Non-Linear Servicer Stake Weighting

Analysis and justification of a non-linear stake weighting update of the PIP-22 parameters.

Parameter Update Proposal by:
Ramiro Rodríguez Colmeiro (ramiro@poktscan.com)
Pablo Frigerio (pablo@poktscan.com)
Nicolas Aguirre (nicolas@poktscan.com)
POKTscan Data Science Team

November 17, 2022

Disclaimer

This work was produced by Pocket Scan Technologies LLC, a company operating on the Pocket Network.

Contact

Pocket Scan Technologies LLC
1621 Central Ave
Cheyenne, WYOMING (WY) 82001

Contact: Michael ORourke (michael@poktscan.com)

Changelog

v0.1	2022-09-13	Data collection.
v1.0	2022-09-21	First version.
v1.1	2022-11-17	Corrected equation 11 and algorithm 3 as noted by @msa6867 (Mark) .

Table of Contents

1 Abstract

This proposal intends to modify the current linear stake weight model of PIP-22 and enable a non-linear stake weight model. The mechanism for such change is already included in the PIP-22, therefore this proposal is only a parameter update proposal.

This modification is required to address a fairness issue that reduces the rewards of base stake nodes and to enable a stake compounding strategy that is beneficial for the quality of service of the network.

We propose to change the $ServicerStakeFloorMultiplierExponent$ to a value lower than 1.0 (under current network state it should range from 0.7 to 0.8). The need for this change roots in the limitations of the linear model to keep inflation constant without reducing the gain of nodes with base stake (those who decided to not perform any compounding). Forcing the compounding of stake was not part of the PIP-22, we want to correct this effect.

In this document we describe the main mechanisms of the PIP-22 and the effects of the linear stake weight modeling. Our main conclusions are:

- The linear weighting model penalizes the nodes that do not perform compounding. We estimate that non-compounded nodes are loosing at least $\sim\ 15\%$ of their rewards.
- With the linear staking model the only valid strategy is to compound at maximum stake. This creates a de-facto increase of the minimum viable stake.
- Observing only the variations of the number of sessions by node does not gives enough information on how the number of relays is modified. Hence, the reduction of the rewards for the base stake nodes is not correctly justified.
- If the performance of the nodes in each bin (measured in relays) is not similar, the calculation of the $ServicerStakeFloorMultiplier$ does not result in constant inflation. Correcting this problem leads to further penalization of nodes in the base stake.

2 Servicer Stake Weighting Overview

As it is publicly known, on August 29th 2022 at 19:31:42 UTC on block 69232 the Servicer Stake Weighting (SSW) proposed in PIP-22 [3] was activated using the parameters defined by the PUP-21 [5]. The activation of this feature enabled the *accumulation* of stake in the servicer nodes in exchange of higher payments per relay done by the node. In the following sub-sections we will try to explain the implications of these new features in the Pocket Network ecosystem, their immediate effects and the possible future scenarios.

2.1 Mechanics of the PIP-22

The stake weighting feature modifies how a node is paid for its service. Prior the PIP-22 the nodes were all paid a fixed amount for each relay that they processed:

$$P_n = R_n K, \tag{1}$$

where R_n is the total number of relays done by the n^{th} node of the network, K a value determined by the Decentralized Autonomous Organization (DAO) and P_n the amount of POKT that was paid to that node. With the activation of the PIP-22 the nodes are able to stake more than 15×10^3 POKT in order to receive a higher amount of POKT for each relay that they perform. This new amount is computed as follows:

$$P_n = R_n K W_n, \tag{2}$$

where W_n is the stake wight multiplier of the n^{th} node. This new multiplier is determined by the amount of POKT that is staked in the servicer node and a new set of parameters determined by the DAO, which currently are:

- SSFM: ServicerStakeFloorMultiplier
- SSWM: ServicerStakeWeightMultiplier
- SSWC: ServicerStakeWeightCeiling
- SSFME: ServicerStakeFloorMultiplierExponent

Using these new parameters the value W_n is produced following algorithm 1.

2.2 Calculation of the PIP-22 parameters - Linear Weighting

The parameters that control the value W_n in equation 2 (SSFM, SSWM, SSWC and SSFME) are determined by the DAO. These parameters are calculated to keep the network minting constant. An explanation of how this happens in a single chain scenario can be found in section 3.2 of our previous report on this subject [4].

Algorithm 1 Calculate node stake weighting

procedure Calculate W_n(Node Stake, SSFM, SSWM, SSWC, SSFME)
  Floored Stake $\leftarrow$ Node Stake $-$ (Node Stake%SSFM)
  Capped Floored Stake $\leftarrow \min($Floored Stake, SSWC$)$
  $W \leftarrow \dfrac{\left(\dfrac{\text{Capped Floored Stake}}{\text{SSFM}}\right)^{\text{SSFME}}}{\text{SSWM}}$
  Return w.
end procedure

In the linear weighting scenario, the only parameter being adjusted is the SSWM. This parameter is obtained as shown in algorithm 2. The SSWM is based on the mean of the stake compounding across all chains, weighted by each chain's traffic.

Algorithm 2 Calculate ServicerStakeWeightMultiplier for Linear weighting

procedure Calculate SSWM
  Chain List $\leftarrow$ get serviced chains
  Acum. SSWM $\leftarrow 0$
  Acum. R $\leftarrow 0$
  for chain $\in$ Chain List **do**
    Acum. Stake $\leftarrow 0$
    Num. Nodes $\leftarrow 0$
    Nodes List $\leftarrow$ get nodes staked in this chain
    for node $\in$ Nodes List **do**
      Acum. Stake$+=$ node[Capped Floored Stake]
      Num. Nodes$+= 1$
    end for
    Base Nodes $\leftarrow \dfrac{\text{Acum. Stake}}{\text{SSFM}}$
    Chain Relays $\leftarrow$ get relays done in this chain
    Acum. SSWM$+= \left(\dfrac{\text{Base Nodes}}{\text{Num. Nodes}}\right)$ Chain Relays
    Acum. R$+=$ Chain Relays
  end for
  Return $\dfrac{\text{Acum. SSWM}}{\text{Acum. R}}$.
end procedure

3 Effects of the linear PIP-22 on the Pocket Network

The mechanics introduced with PIP-22 have a large effect on the Pocket Network landscape, in this section we try to: Describe the discussions behind the approval of the PIP-22; Show the current state of the network; Discuss the probable future scenarios of the linear weighting.

