = 0.241$	0.031	$0.241 * 0.031 = 0.0075$
Partial (10-fold)	10%	320 ETH	$\frac{7{,}168}{297{,}248} = 0.024$	0.156	$0.024 * 0.156 = 0.0037$
Fully (64-fold)	30%	2,048 ETH	$\frac{3{,}360}{297{,}248} = 0.011$	1.000	0.0113*1.000 = 0.0113
TOTAL	100%		1.00		

The above scenario tells an interesting story. If the lion share of the total stake is shared equally between single and fully consolidated validators, then it is approximately equally likely for either category to be selected as a proposer of the next block. The varying proportions of, and extent of, consolidation would appear to be less desirable, but we need to put these observations in the context of the total stake for each of the categories in the scenario.

Taking a very simple example, say we have a validator set size of 100, where 32 validators are run by single stakers, and there is 1 large staker running 64 validators. The large staker decides to consolidate all 64 validators into one 'super' validator. Then the probability of the large staker's validator being selected as the candidate index for proposing the next block is $\frac{1}{33} = 0.03$, where previously it would have been $\frac{64}{100} = 0.64$ that one of its validators would be the candidate index selected. Once the check for proposer eligibility is made, the super validator will always pass that check, i.e. probability of 1. For a solo validator to be the selected candidate index would also be 0.03, but the probability that one of the solo stakers is selected, rather than a super validators is $\frac{32}{33} = 0.97$.

The probability that no validator in the active validator set is selected as proposer immediately following the switch to the new maximum effective balance EIP-7251, can be calculated as follows:

We know that if a candidate did not pass the selection process, then the next one in the shuffled index is used to check if they pass the proposer eligibility check.

Therefore we are interested in the probability that every validator in the active validator set failed the eligibility check.

We know that prior to consolidation, all the validators will have an EB of 32 ETH.

$\therefore$ *P(no proposer selected)* $= \binom{n}{k} p^k q^{(n-k)} = \binom{716{,}800}{716{,}800} 0.984^{716{,}800} 0.016^0 = (0.984)^{716{,}800}$

$\therefore$ *P(no proposer selected)* $= \ 0.984^{716{,}800} = 0$

Although for each validator the probability of passing the proposer check is quite low, viz. 0.016, it is highly unlikely, and essentially impossible, for every single validator in the validator set to fail the selection test.

The probability of selecting the first candidate = 1/n, the probability of the next is 1/(n-1) .... $1/1 = \frac{1}{716{,}800!}$

Proposer selection BN

Building on Scenario 3, we can develop a simple Bayesian network (BN) to explore the consequences of various choices. The total number of validators before any consolidation of stake is assumed to be 716,800. Moreover, we will make some assumptions about a possible distribution of consolidated and single validators for various categories of staker. All these assumptions can be replaced with alternative assumptions in the BN.

The diagram in Figure 4 on page 11 is visual representation of the example scenario depicted in the BN.

The following tables describe the key factors (nodes) in the BN and the node probability table (NPT) attached to each node. Figures 5 and 6 depict the BN structure and the resulting marginal probabilities when we run the network, respectively.

