Additional insights from web3 reward emission research

We are currently undertaking a large-scale research effort to create better frameworks for token mechanics especially for Web3 protocols in the hardware resource provisioning sector.

We have recently published part one, where we have compared emission schedules of various Web3 networks, including of course Render. You can find the report here.

Even though Render hasn't started the emissions voted on with RNP001, we wanted to share some additional data with the Render community considering the planned emissions of RNP001 that are not in the report.

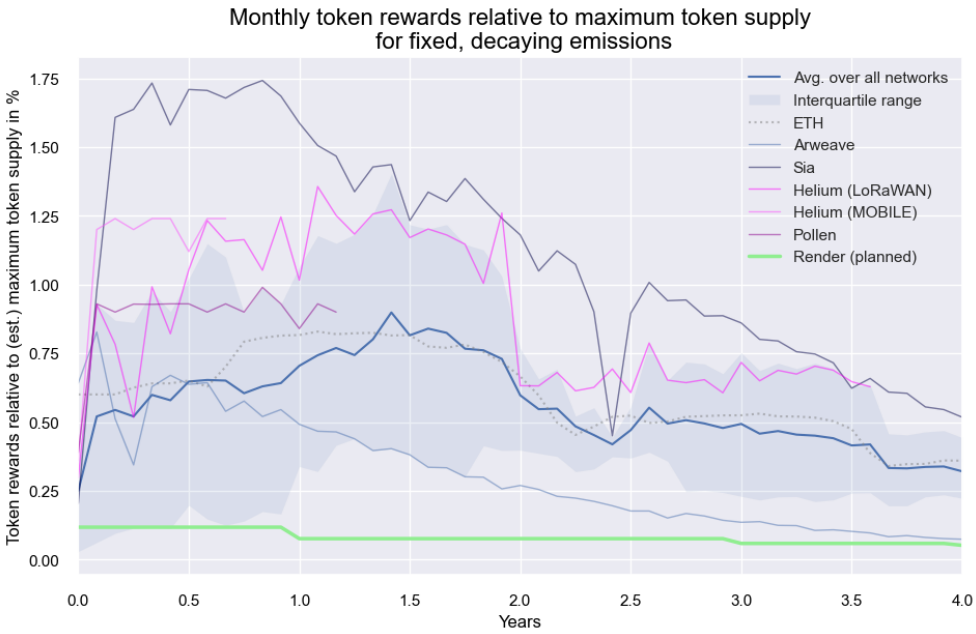
First off, we have classified Render's token rewards to be in the category of fixed and decaying emissions: despite being constant over some periods (1st year, year 2-3) they are decreasing (hence decaying) as a function solely depending on time (hence fixed).

Within that category there are currently also the following projects we analyzed:

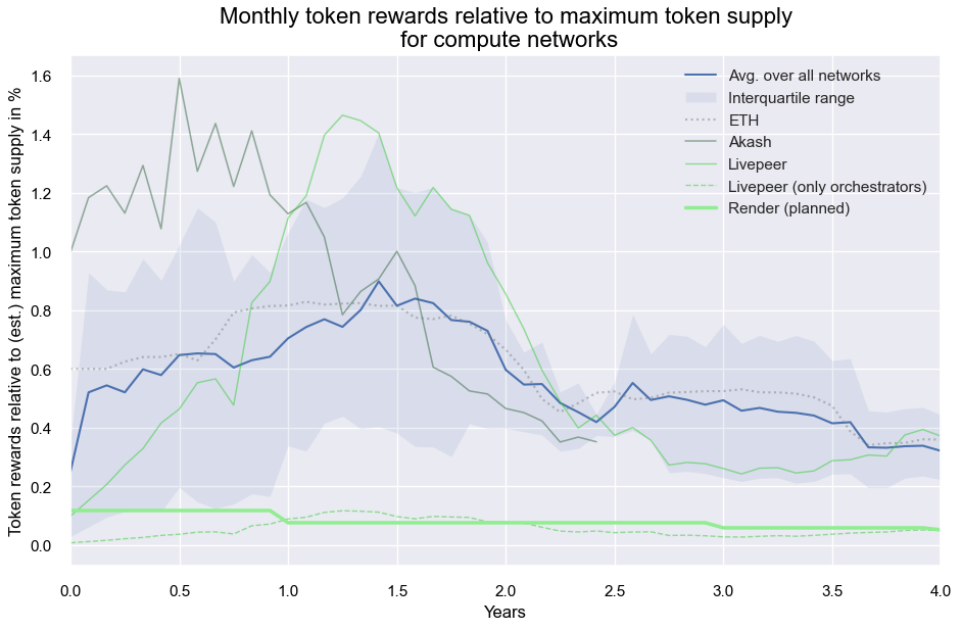
- Arweave
- Helium (LoRaWAN)
- Helium (MOBILE)
- Sia
- Pollen

In comparison to the other projects in this category, Renders planned emissions are the lowest, which of course is related to the fact that the network is already up and running for a while with an existing token holder base (so hence, limited comparability).

Below chart shows the average (blue line) and the interquartile range (blue shaded) of the monthly token rewards of all Web3 infrastructure networks together with the projects of the fixed and decaying emissions category highlighted:



For Render it might actually be more relevant to compare it to the networks providing similar services versus comparing it to projects with similar token emissions. So here we go:



Putting the limited comparability aside, it is interesting to see that Render's (planned) relative token emissions are on the same level as the rewards orchestrators got as reward share for providing services on the Livepeer network. More information around caveats like the mentioned comparability and the data are available in this repo that also contains the paper with further details and some links that might be interesting as well, e.g. on recent work around Burn and Mint equilibrium models.

We hope this information yields some insightful information for the Render community and we are open to your feedback. What would be some additional aspects/comparisons you would like to see?