3.1 Network state prior and post PIP-22

Prior the activation of the PIP-22 the network was composed of $\sim\ 35 \times 10^3$ *servicing* nodes [1]. The total staked POKT was approximately $35 \times 10^3\ 15 \times 10^3$POKT $= 525 \times 10^6$POKT. The price of the POKT token at that time was around the 0.09 USD for each POKT mark, meaning that most of the nodes with a low average minting ($\sim\ 25.67$ POKT) per node by day were earning less than the costs of the servers they ran. The community proposed, among other things, to enable the stake weighting and *consolidate* the stake of several nodes into a single node which would be paid more for each relay. To put if simple, if a node runner had 4 nodes he/she could consolidate the stake of the 4 nodes into a single node that would be paid 4 four times more than a non-consolidated node, thus cutting the node running costs by four and leave the POKT income untouched. An other more technical aspect of the PIP-22 was that the number of nodes was growing to large and the blocks were becoming difficult to process due to their size. Without entering in technical details, this proposal had some details that were mentioned during the discussion:

1. The total minted could not interfere with the WAGMI proposal [2], which controlled the total minting. This limitation meant that the total POKT minted, with or without SSW, should remain constant. This increment in minting could appear due to increment of the expected sessions by node. A node in a network without compounded nodes has a lower number of sessions than the same node in a network with the same amount of staked POKT but with nodes with compounded stake. This means that it was expected to an increase in the number of relays, this justifies a node weighting $W_n < 1.0$ for nodes with stake equal to SSFM (i.e. in the first weight bin).
2. It was not possible to know the sources of the POKT that was going to be used for node compounding. It was initially proposed that the POKT would come from the un-staking of nodes. If a node runner had 4 nodes, he/she would un-stake 3 of them in order to create a new node with 4 times the initial stake. In this scenario the number of nodes was expected to fall to number near $\sim\ 20 \times 10^3$ nodes. The possibility of external POKT entering to the system was not taken into account as an important source of compounding POKT.

[1] We consider the network state at the exact time that PIP-22 was approved. It can be argued that the number of nodes was higher, since by the time of the PIP-22 approval some un-staking was done in preparation for it, however it is difficult to know how much was due to unprofitability and how much due to PIP-22. In either case the total number of nodes at this time does not play a central role in this proposal.

3. It was also not possible to know which kind of Quality of Service (QoS) node runners were going to compound. We expected low QoS nodes to compound faster due to economic pressure, others stated that high QoS nodes would compound faster due to being more proactive than low QoS node runners. The proposal expected the QoS of the compounded nodes to be similar to the network average (no correlation between QoS and compounding probability).
4. The effects of this proposal on node runners without the capital to compound into a higher bin was not analyzed. Nevertheless all stake bins should have been taken into account in the model of this proposal. The Pocket Network's nodes base reward should remain unchanged, the proposal intended to enable additional rewards due to more staked POKT, not the reduction of the reward for nodes with the base stake.
5. This proposal did not analyze the QoS of the nodes and how the reduction of the number of nodes could impact this critical aspect of the Pocket Network. It is known that the Pocket Network was overprovitioned by the time of the PIP-22 (and probably it still is), however the changes introduced by the PIP-22 added an extra degree of freedom to the node runner, it is possible to either have a high QoS or a high stake in order to produce more rewards.

After the activation of the PIP-22 the SSW parameters were calculated using algorithm 2 and the resulting (current) multipliers are presented in table 1. However, the network is shaping to be different. The reduction of the number of nodes was not as large as expected, currently the number of nodes is $\sim\ 28 \times 10^3$. The total staked POKT rose to $\sim\ 624 \times 10^6$, meaning that much of the POKT used to compound nodes came from external sources [2]. Also there seems to be a correlation between the QoS of the node runner and the compounded nodes proportion. This correlation leads to a bias in the calculation of SSWM using algorithm 2, which expects the compounded nodes to have a minting level similar to the network's average. The observed changes between the prior PIP-22 network and the current state of the network can be seen in table 2.

Table 1: Stake weighting multipliers with PUP-21 values and ValidatorStakeWeightMultiplier $= 1.4687$ (2022-09-13)

Stake	Multiplier
15 kPOKT	0.6809
30 kPOKT	1.3617
45 kPOKT	2.0426
60 kPOKT	2.7235

Table 2: Main features of the Pocket Network previous and post PIP-22 implementation.

	Peak	prior PIP-22	post PIP-22	change from peak	change from PIP-22
Staked Nodes	45×10^35	35×10^35	28×10^35	-20%	-37.7%
Staked POKT	675×10^6	525×10^6	624×10^6	18.8%	-7.75%

[2] This point can be argued depending on how one take the initial number of nodes, if the peak of $\sim\ 45 \times 10^3$ nodes is taken, then there was a reduction of the total stake, which cannot be explained due to PIP-22.

3.2 Linear Stake Weighting Future

It is difficult to forecast what will happen to the Pocket Network nodes distribution, but given the observed behavior during the transition to stake weighted rewards we can expect the following:

1. The number of nodes will not decrease as much as expected. Further reductions could be seen, but the mark of $\sim 20 \times 10^3$ servicer nodes will not be reached in the short term.
2. The node runners will try to push their nodes to the max stake, increasing the percentage of nodes in the highest bin. This clearly the best strategy, as the cost of running a node is fixed and the return of POKT per POKT invested is constant (when staking at bin edges).
3. The number of nodes with the base stake, 15×10^3 POKT and all other stake bins except the maximum, will continue to decline probably due to additional stake rather than compounding. The current network has increased its total stake by $\sim 20\%$, and it is natural to think that idle POKT generated from new minting will be used to increase the stake of those nodes that are not at max stake.

The extreme case of the previous points lead to a network were the majority of the nodes are in the highest stake bin. In such network, the application of algorithm 2 will result in a SSWM ≈ 4.0, which in turn (applying algorithm 1) results in a multiplier $W_{60\text{ K}} \approx 1.0$ and $W_{15\text{ K}} \approx 0.25$. This will defeat the purpose of stake weighting and reduce the PIP-22 effects to a mere increase of base stake, since all other bins will be highly penalized. We believe that before reaching this situation the DAO will have to modify the SSW the parameters. Two possible amendments could be:

1. Increase the stake wight floor SSFM, the stake cap SSWC and/or increase the node base stake (StakeMinimum).
2. Apply a non-linear staking, lowering the incomes of nodes in the higher bins in order to keep minting constant for the base level nodes.

The first of these options is the most simple course of action, however this would only prolong the race but the result in the long term will be the same. The pressure will be put in the lower stake bins until they are forced out of the system, generating an endless and self-defeating pursuit. Moreover this is not compatible with future needs of the Pocket Network, i.e. if in the future the more nodes are needed due to an increase of the network relays, the stake weighting mechanism should be turned off to encourage the breaking-up of high stake nodes. There is no middle ground, the linear staking will always push to maximum stake for highest profitability.

The second option is to change the setting to a non-linear scheme, using SSFME < 1, we will develop and further justify this solution in the following sections.

4 Effects of Linear Servicer Stake Weighting on Base Stake Nodes

In a previous report we theorized that the increase in the number of relays served by a node, that appears as consequence of the reduction in the total network nodes, would not suffice to compensate the reduction of POKT paid by relays after the implementation of SSW (a reduction due to a $W_{min} < 1$). We modeled the Cherry Picker [1] behavior and computed a Monte Carlo simulation [7] of the expected outcome. Our study [4] showed that under some strong assumptions, non-compounded nodes were loosing $\sim 14\%$ of their expected POKT gains. A secondary conclusion that derives from this study is that the increase in the session selection probability does not directly results in an increase of the relays, as the linear weighting expects. This can be seen from the fact that the SSWM is equal the to the increase of session selecting probability (see line 15 of algorithm 2) and then it's inverse value is used to calculate $W_{15\text{ K}}$ (see line 4 of algorithm 1), we will properly justify this in section 4.5.