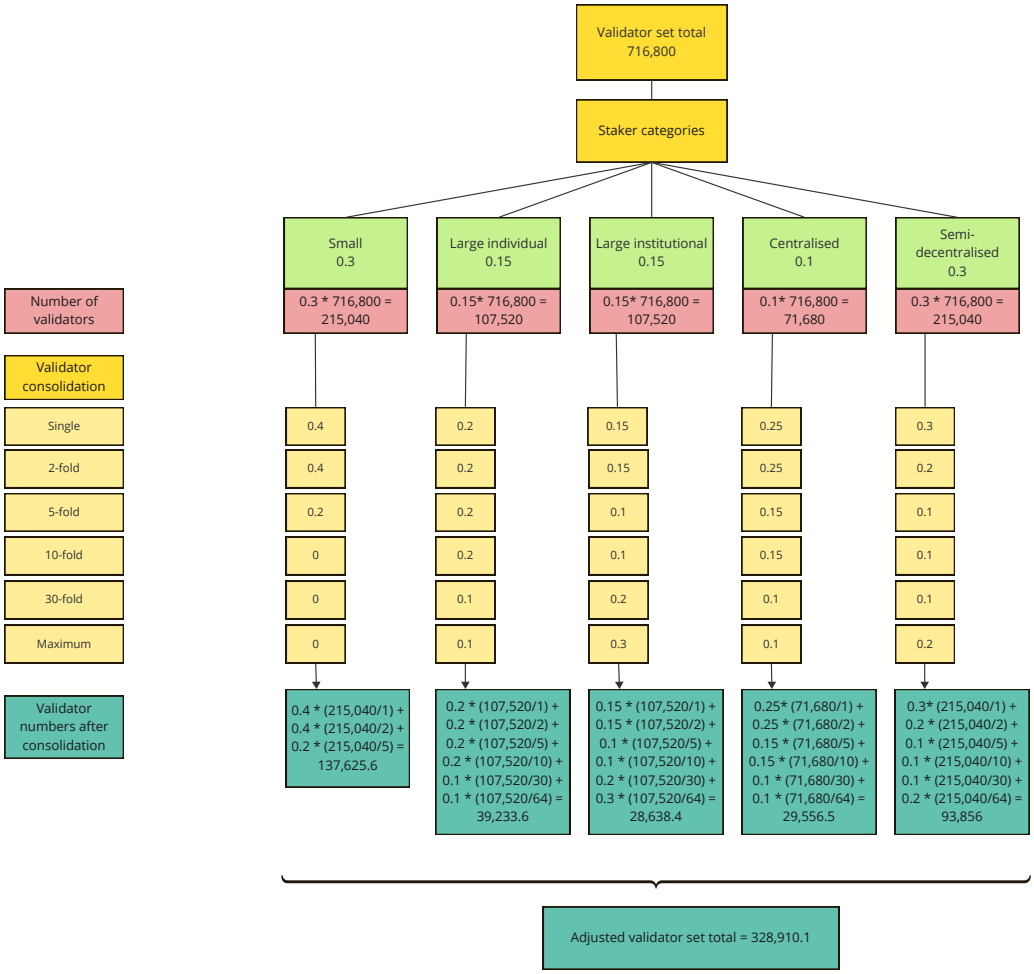


Figure 4: Visual representation of the example scenario for validator consolidation

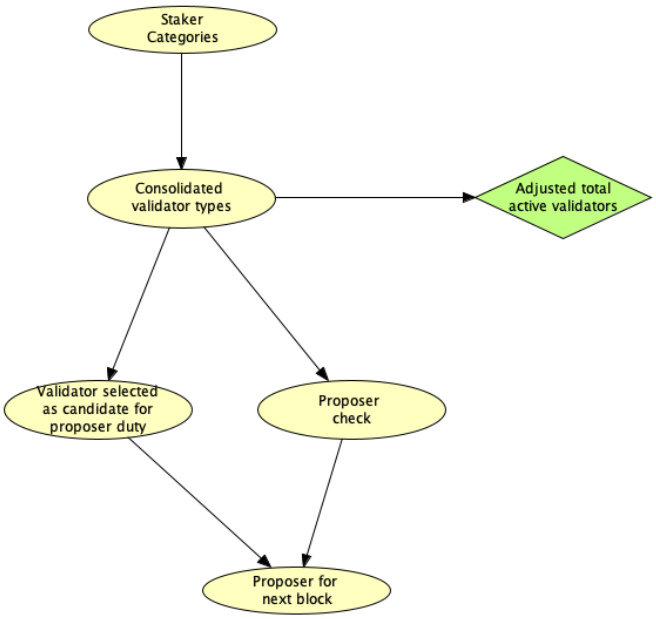


Figure 5: BN for proposer selection after EIP-7251

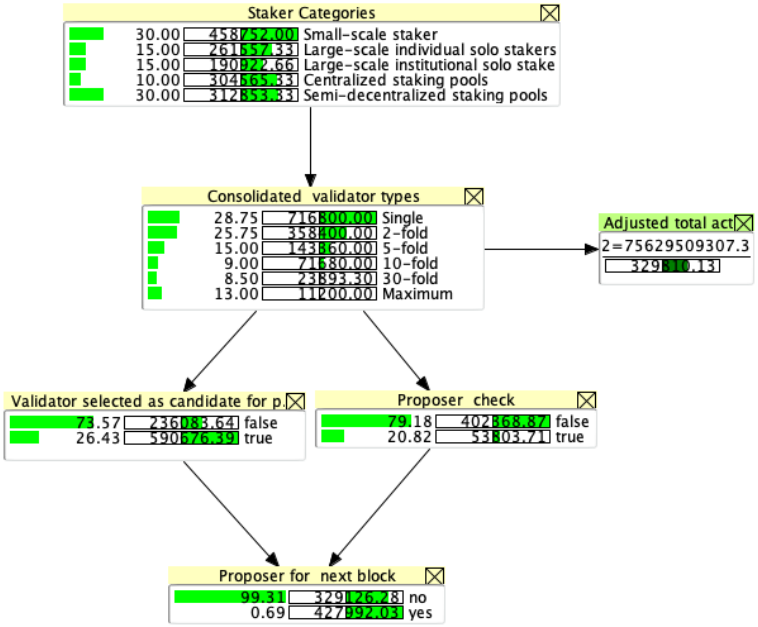


Figure 6: Running the proposer selection BN

Table 2: Staker categories - NPT

Category	**Description**	**Fraction of validator set**
Small-scale solo stakers	32 - a few hundred ETH	0.30
Large-scale individual solo stakers	1000+ ETH	0.15
Large-scale institutional solo stakers	companies staking their own ETH	0.15
Centralized staking pools		0.10
Semi-decentralized staking pools	Rocketpool, Lido... each one is different	0.30

Table 3: Consolidated validator types - conditional probability table (CPT)