In the following sections we pretend to support the conclusions of the previous report using data from before and after the PIP-22 implementation. The data before the PIP-22 is from before 2022-06-30 (the PIP-22 approval date) and the data after the PIP-22 is data from 2022-09-01 and after (two days after PIP-22 activation). The period between these dates is the *transition period*, where the unstaking and compounding takes place. The data that we use is public, it comes from the Pocket Network Blockchain (PNB) and the Cherry Picker (CP) database. While the PNB data can be directly retrieved from the network, the CP historical data is not available, we track and keep our own records of this data.

4.1 Selecting a good chain to test

The traffic in the Pocket Network is characterized by its constant fluctuations [6] which make long term analysis difficult. Since we are trying to measure the effects of the node reduction we need to keep all other factors, such as relays growth or traffic migration as low as possible. To this end we analyzed the fluctuations of the network along the major 8 chains in terms of relay volumes. The first metric to analyze is the full network relay volume coefficient of variation of a chain:

$$\widehat{CV}_n = \frac{\sigma_{c,n}}{\overline{x}_{c,n}}, \tag{3}$$

where $\sigma_{c,n}$ is the standard deviation and $\overline{x}_{c,n}$ the mean of the blockchain c for the total *network* traffic. The larger this value is, the larger are the fluctuations of the blockchain network traffic over time. The second metric that we analyzed is the local variation coefficient. This is the mean coefficient of variation over each of the CP gateways, weighted by their individual traffic:

$$\widehat{CV}_l = \frac{\sum_{g \in G} \widehat{CV}_g R_g}{\sum_{g \in G} R_g} \tag{4}$$

where R_g is the amount of relays observed in gateway g and with:

$$\widehat{CV}_g = \frac{\sigma_{c,g}}{\overline{x}_{c,g}}, \tag{5}$$

where $\sigma_{c,g}$ is the standard deviation and $\overline{x}_{c,g}$ the mean of the blockchain c for the gateway g.

The results of calculating the $\widehat{CV}_n$ and $\widehat{CV}_g$ for each of the main blockchains are presented in table 3.

Table 3: Network variation coefficients and total traffic for the main Pocket Network blockchains in the period from 2022-06-28 to 2022-09-12

Blockchain	Avg. Relays 24Hs	$\widehat{CV}_n$	$\widehat{CV}_l$
0009	230×10^6	0.095	0.32
0049	31×10^6	0.17	0.39
0027	98×10^6	0.22	0.51
0040	212×10^6	0.32	0.47
0005	34×10^6	0.36	0.57
0004	78×10^6	0.48	0.87
0021	134×10^6	0.56	1.10
03DF	30×10^6	0.57	0.67

The most stable blockchain can be easily determined by inspecting a scatter plot of each blockchain $\widehat{CV}_n$ vs $\widehat{CV}_g$, as presented in figure 1. It can be clearly seen from the figure that the 0009 chain (Polygon Mainnet) is the most stable, followed closely by 0049 (Fantom), however this last one's traffic is much lower ($230M$ daily relays vs $31M$ daily relays, see table 3). For this reason we chose to focus on bockchain 0009 for our main calculations.

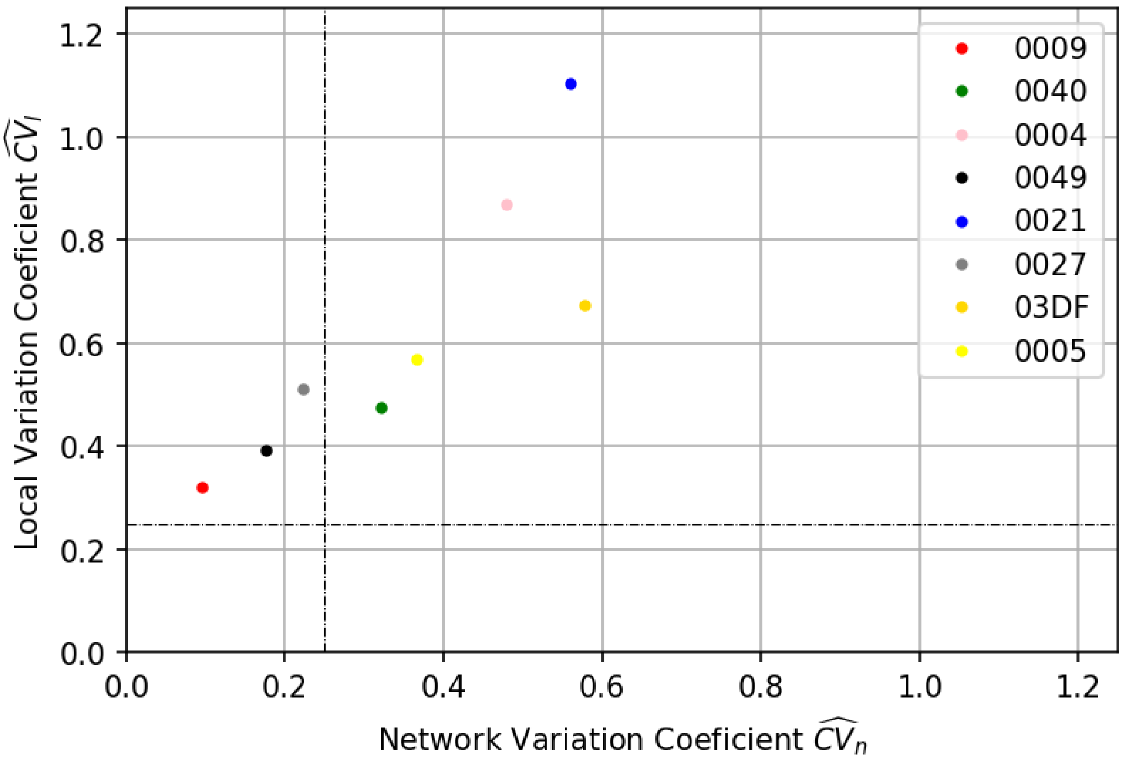

Figure 1: Scatter plot of the Pocket Network main blockchains using the network and local variation coefficients. The most stable networks are those near the $[0, 0]$ coordinate. As reference a pair of dashed lines are placed at 0.25 of each coefficient. The most stable chain is the red dot, the blockchain 0009.

4.2 Evolution of the number sessions and relays by node

The two main parameters affecting the PIP-22 base node rewards are the number of sessions and the number of relays that a node works. These two parameters are directly affected by the number of nodes in the network and understanding how they have evolved is very important to asses the fairness and effectiveness of the PIP-22 linear model. For ease of notation we will always refer to values in 24HS periods, unless stated otherwise.

Mathematical Definition

The number of sessions that a node expects serve is directly proportional to the number of nodes that are currently staked for a given blockchain. The probability of being selected for a given chain is simply [3]:

$$S_c \approx \frac{\text{NBS}}{N_c}, \tag{6}$$

where N_c is the number of nodes staked for blockchain c, $NBS = 24$ is the number of *nodes by session*. Given the current network parameters there are 24 sessions in 24HS,

[3] This is not entirely correct since the selection is without replacement, but given a large N_c this deviation is minimal.

roughly a session per hour (4 blocks of 15 minutes per session), for each staked app. Then, the amount of sessions that a single node expects to see over a 24HS period is calculated following a binomial distribution:

$$\overline{S_c} \approx 24\, A_c\, S_c, \tag{7}$$

where A_c is the number of *staked apps* requiring a session on chain c [4]. It can be seen that if the number of nodes N_c goes down, the session selection probability S_c increases and hence the number of expected sessions by node $\overline{S_c}$ also increases.

The number of relays by node is tightly coupled with the number of sessions by node $\overline{S_c}$, however their connection is not evident. The number of relays by node follow an stochastic process $\{CP(s) \,:\, s \in S\}$, with S being all possible combinations of sessions and governed by the Cherry Picker [5]. The discussion in the linear implementation of PIP-22 implies that this stochastic process is somewhat linear with $\overline{S_c}$:

$$\overline{R_c} \approx L\overline{S_c}, \tag{8}$$

where the function L is a constant value that converts the number of expected sessions $\overline{S_c}$ in the number of expected relays $\overline{R_c}$.

Empirical Definition

The average number of sessions that a node serve in 24 HS can be calculated empirically for a group of nodes in a given time period as:

$$\overline{S_c} = \frac{\sum_{b=1}^{B} Ws_{b,c}}{B\, N}, \tag{9}$$

where $Ws_{b,c}$ is the total number of worked sessions for block b in chain c, B is the number of blocks to use as average (≈ 96 for a day) and N the number of nodes in the calculation group. The larger the number of nodes N the better the average accuracy.

The number of relays by session that a group of nodes expects to see in 24 HS is calculated empirically as:

$$\overline{R_c} = \frac{\sum_{b=1}^{B} Wr_{b,c}}{B\, N}, \tag{10}$$

where $Wr_{b,c}$ is the total number of worked relays for block b in chain c, B is the number of blocks to use as average and N the number of nodes in the calculation group. In this case, the number of nodes in the group should be as similar as possible in terms of QoS in order to obtain a meaningful value. Due to the large variations in the number of relays, a more stable version of this mean is the worked relay proportion:

$$\overline{Rp_c} = \frac{\sum_{b=1}^{B} Wr_{b,c}}{\left(\sum_{b=1}^{B} R_{b,c}\right) N}, \tag{11}$$

where R_c is the number of total relays in chain c at block b.

[4] The variable A_c is also an random variable, for sake of simplicity we will consider a constant number of apps asking for sessions.

[5] Going deeper in this subject will make the reading complex, we will leave the CP modeling discussion aside.

Measured Values

The previously presented metrics ($\overline{S_c}$ and $\overline{Rp_c}$) were calculated for each day from 2022-06-28 to 2022-09-12 using public block information for 13 node runners with a large amount of nodes. The evolution of the average sessions by node $\overline{S_c}$ for chain $c = 0009$ can be seen in figure 2.

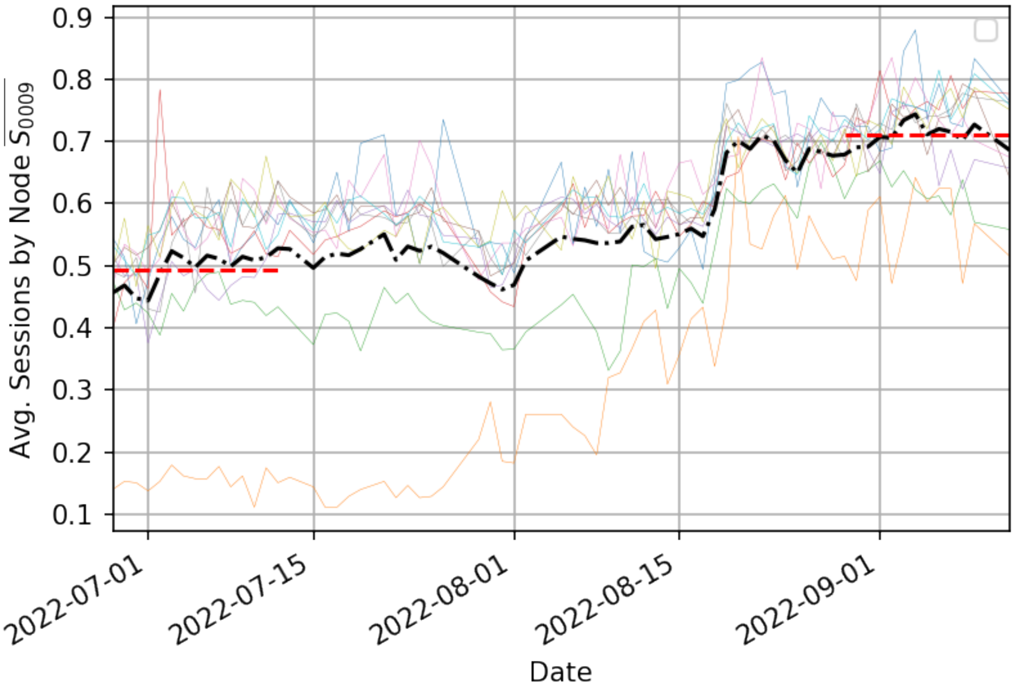

Figure 2: Evolution of the average number of sessions that a node serve in 24 HS ($\overline{S_c}$) for the blockchain 0009 as a function of the date. The thin lines represent different node providers, the black line represents the average. The red segments represent the means at the start and end of the period respectively. The average at the beginning is 0.49 and at the end 0.71 sessions by node.

The evolution of the worked relays proportion ($\overline{Rp_{0009}}$) for the same period can be seen in figure 3. In this case the graph is much more spread.