Validator consolidation	**Staker categories**				
	Small-scale staker	Large-scale individual staker	Large-scale institutional staker	Centralised staking pools	Semi-decentralised staking pools
Single	0.4	0.2	0.15	0.25	0.3
2-fold	0.4	0.2	0.15	0.25	0.2
5-fold	0.2	0.2	0.1	0.15	0.1
10-fold	0	0.2	0.1	0.15	0.1
30-fold	0	0.1	0.2	0.1	0.1
Maximum	0	0.1	0.3	0.1	0.2

Table 4: Proposer check - CPT

Proposer check passed	**Validator consolidation**					
	Single	2-fold	5-fold	10-fold	30-fold	Maximum
false	0.984375	0.96875	0.921875	0.84375	0.53125	0.0
true	0.015625	0.03125	0.078125	0.15625	0.46875	1.0

Note that the probability that any one validator in the heterogeneous validator set is selected is $\frac{1}{329{,}810} = 0.000003$, but the probability that the validator belongs to a certain 'type' of validator differs as shown in Table 5. Similarly, the probability that a validator from a particular staker is selected as the candidate index is proportional to the number of validators in the active validator set, regardless of whether it is consolidated or not.

Table 5: P(validator selected as candidate proposer)

Validator consolidation	**Marginal probability**	**Number of validators**	**P(selected as candidate)**
Single	28.75%	$\frac{0.2875*716{,}800}{1} = 206,080$	$\frac{206{,}080}{329{,}810} = 0.6248$
Partial (2-fold)	25.75%	$\frac{0.2575*716{,}800}{2} = 92,288$	$\frac{92{,}288}{329{,}810} = 0.2798$
Partial (5-fold)	15.00%	$\frac{0.1500*716{,}800}{5} = 21,504$	$\frac{21{,}504}{329{,}810} = 0.0652$
Partial (10-fold)	9.00%	$\frac{0.0900*716{,}800}{10} = 6,451$	$\frac{6{,}451}{329{,}810} = 0.0196$
Partial (30-fold)	8.50%	$\frac{0.0850*716{,}800}{30} = 2,031$	$\frac{2{,}031}{329{,}810} = 0.0062$
Fully (64-fold)	13.00%	$\frac{0.1300*716{,}800}{2{,}048} = 1,456$	$\frac{1{,}456}{329{,}810} = 0.0044$
TOTAL	100%	329,810	1.0000

Sync committee

As discussed in the blog post, the expectation is that this selection process will work as designed even when there is a large increase in MaxEB [36].

Sync committee members are selected with replacement from the active validator set. The probability of being selected is proportional to the ratio of the validator's effective balance, i.e. the number of increments, n, to the total number of increments in the validator set, T, i.e. $\frac{n}{T}$ [17].

The denominator for the acceptance probability is MaxEB which means the probability will change from $\frac{b}{32}$ to $\frac{b}{2{,}048}$, where b = *validator effective balance*. Check whether this adversely affects validators with a small stake, apart from the expected increase in time to select the committee, as D'Amato and Neuder pointed out [15].

From the discussion in Github it is interesting to note that it was a deliberate choice to select sync committee members **with replacement**. This strategy is necessary if there is a situation where the number of active validators is less than pre-determined sync committee size.

Therefore, theoretically a validator can be chosen more than once to be in the committee, and consequently we can conclude that the size of the next sync committee is always $\leqslant SYNC\_COMMITTEE\_SIZE$.

The committee selection process is handled in *get_next_sync_committee_indices* [17] shown below:

```
def get_next_sync_committee_indices(state: BeaconState) -> Sequence[ValidatorIndex]:
    """
    Return the sync committee indices, with possible duplicates, for the next sync
        committee.
    """
    epoch = Epoch(get_current_epoch(state) + 1)

    MAX_RANDOM_BYTE = 2**8 - 1
    active_validator_indices = get_active_validator_indices(state, epoch)
    active_validator_count = uint64(len(active_validator_indices))
    seed = get_seed(state, epoch, DOMAIN_SYNC_COMMITTEE)
    i = 0
    sync_committee_indices: List[ValidatorIndex] = []
    while len(sync_committee_indices) < SYNC_COMMITTEE_SIZE:
        shuffled_index = compute_shuffled_index(uint64(i % active_validator_count),
            active_validator_count, seed)
        candidate_index = active_validator_indices[shuffled_index]
        random_byte = hash(seed + uint_to_bytes(uint64(i // 32)))[i % 32]
        effective_balance = state.validators[candidate_index].effective_balance
        if effective_balance * MAX_RANDOM_BYTE >= MAX_EFFECTIVE_BALANCE * random_byte:
            sync_committee_indices.append(candidate_index)
        i += 1
    return sync_committee_indices
```

Currently:
$SYNC\_COMMITTEE\_SIZE = 2^9 = 512$ *validators*
$EPOCHS\_PER\_SYNC\_COMMITTEE\_PERIOD = 2^8 = 256$ *epochs*, which is $\approx 27 hours$, i.e. sync committees are roughly in place for a day before the next committee takes over.

The decision to accept a selected candidate validator to participate in the sync committee, follows the same logic as is used for the acceptance of the candidate validator as a proposer. However, in this instance, we continue with the process until we have 255 sync committee members,

which may contain duplicates as mentioned previously.

To demonstrate that this process works as intended we consider a simple example scenario:

Table 6: Simple sync committee example scenario

	Before consolidation	**After consolidation**					
Staker	Single	Single	2-fold	5-fold	10-fold	Maximum	Total
A	128	128	0	0	0	0	128
B	128	0	0	0	0	2	2
C	128	14	5	4	2	1	26
TOTAL	768	142	5	4	2	3	156

Assuming that "After consolidation" in Table 6 represents the decisions made by each of the stakers regarding the validators they run after the implementation of EIP-7521. We will now consider the probabilities prior to and after EIP-7521 for each of these three stakers. Table 7 summarises the various probabilities for sync committee membership.

Table 7: Sync committee probabilities for example scenario

	Before EIP-7251		**After EIP-7521**			
Staker	P(selection)	P(acceptance)	P(sync committee member)	5-fold P(selection)	10-fold P(acceptance)	P(sync committee member)
A	128	128	0	0	0	0
B	128	0	0	0	0	2
C	128	14	5	4	2	1
TOTAL	768	142	5	4	2	3

Slot committees

Selection of the 32 slot committees for each epoch does not take validator effective balance into account. However, when single slot finality is implemented, the fact that different committees may have different weights would not be relevant since all the active stake will vote in each slot.

Currently, however, the weight distribution across slot committees could vary, depending on the percentage of membership of the committees with partially or fully consolidated validators.

Aggregators

For the existing distribution of the expected number of aggregators per sub-committee, refer to Edgington's online book [17]. If more than one of the virtual validators of a consolidated validator is chosen, then it is still just that one validator in the set of aggregators.

3.2 Analysis of Rewards & Penalties

3.2.1 Rewards

Elowsson (2024) published a comprehensive post on ethresear.ch on Ethereum issuance. This post included analysis and modelling of consensus incentives and variability of solo staker reward, as well as several other aspects relevant to issuance and staker yield (Properties of issuance level: consensus incentives and variability across potential reward curves).
Proposer rewards

Proposer award for attestations

Proposer award for sync committees

Attestation rewards

Whistleblower reward

Sync committee rewards

3.2.2 Penalties

Slashing penalties
Listed below are links to posts about slashing penalties, the effect on them when EIP-7521 is implemented, including some proposals for modifications to existing penalties:

1. The cryptoeconomics of slashing by Kannan and Deb.
2. Slashing penalty analysis by Neuder and Monnot.
3. DRAFT] Slashing penalty analysis; EIP-7251 by Neuder and Monnot.
4. MaxEB slashing risks by dapplion
5. Slashing simulation code

In the 3^{rd} blogpost, Neuder and Monnot propose the following for current slashing penalties [37]:

- Changes to existing penalties:
  - Changing the *initial penalty* to be either fixed, or scaled sublinearly
  - Changing the *correlation penalty* to scale quadratically rather than linearly.
- Unchanged penalties:
  - *Attestation penalties*
  - *Inactivity leak penalties*

Initial slashing penalty
The initial slashing penalty is proportional to the validator's effective balance, $MIN\_SLASHING\_PENALTY\_QUO$
32, giving a maximum penalty of 1 ETH if the slashed validator has 32 ETH, the current MaxEB.
If left unchanged, a fully consolidated validator would incur an initial slashing penalty of 64 ETH.

There appears to be general agreement that this initial penalty is too high, and may deter larger stakers such as Lido to support EIP-7521.

Neuder and Monnot suggest that the initial penalty could either be changed to a constant value, or through a monotonically increasing function, e.g. from the family of polynomials. The latter appears to be the preferred option. However, mention has also been made of a zero initial slashing penalty since a slashed validator already incurs additional penalties while waiting for its exit epoch.

For a monotonically increasing function, the authors propose the following function to calculate the revised initial slashing penalty:
$initial\ penalty = \frac{EB^x}{32},\ \ x \leqslant 1$

Their blogpost has graphs for $x = 1, \frac{15}{16}, \frac{7}{8}, \frac{3}{4}, \frac{1}{2}$ and a line for a constant initial penalty of 1 ETH [37].

The authors conclude that visually it appears that $x = \frac{3}{4}$ and $\frac{7}{8}$ are good choices in terms of balancing the size of the initial slashing penalty and the risk for a consolidated validator.

Correlation penalty

This penalty is incurred roughly 18 days (4,096 epochs) after the slashing event, half way between its exit epoch and the slashing event [17]. The correlation penalty is important in penalising apparent coordinated attacks on the chain, and is the only other penalty that Neuder and Monnot propose to alter [37].