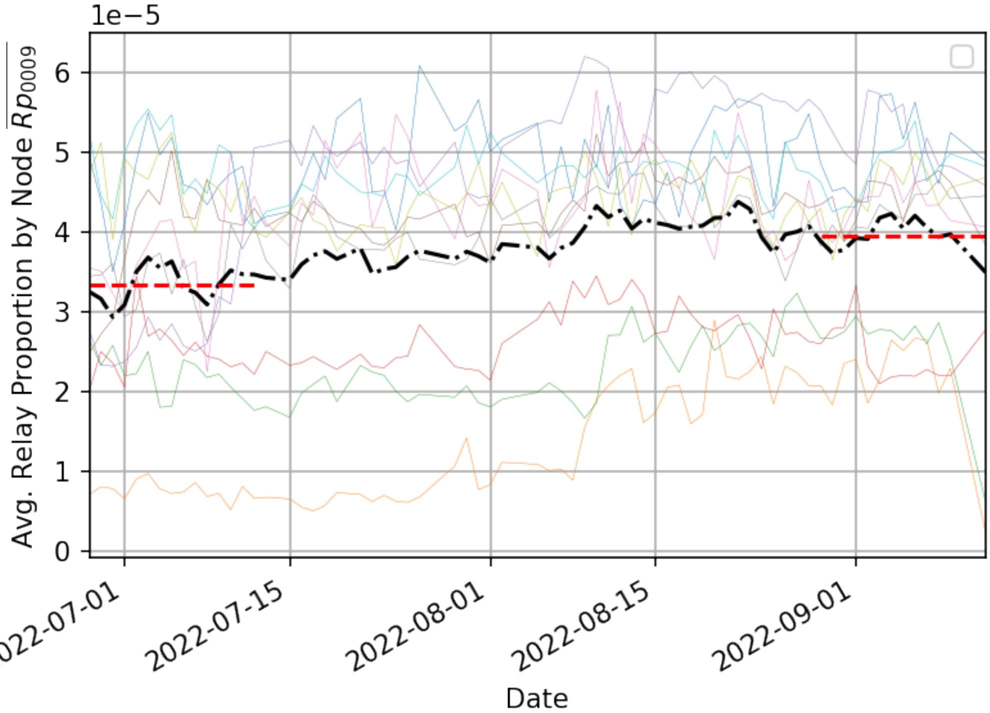

Figure 3: Evolution of the average relay proportion that a node work in 24 HS ($\overline{Rp_c}$) for the blockchain 0009 as a function of the date. The thin lines represent different node providers, the black line represents the average. The red segments represent the means at the start and end of the period respectively. The average at the beginning is 3.34×10^{-5} and at the end 3.95×10^{-5}.

These values were averaged using a period of two weeks at the beginning and at the end of the observed period (red lines shown in figures 2 and 3). The results are shown in table 4. The relation between these increments is calculated as:

$$Vr_c = \frac{variation(\overline{Rp_{0009}}^{ini}, \overline{Rp_{0009}}^{end})}{variation(\overline{S_{0009}}^{ini}, \overline{S_{0009}}^{end})} = \frac{0.18}{0.44} = 0.41 \tag{12}$$

Linear weighting expects $Vr_c \approx 1$ to justify the multiplier of base stake nodes. A $Vr_c < 1$ indicates that the growth in the number of sessions by node is larger than the the growth of relays proportion by node. It can be seen that relation is far from the expected $Vr_c = 1$, the value $Vr_c << 1$ means that the nodes in the first bin (with 15×10^3POKT staked) are seeing their rewards reduced.

Table 4: Changes in the number of sessions by node and relays proportion worked by node prior and post PIP-22.

	Prior	Post	change
Avg. Sessions ($\overline{S_{0009}}$)	0.49	0.71	44%
Avg. Worked Relays Prop. ($\overline{Rp_{0009}}$)	3.34×10^{-5}	3.95×10^{-5}	18%

We proceed to analyze if the factor Vr_c of equation 12 is affected differently by the QoS of the node runner. For each of the tested node runners this value was derived following the same steps. The QoS of the node runners was obtained using the CP data. The average QoS is obtained as an average of the median response time in the different cherry picker gateways, weighted by the gateway number of relays. The results are shown in figure 4. Given the low amount of statistics (13 points, one per provider), no evident relation is found between the QoS and the variation value Vr_c.

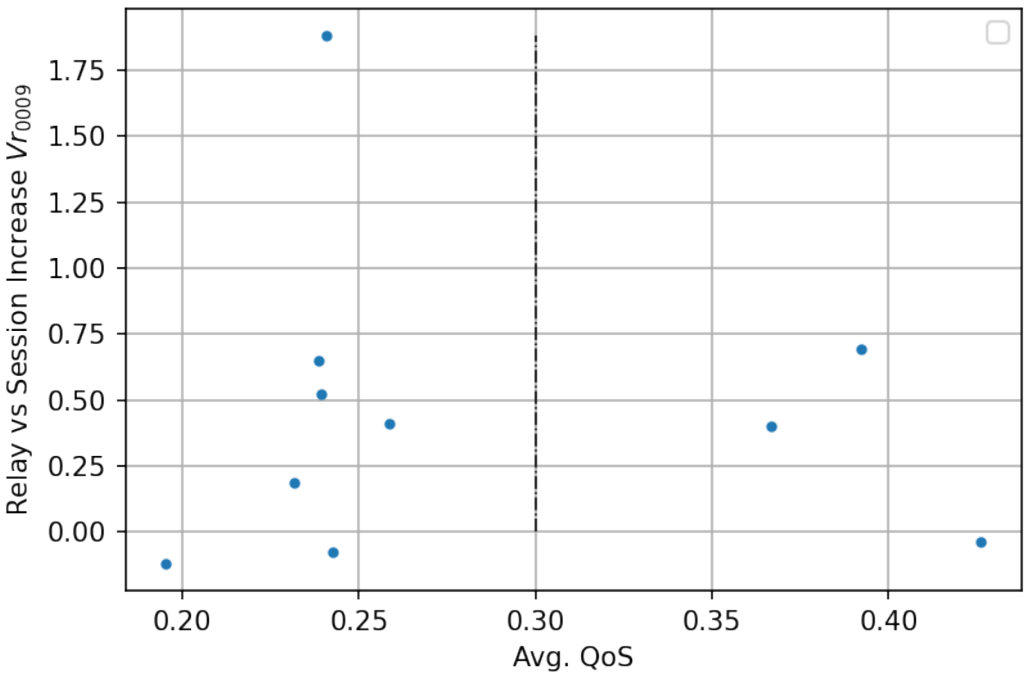

Figure 4: Values of Vr_c in chain $c = 0009$ for each of the 13 tested providers. The vertical dashed black line represents an empirical threshold value for the QoS.

4.3 Analysis of correlation between number of sessions and relays by node

An other way to observe if there is a linear relationship between the number of sessions and the number of relays ($\overline{S_c}$ and $\overline{Rp_c}$) is to measure the coefficient of determination D (the squared correlation coefficient) using two samples from each group. This coefficient indicates how much variability can be explained from $\overline{Rp_c}$ by assuming a linear relationship with $\overline{S_c}$ (see equation 8). If the number of sessions determinate significantly the number of relays served we would expect $D \approx 1$. On the other hand if $D \approx 0$ it means that we cannot predict $\overline{Rp_c}$ using $\overline{S_c}$ and a linear model.

The coefficient of variation was computed using the data of each studied node runner in the 69 days of history. Thus we calculated 13 values of D, one per provider, and each D was calculated using 69 data points. Each of these points is shown as a function of the provider avg. QoS in figure 5. In this figure a relation seems to exist, if we use a threshold avg. $QoS = 0.3s$ to divide the node runners in two tiers the difference is clearer. Faster nodes will be tier A and slower ones tier B. This is shown as boxplots in figure 6. The values obtained for each group are presented in table 5. It can be seen from the values in the table that the determination coefficient is generally low, indicating that there is not an strong linear relation between $\overline{Rp_c}$ and $\overline{S_c}$. This low determination values indicate that using only $\overline{S_c}$ as predictor of $\overline{Rp_c}$ is not justified for a linear model. Also when the nodes are segmented into tiers it can be seen that the determination coefficient is lower for faster nodes (Tier A nodes). This difference indicates that there is some mechanic of the CP that is affecting the increments of relays in each of the observed tiers [6].