The penalty for a slashed validator is calculated as follows using the previous 36 days from this "half-way" epoch.

$$correlation\ penalty = \left\lfloor \frac{3 * EB * SB}{TB} \right\rfloor,\ \ where$$

$3 = Bellatrix\ multiplier$

$EB = slashed\ validator's\ effective\ balance$

$SB = total\ slashable\ balance$

$TB = total\ effective\ balance\ of\ the\ beacon\ chain$

$$\therefore if\ SB = \frac{1}{3} * TB \implies\ \ correlation\ penalty = EB$$

$$Similarly,\ if\ 3 * EB * SB < TB \implies penalty = 0\ due\ to\ integer\ division$$

There is currently no correlation penalty for isolated slashing events and this continues to be the case for a fully consolidated validator with 2,048 ETH effective balance as shown below using 24 million ETH as the total staked ETH, (TB):

$$SB = EB\ for\ an\ isolated\ slashing$$

$$\therefore 3 * EB * SB = 3 * EB * EB$$

$$Assuming\ EB = 2{,}048\ \&\ TB = 2.4 * 10^7,\ then$$

$$3 * EB * EB = 1.2582912 * 10^7 < 2.4 * 10^7$$

$$\therefore penalty = 0$$

For EIP-7521 the authors propose a function that preserves the requirement that when the total slashed balance is $\frac{1}{3}^{rd}$ of the total balance, the entire balance of the validator is slashed:

$$penalty' = \frac{3^2 * EB * SB^2}{TB^2}$$
$$\therefore if\ SB\ = \frac{TB}{3},\ then$$
$$penalty' = \frac{3^2 * EB * \left(\frac{TB}{3}\right)^2}{TB^2}$$
$$\therefore penalty' = EB$$

Moreover, the new correlation penalty function scales quadratically as opposed to the current function that scales linearly. With the proposed new function, the slashing penalties are substantially reduced for all validators, regardless of whether they have consolidated stake.

Refer to the blogpost to view graphs showing the comparative correlated slashing penalties for solo (32 ETH), partially consolidated (256 ETH), and fully consolidated (2,048 ETH) validators applying both the current and the proposed functions [37].

Attestation penalty

Once a validator is slashed, their attestations (source, target and head votes) are deemed to be invalid and hence they incur attestation penalties for 8,192 epochs until their exit epoch.

Different weights are attached to each vote, but only the source (weight = 14) and target (weight = 26) votes incur penalties. For each of the 8,192 epochs the slashed validator will incur:

$$Given:$$
$$base\ reward = \frac{64}{\left\lfloor\sqrt{TB}\right\rfloor},\ weight\ denominator = 64,$$
$$source\ weight = 14\ \&\ target\ weight = 26$$

$$\therefore epoch\ attestation\ penalty = \frac{base\ reward * EB * (14 + 26)}{64}$$
$$\therefore\ \ if\ TB \approx 24 million\ ETH = 2.4 * 10^6 * 10^9\ \ Gwei$$
$$the\ integer\ square\ root\ of\ 2.4 * 10^6 * 10^9\ \ Gwei = 154,919,333$$
$$base\ reward = \frac{64 * 10^9}{154,919,333} = 413\ \ Gwei$$

$$\therefore for\ a\ \textbf{\textit{solo staker}}\ with\ 32\ ETH:$$
$$total\ attestation\ penalty\ for\ 8{,}092\ epochs = 8192 * \frac{413 * 32 * 40}{64}\ \ Gwei \approx 0.06767\ \ ETH$$
$$for\ a\ \textbf{\textit{fully consolidated}}\ validator\ with\ 2{,}048\ ETH:$$
$$total\ attestation\ penalty\ for\ 8{,}092\ epochs = 8192 * \frac{413 * 2048 * 40}{64}\ \ Gwei \approx 4.331\ \ ETH$$

This attestation penalty for a large slashed staker seems acceptable, but could potentially be adjusted by changing the number of epochs that the validator is deemed as being "offline".The size of this penalty needs to be such that the security model is not compromised. In other words it should never be a better option to self-slash to avoid inactivity penalties. Therefore, it needs

to be greater than the inactivity penalties for an unslashed validator that is exiting and offline [37].

Inactivity leak penalty
An *inactivity leak* is signalled by the protocol when the chain has not been finalising for 4 epochs. During an inactivity leak online validators will not penalised, so although no rewards are being earned, the penalty is 0.

On the other hand, offline validators, which includes slashed validators waiting to exit, start 'leaking' state. The loss of stake means that the relative weight of the online validators will increase, which helps the chain to start finalising again. The inactivity penalty can be quite severe.

Using the current method, the authors calculated the penalty for three different effective balances: 32 ETH (solo validator), 256 ETH (partially consolidated validator), and 2,048 ETH (fully consolidated validator).

Table 8: Inactivity leak penalties

validator size	16 epoch leak	128 epoch leak	1024 epoch leak
32 ETH	0.000259 ETH	0.0157 ETH	1.00 ETH
256 ETH	0.00208 ETH	0.126 ETH	8.01 ETH
2048 ETH	0.0166 ETH	1.01 ETH	64.1 ETH

Potential slashing during consolidation
The points summarised here arose from discussions amongst several people, including Lion, Mikhail Kalinin, Francesco, Barnabé Monnot, Mike Neuder, Roberto Saltini.

Importantly, we need to assess what effect the implementation of EIP-7521 could have on the risk of being slashed.

As mentioned earlier Lion created a model for slashing risk profile. The link to the source Python code has edit permissions, with an accompanying document on MaxEB Slashing Risk.

Scenarios:

1. Validation keys are stolen

   The balance is secured by withdrawal credentials, therefore malicious consolidation is not anticipated to be valuable for an attacker.

   Nonetheless, it would be helpful to know how pools manage withdrawals. To the best of our knowledge LIDO has single withdrawal credentials for all validators, and Rocket Pool different withdrawal credentials for each validator. On the other hand, centralised stakers, such as Coinbase, apparently try to obfuscate and will therefore probably have different sets of credentials.

2. Chain forks during consolidation of validators

   It is vital that we keep track of slashing across both the source and target for consolidation, rather just one or the other. Francesco demonstrated the need for this by visualising the scenario in figure 7 on page 21. The scenario is described below.

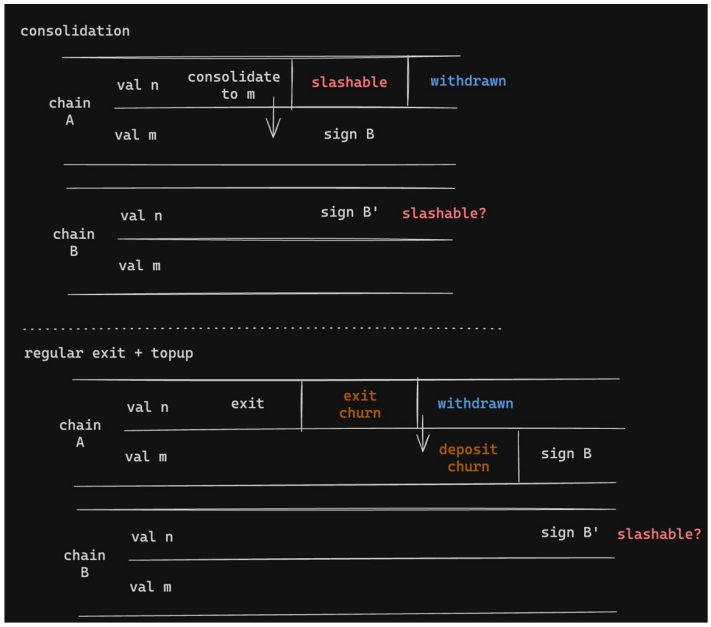


Figure 7: A fork where consolidation is yet to occur and one where it has happened

Scenario 2 - fork during consolidation

In figure we see that validator n consolidates into validator m on chain A, but not on chain B. Validator n then signs two different blocks at the same slot, or does surround voting, in a way that is not deemed to be slashable in the existing consolidation pull request (PR). Based on this potential scenario, Francesco wondered if it would be that complicated to enable slashing "across consolidations".

Mikhail suggested that this would be possible, providing that chain B is aware that the consolidation happened on chain A. Moreover, a slasher would need to resolve the final consolidation index in order to detect a new source of slashable offences between the source and target of consolidated validators.

Mikhail suggested three options to enable slashing across consolidations:

1. pass the consolidating stake through the activation churn.
   Bypassing the exit churn would be the improvement for stakers and a consolidation to be reverted if the source is slashed before the EB was fully activated, without the necessity to make the target liable for the source's slashings.
   Milhail feels that this option is more natural from the protocol perspective.

2. finalise consolidation on-chain before activating a consolidated balance. This is akin to waiting for *activation_eligibility_epoch* to be finalised.
   This is Mikhail's preferred option, with the addition of improving the slasher's design.

3. accompanying the slashing message with the consolidation from another chain.

3.3 Analysis of Ethereum ecosystem & protocol

3.3.1 Worst case: Full consolidation

3.3.2 Griefing / Discouragement attacks

According to Buterin a griefing attack is when a validator acts maliciously inside a consensus mechanism to reduce other validators' revenue even at some cost to themselves to encourage the victims to drop out of the mechanism [8].

The two main motivations for reducing the number of participants are most likely because fewer participants:

- mean greater rewards for those remaining in the mechanism
- helps to prepare an attack on the chain by reducing the cost of an attack

Some strategies have already been put in place to avoid discouragement attacks [17]:

- inverse square root scaling of validator rewards
- scaling of rewards with participation (viz. for each "source, target, and head vote, the attester's reward is scaled by the proportion of the total stake that made the same vote")
- zeroing attestation rewards during an inactivity leak
- rate limiting of validator exists, which means that an attacker needs to sustain an attack for longer and at greater cost in order to achieve the same outcome.

3.3.3 Consolidation of validators

3.3.4 Withdrawals

3.3.5 Centralisation forces

Historic and current trends for stake concentration are important to observe as these are warning signs that the chain is becoming more vulnerable to collusion. The consolidation of stake through EIP-7251 (currently in draft form) [34] is unlikely to change the dynamics of stake concentration, since it is encouraging consolidation of validators already being operated by stakers. However, with more node operators and validators joining larger staking pools staked ETH will become more one-sided in favour of staking pools.