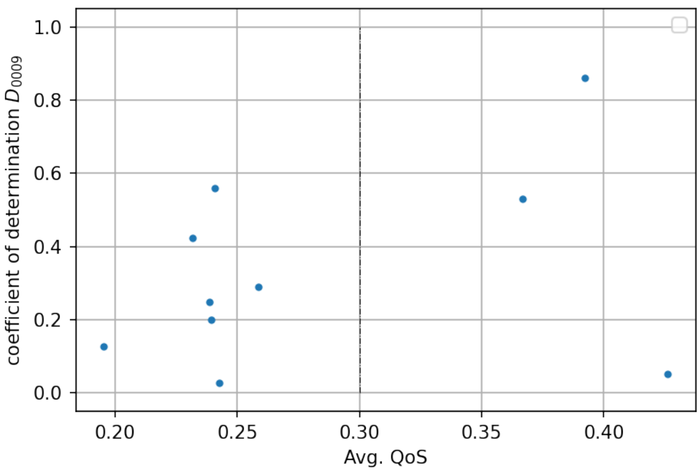

Figure 5: Coefficients of Determination D_c in chain $c = 0009$ for each of the 13 tested providers. The vertical dashed black line represents an empirical threshold value for the QoS.

[6] It is worth noting that we are comparing relative values of increase, not absolute values, we beliebe that the CP is working as expected.

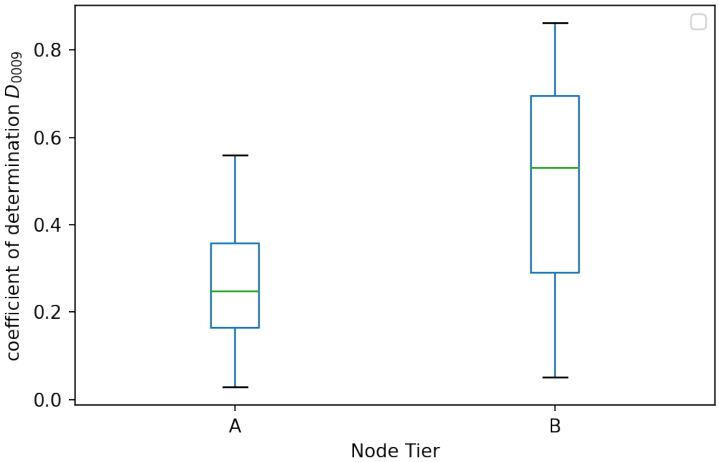

Figure 6: Coefficients of Determination D_c in chain $c = 0009$ for the tested providers grouped using the avg. QoS threshold of $QoS = 0.3s$. Faster nodes are in Tier A, slower ones in Tier B.

Table 5: Determination coefficients statistics for each group of node providers. Faster nodes are in Tier A, slower ones in Tier B.

Tier	**mean** D_{0009}	**std** D_{0009}
A	0.27	0.18
B	0.48	0.41

4.4 Effects of additional corrections due to unexpected over-minting

One final issue of the PIP-22 linear model that we want to address is the problem with the calculation of the SSWM when the compounding nodes do not have an average relays by node $\overline{R_c}$ similar to the network. The line 15 of algorithm 2 can be interpreted as a weighted *average bin* of the network. For clarity we will reproduce this expression here:

$$\text{Avg. Bin} = \left(\frac{\text{Base Nodes}}{\text{Num. Nodes}} \right). \tag{13}$$

This interpretation can only be correct if the nodes composing the "Num. Nodes" variable have an efficiency ($\overline{R_c}$) similar to the network efficiency. If all the nodes compounding have a high efficiency their minting will be higher than expected. To solve this problem in the SSWM should be increased even further. We wont analyze this any further, but an increase of SSWM beyond the value indicated by algorithm 2 will not correspond to any session by node increase. Currently this situation is happening and an over-minting exists. The table 6 shows the averages of the nodes in each compounded bin. It can be seen that the node runners at outside the base bin have an average $\sim 10\%$ higher than the network average.

Table 6: Avg. Relays by node ($\overline{R}$) for each compounded bin and the network average in the Pocket Network (2022-09-21).

Bin	Nodes	$\overline{R}$
15 K POKT	31 K	21685
30 K POKT	40 K	463
45 K POKT	39 K	992
60 K POKT	41 K	4374
Network	37 K	27514

4.5 Relation of Base Stake Nodes Multipliers and the Number of Expected Sessions

With the previous definition of equation 7 we can show the relation of between the multiplier assigned to nodes in the base stake level and the increment of the expected sessions by node. By applying the algorithm 1 for nodes with Node Stake $=$ SSFM, we will obtain:

$$W_{base} = \frac{1}{\text{SSWM}} \tag{14}$$

From algorithm 2 we know that:

$$\text{SSWM} \approx \frac{\text{Base Nodes}}{\text{Num. Nodes}}, \tag{15}$$

where Base Nodes is the number of base stake nodes that can exist given the current staked POKT in the network. Then:

$$W_{base} \approx \frac{\text{Num. Nodes}}{\text{Base Nodes}}. \tag{16}$$

Now this relation is the same as the relation of the expected number of sessions by node $\overline{S_c}$:

$$W_{base} \approx \frac{\overline{S_c}^{base}}{\overline{S_c}^{staked}} = \frac{24\, A_c\, S_c^{base}}{24\, A_c\, S_c^{staked}}, \tag{17}$$

we consider that the number of applications (A_c) does not change, and replace S_c for its definition in equation 6:

$$W_{base} \approx \frac{S_c^{base}}{S_c^{staked}} = \frac{\frac{\text{NBS}}{N_c^{base}}}{\frac{\text{NBS}}{N_c^{staked}}} = \frac{N_c^{staked}}{N_c^{base}}, \tag{18}$$

where N_c is the number of staked nodes on chain given chain c. Changing the notation from algorithm 2 to the one used in equation 6 the relation is evident:

$$W_{base} \approx \frac{\text{Num. Nodes}}{\text{Base Nodes}} \approx \frac{N_c^{staked}}{N_c^{base}}. \tag{19}$$

5 Proposed Non-Linear Stake Weighting

In the previous section we showed that the linear stake weighting model is penalizing the base stake nodes. This penalization comes from assigning the base nodes a reduction of their rewards proportional to the inverse of the increment of the sessions by node. We showed that using the expected sessions by node as a proxy of the relays by node is not valid. Moreover we showed using Pocket Network blockchain data that the increment of relays is lower than the expected, while we observed an increment of 44% in sessions by node we only observed a 18% increase in the number of worked relays (see table 4). To remedy this unfair reduction of the income of base stake nodes the W_{15} K must be increased, but increasing this parameter in the linear staking model will create an over-minting that interferes with the inflation control. The stake weighting should not interfere with the tokenomics objectives of the network. The only solution to the base stake unfairness, given the rules of PIP-22, is to transition to a non-linear stake weighting model. In this model the minting will kept constant for base stake nodes and the gains of the higher stake nodes will be modified to keep minting constant. In this new scenario the more POKT that is staked in a single node, the lower the return per POKT will be. However, the costs per node will remain constant. Finding a *sweet spot* of POKT return per POKT invested will not be a simple strategy of going to the largest bin and it wont be either stay in the base stake. We wont analyze the economic view of the non-linear stake weight, since the PIP-22 was not focused on the economics either.