3.3.6 Bayesian network model

We build a BN model to capture the key factors and interactions for an increase in MaxEB as proposed in EIP-7521.

We assume that different groups, or categories, of stakers have different characteristics which will in turn influence their willingness to consolidate stake, as well as the uptake of validator consolidation.

Key factors

Staker categories
Vitalik Buterin suggested potential categories as follows:

- Small-scale solo stakers (32 - a few hundred ETH)
- Large-scale individual solo stakers (1000+ ETH)
- Large-scale institutional solo stakers (ie. companies staking their own ETH)
- Centralised staking pools
- Semi-decentralised staking pools (Rocketpool, Lido... each one is different)
- Each of the above, but using distributed validator technology (DVT)

Although each of the categories will differ with the use of DVT, we do not anticipate that DVT will influence their behaviour with respect to validator consolidation in any meaningful way.

It will be challenging to assign a distribution across the proposed staker categories. However with current work on identifying validators, we would be able to assign some validators to staker categories and for those that are unknown, we can either replicate the distribution across the validators, or adjust them based on an assumption on the probability distribution across the unknown group. With the BN we can also apply other distributions and see to what extent they affect the downstream probabilities.

Validator consolidation
The implementation of EIP-7521 is likely to eventuate in a varying degree of consolidation within and across the various categories. We can use intuition for an initial uptake and vary this to test the downstream consequences of various scenarios.

Health of the Ethereum ecosystem Ether alpha combines various aspects of the ecosystem to give an overall impression of the health of the network. They pull information from various sources for 'Project Sunshine' dashboard [3].

The Rated network also provides detailed metrics and visualisations to gauge the health of the network [33].

4 Bibliography

References

[1] Laurence Aitchison, Nicola Corradi & Peter E Latham (2016): *Zipf's Law Arises Naturally When There Are Underlying, Unobserved Variables. PLoS computational biology* 12(12), p. e1005110, doi:10.1371/journal.pcbi.1005110.

[2] Ether alpha (2023): *Client diversity: Resource site to assist in Ethereum client diversity efforts.* Available at `https://clientdiversity.org/`.

[3] Ether alpha (2023): *Project Sunshine.* Available at `https://ethsunshine.com/`.

[4] Ether alpha (2023): *Validator queue: Dashboard to monitor validator enter/exit queues and wait times.* Available at `https://www.validatorqueue.com/`.

[5] Aditya Asgaonkar (2023): *Removing Unnecessary Stress from Ethereum's P2P Network.* Available at `https://ethresear.ch/t/removing-unnecessary-stress-from-ethereums-p2p-network/15547`.

[6] Andrew Breslin (2022): *What is staking?* Available at `https://consensys.net/blog/ethereum-2-0/what-is-staking/`.

[7] Vitalik Buterin (2017): *The Meaning of Decentralization.* Available at `https://medium.com/@VitalikButerin/the-meaning-of-decentralization-a0c92b76a274`.

[8] Vitalik Buterin (2018): *Discouragement Attacks.* Available at `https://eips.ethereum.org/assets/eip-2982/ef-Discouragement-Attacks.pdf`.

[9] Vitalik Buterin (2020): *Vitalik's Annotated Ethereum 2.0 Spec.* Available at `https://notes.ethereum.org/@vbuterin/SkeyEI3xv`.

[10] Vitalik Buterin (2023): *Don't overload Ethereum's consensus.* Available at `https://vitalik.ca/general/2023/05/21/dont_overload.html`.

[11] Vitalik Buterin (2023): *Paths toward single-slot finality.* Available at `https://notes.ethereum.org/@vbuterin/single_slot_finality`.

[12] Rajan Chattamvelli (2020): *Discrete distributions in engineering and the applied sciences.* Synthesis lectures on mathematics and statistics ; 34, Morgan & Claypool Publishers.

[13] Liquid Collective (2023): *Ethereum's activation and exit queues.* Available at `https://liquidcollective.io/eth-activations-and-exits/`.

[14] Matt Corva & Bill Hughes (2023): *Staking is Data Validation, Not Investment.* Available at `https://consensys.net/blog/news/staking-is-data-validation-not-investment/`.

[15] Francesco D'Amato & Mike Neuder (2023): *Security Considerations and Spec Changes for a MAX_EFFECTIVE_BALANCE Increase.* Available at `https://notes.ethereum.org/@fradamt/meb-increase-security`.

[16] dapplion (2023): *Ethereum specs pull request: Add upper epoch churn limit #3448.* Available at `https://github.com/ethereum/consensus-specs/pull/3448`.

[17] Benjamin Edgington (2023): *A technical handbook on Ethereum's move to proof of stake and beyond*. Technical Report. Available at `https://eth2book.info/latest`.

[18] ethereum.org (2022): *Pooled staking*. Available at `https://ethereum.org/en/staking/pools/`.