5.1 Parameter Calculation

It is difficult to define a precise calculation of the PIP-22 parameters given the complexity of the Pocket Network. We suggest to remain optimistic and keep using the expected sessions per node as the main feature for estimating the relays per node, however we propose to make up for the observed bias. As it is shown in table 4 and in our previous report [4], the differences observed are between 25% and 35%, this means that given an increase of 20% in the number of sessions results in an increase of 10% to 15% in the number of relays.

The calculation of the new parameters should be done as presented in algorithm 3.

If the minting needs to be reduced from the value given by algorithm 2 (as expected from our observations in section 4.4), the line 13 of the proposed algorithm should be changed, not the base multiplier calculated in line 4. This means that over-minting should be discounted from compounded nodes, not from base stake nodes that should not be affected.

Algorithm 3 Calculate ServicerStakeWeightMultiplier and ServicerStakeFloorMultiplierExponent for Non-Linear weighting

procedure Calculate SSWM and SSFME(k_{fair})
    SSWM$_{lin}$ $\leftarrow$ get linear SSWM using algorithm 2
    $W_{base} \leftarrow \frac{1}{\text{SSWM}_{lin}}$
    $W_{base} = (1 - W_{base})k_{fair} + W_{base}$
    SSWM $\leftarrow \frac{1}{W_{base}}$ using algorithm 2
    Num. Nodes $\leftarrow$ get number of staked nodes
    SSFME $\leftarrow 1.0$
    lowest error $\leftarrow 9e99$
    search space $\leftarrow$ space$(ini = 1, end = 0, step = -0.001)$
    for SSFME$_{iter}$ $\in$ search space **do**
        $mults \leftarrow$ algorithm 1 (SSWM , SSFME$_{iter}$)
        nbb $\leftarrow$ get nodes staked by bin
        current error $\leftarrow abs\left(1 - \frac{\sum_{bin=1}^{BINS}(\text{nbb}[bin] \cdot mults[bin])}{\text{Num. Nodes}}\right)$
        if current error $\leq$ lowest error **then**
            SSFME $\leftarrow$ SSFME$_{iter}$
            lowest error $\leftarrow$ current error
        end if
    end for
    Return SSWM, SSFME.
end procedure

5.2 Changes by Stake Level against Linear Staking

The proposed weighting model change will affect how rewards are distributed. Under current the current SSWM $= 1.4687$ the expected changes for a $k_{fair} = 0.35$ and $k_{fair} = 0.25$ are shown in table 7 and table 8 respectively.

Table 7: Stake weighting multipliers comparison between linear model (SSWM $= 1.4687$ and SSFME $= 1$) and non-linear model with $k_{fair} = 0.35$ (SSWM $= 1.2763$ and SSFME $= 0.705$)

Stake	Linear Mult.	Non-Linear Mult.	Change
15 kPOKT	0.6809	0.7835	15.72%
30 kPOKT	1.3617	1.2728	-6.53%
45 kPOKT	2.0426	1.6905	-17.23%
60 kPOKT	2.7235	2.0677	-24.08%

Table 8: Stake weighting multipliers comparison between linear model (SSWM $= 1.4687$ and SSFME $= 1$) and non-linear model with $k_{fair} = 0.25$ (SSWM $= 1.3330$ and SSFME $= 0.795$)

Stake	Linear Mult.	Non-Linear Mult.	Change
15 kPOKT	0.6809	0.7502	10.18%
30 kPOKT	1.3617	1.2971	-4.74%
45 kPOKT	2.0426	1.7869	-12.52%
60 kPOKT	2.7235	2.2428	-17.65%

The non-linear effect is more clearly shown graphically. In figure 7 the multiplier values of the current linear model and the proposed model are plotted against the stake value, along the *imaginary* continual stake evolution lines for reference.

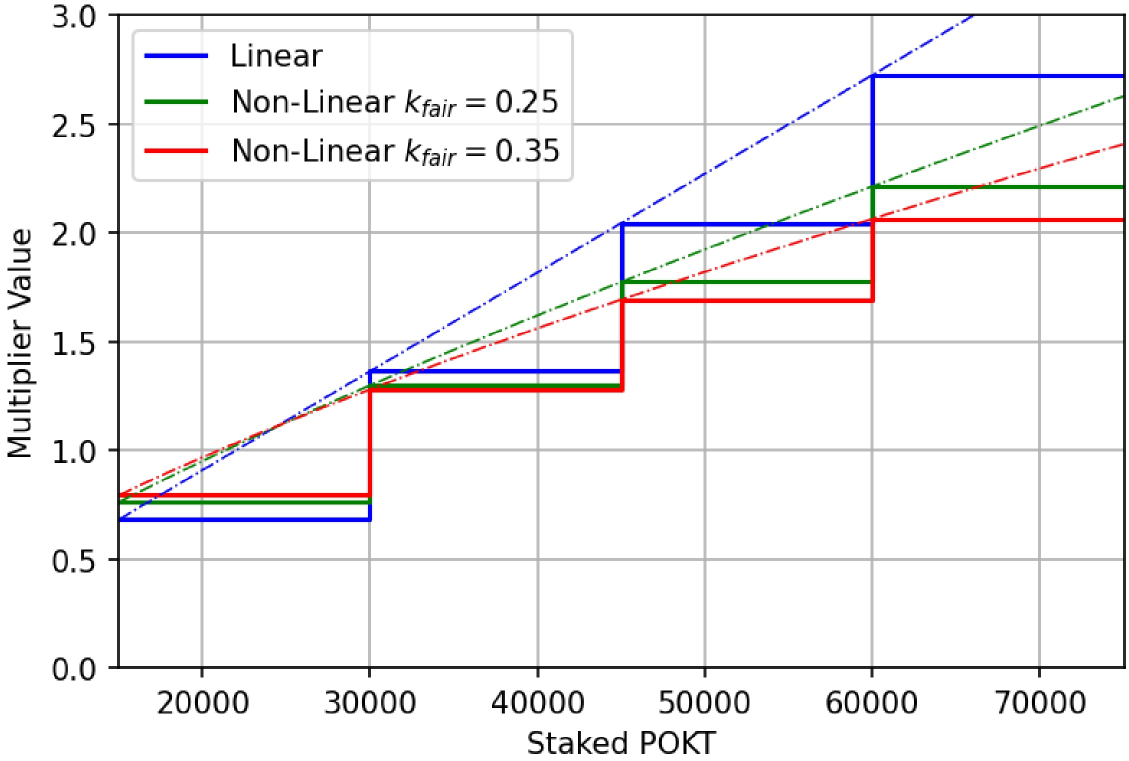

Figure 7: Cahnges in multiplier scales for the current linear model (blue line) and the proposed non-linear model with $k_{fair} = 0.25$ (green line) and $k_{fair} = 0.35$ (red line). The thin dashed lines show the changes in the stake weighting from linear to non-linear.