[19] ethereum.org (2022): *Solo staking*. Available at `https://ethereum.org/en/staking/solo/`.

[20] ethereum.org (2022): *Staking as a service (Saas)*. Available at `https://ethereum.org/en/staking/saas/`.

[21] ethereum.org (2022): *Staking with Ethereum*. Available at `https://ethereum.org/en/staking/`.

[22] Etherscan (2023): *BeaconScan: The Official Etherscan Beacon Chain Ethereum 2.0 Explorer*. Available at `https://beaconscan.com/`.

[23] Adem Efe Gencer, Soumya Basu, Ittay Eyal, Robbert Van Renesse & Emin Gün Sirer (2018): *Decentralization in Bitcoin and Ethereum Networks*. Technical Report, Berlin, Heidelberg. arXiv:1801.03998v2.

[24] Hex (2023): *Lido on Ethereum Validator & Node metrics*. Available at `https://app.hex.tech/8dedcd99-17f4-49d8-944e-4857a355b90a/app/3f7d6967-3ef6-4e69-8f7b-d02d903f045b/latest`.

[25] Viet Tung Hoang, Ben Morris & Phillip Rogaway (2014): *An Enciphering Scheme Based on a Card Shuffle*. arXiv:1208.1176.

[26] Charles I Jones (2015): *Pareto and Piketty: The Macroeconomics of Top Income and Wealth Inequality*. *The Journal of economic perspectives* 29(1), pp. 29–46.

[27] Aris Koliopoulos (2023): *Solo stakers: The backbone of Ethereum*. Available at `https://blog.rated.network/blog/solo-stakers`.

[28] Dániel Kondor, Márton Pósfai, István Csabai & Gábor Vattay (2014): *Do the rich get richer? An empirical analysis of the Bitcoin transaction network*. *PloS one* 9(2), pp. e86197–e86197.

[29] Qinwei Lin, Chao Li, Xifeng Zhao & Xianhai Chen (2021): *Measuring Decentralization in Bitcoin and Ethereum using Multiple Metrics and Granularities*, pp. 1–8. arXiv:2101.10699.

[30] Nansen (2023): *Eth2 Dashboard*. Available at `https://pro.nansen.ai/eth2-deposit-contract`.

[31] Nansen (2023): *The Ethereum Shanghai (Shapella) Upgrade Dashboard*. Available at `https://query.nansen.ai/public/dashboards/Hk93n66vsO0uvycfui8ypF2xcpNhpraxfwX5AWZJ`.

[32] U Natale (2022): *Analyzing Ethereum Cryptoeconomics: the validator's perspective*. Technical Report, Chorus. Available at `https://docs.google.com/document/d/1r640UQOm2z-Q9nsJzqBq3BVgCtTL1_Yc7WnPp4jEBgk`.

[33] Rated network (2023): *Network overview*. Available at `https://www.rated.network/overview?network=mainnet&timeWindow=all&rewardsMetric=average`.

[34] Mike Neuder (2023): *EIP 7251 Maximum effective balance increase proposal [DRAFT]*. Available at https://github.com/michaelneuder/EIPs/blob/max-eb-increase/EIPS/eip-increase-maxeb.md.

[35] Mike Neuder (2023): *Security Considerations and Spec Changes for a MAX_EFFECTIVE_BALANCE Increase*. Available at https://notes.ethereum.org/nHqON5l7SACkL_nPwz8Vqw.

[36] Mike Neuder, Francesco D'Amato, Aditya Asgaonkar & Justin Drake (2023): *Increase the MAX_EFFECTIVE_BALANCE – a modest proposal*. Available at https://ethresear.ch/t/increase-the-max-effective-balance-a-modest-proposal/15801/3.

[37] Mike Neuder & Barnabé Monnot (2023): *[DRAFT] Slashing penalty analysis; EIP-7251*. Available at https://notes.ethereum.org/@mikeneuder/slashings-eip-7251.

[38] Rocket Pool (2023): *Rocket Pool: How Ethereum Staking Works*. Technical Report. Available at https://docs.rocketpool.net/guides/staking/overview.html#how-ethereum-staking-works.

[39] Roberto Saltini (2023): *Upper bound on the probability of one majority dishonest committee in the context of MAX_EFFECTIVE_BALANCE increase*. Available at https://notes.ethereum.org/nHqON5l7SACkL_nPwz8Vqw.

[40] James Smith & Rodrigo Vasquez (2023): *Staking survey: Key trends, take aways, and predictions*. Technical Report, Ethereum Foundation and EthStaker. Available at https://lookerstudio.google.com/u/0/reporting/cafcee00-e1af-4148-bae8-442a88ac75fa/page/p_ja2srdhh2c.

[41] Balaji S. Srinivasan & Leland Lee (2017): *Quantifying Decentralization*. Available at https://news.earn.com/quantifying-decentralization-e39db233c28e.

[42] Gavin Wood (2016): *Ethereum: a secure decentralized generalised transaction ledger*. Available at https://github.com/ethereum/yellowpaper.