5.3 Changes in Stake Strategy

By using a non-linear stake weighting the best strategy is not simply going to the largest bin. We will not dive into economical models since there is a large amount of business models and the objective of this document is not economical.

Suppose that there are two different node runners, node runner A and node runner B. Both providers have the same hardware costs around 85% of the income of a an average node in the base bin and the same number of tokens to stake:

- Node Runner A:
  - POKT to Stake: 180×10^3POKT
  - Node Hardware Cost: $19 \frac{\text{POKT}}{\text{day}}$
  - Node Performance : $32 \times 10^3 \frac{\text{Relays}}{\text{day}}$
- Node Runner B:
  - POKT to Stake: 180×10^3POKT
  - Node Hardware Cost: $19 \frac{\text{POKT}}{\text{day}}$
  - Node Performance : $48 \times 10^3 \frac{\text{Relays}}{\text{day}}$

These node runners will try to maximize their POKT income, depending on the number of nodes and stake weight their income will be different. Having 180×10^3POKT to stake, each node runner can create either 12 nodes in base stake, 6 in the second bin, 4 in the third bin or 3 at the top bin. For each of these possible staking strategies and for each of the staking models (linear or non-linear) the number of POKT obtained by day is found in table 9. The values were calculated using equation 2 and discounting the total hardware cost. It can be seen that in the linear stake weight model the strategy is to go for the highest stake bin, however using a non-linear stake model the node runner B will find the best return at the second bin. This is important since the node runner B has a higher QoS than the node runner A, this means that the network in this example will have 3 low QoS nodes and 6 high QoS nodes. In the linear stake weighting scenario, the network will have 3 low QoS nodes and 3 high QoS nodes. The non-linear model, in this scenario, will have more high QoS nodes than low QoS nodes.

Table 9: Total POKT minted, in each stake level, for node runner A and B using linear stake model (SSWM $= 1.4687$ and SSFME $= 1$) and non-linear model with $k_{fair} = 0.35$ (SSWM $= 1.2617$ and SSFME $= 0.695$).

		Node Runner A		**Node Runner B**	
Nodes	**Stake**	Linear	Non-Linear	Linear	Non-Linear
12	15×10^3	33.9	73.4	147	206
6	30×10^3	129	120	243	227
4	45×10^3	616	126	275	222
3	60×10^3	177	127	291	215

6 Conclusions and Discussion

In this document we have exposed the issues of fairness in base stake nodes, proving that there is no justification for the current reduction of minting that is being enforced. We have also shown that the current linear weighting strategy is no viable in the long term since it reduces to an increase in the base stake weight which was not the objective of PIP-22.

More specifically, we have outlined the current PIP-22 mechanism and analyzed the linear weighting implementation. Using historical data from 2022-06-28 to 2022-09-12 we have studied the variations in the expected number of sessions by node and the variations of the relays by node. We used this data to support our previous models [4] and expose the limitations of the linear model in terms of fairness and its ultimate goals. We conclude that:

- The linear weighting model penalizes the nodes that do not perform compounding. This happens since an increase in the number of sessions do not translates in the same increment in the number of served relays. We estimate that non-compounded nodes are loosing $\approx 15\%$ of their rewards.
- The linear staking model is not sustainable, since the only valid strategy is to compound at maximum stake. When the majority of the network reaches this bin the stake weight will play no role. This will only create a de-facto increase of the minimum viable stake. This was not the purpose of the PIP-22.
- Observing only the variations of the number of sessions by node does not gives enough information on how the number of relays was modified. This means that any model that uses the increment of the number of sessions to predict an increase in the number of relays served is not correct. The linear weighting does this, hence the reduction of the rewards for the base stake nodes is not correctly justified.
- The calculation of the SSWM expects the performance of the nodes in each bin (measured in relays) to be similar. If this is not true the minting wont be controlled as expected, leading to under- or over-minting. Under current conditions, with high-performing nodes in the highest bins, the network is minting more POKT than expected. Correcting this under the linear model will create a further reduction of the rewards of non compounding nodes, this time it will be completely unjustified.

For this reasons we propose to implement the non-linear weight model included in PIP-22. In other words, set the $ServicerStakeFloorMultiplierExponent$ to a value lower than 1.0 (ranging from 0.7 to 0.8). This change will correct the unfair reduction of the gains observed in nodes at base stake. These values will increase the minting of base stake nodes by $\approx 16\%$ while reducing the minting of the max staked nodes by $\approx 24\%$. Also, implementing a non-linear model will enable the stake weighting to avoid reducing the minting of base nodes due to inhomogeneities in the performance of the nodes at different stake levels. Adjusting the $ServicerStakeFloorMultiplierExponent$ instead of the $ServicerStakeFloorMultiplier$ is the correct way (in terms of fairness) for eliminating the over-minting. Finally the non-linear weighting creates staking scenarios where the best strategy is not to stake at the highest bin. In this scenarios the high QoS can find an optimal stake level lower than low

QoS nodes, which can result in an improvement of the overall network QoS.

Appendix

In the appendix section the main graphs observed in section 4 are reproduced for other chains. Please remember that all these other chains are less stable than chain 0009 (see section 4.1).

In order to keep the document as succinct as possible the appendixes are published separately.

Reference List

[1] Adam. *Demystifying the Cherry Picker*. 2021. url: https://www.blog.pokt.network/demystifying-the-cherry-picker/?hss_channel=tw-789201646685392896 (visited on 09/01/2022).

[2] Adam. *PUP-11: WAGMI Inflation*. 2021. url: https://forum.pokt.network/t/pup-11-wagmi-inflation/1369 (visited on 12/01/2021).

[3] Andy-Liquify. *PIP-22: Stake-weighted Servicer Rewards*. 2022. url: https://forum.pokt.network/t/pip-22-stake-weighted-servicer-rewards/3101 (visited on 09/01/2022).

[4] Ramiro Rodríguez Colmeiro. *Simulation of a plausible post-PIP-22 network landscape and the effects of the PUP-21 values on the node rewards.* 2022. url: https://github.com/pokt-scan/poktscan-pip-22-simulations (visited on 07/29/2022).

[5] msa6867. *PUP-21 Setting parameter values for the PIP-22 new parameter set*. 2022. url: https://forum.pokt.network/t/pup-21-setting-parameter-values-for-the-pip-22-new-parameter-set/3200 (visited on 09/01/2022).

[6] Ramiro Rodríguez Colmeiro Pablo Frigerio. *Analysis and Forecasting of the Main Pocket Network Chains Traffic Using Statistical Tools.* 2022. url: https://github.com/pokt-scan/infracon-2022 (visited on 05/01/2022).

[7] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.