Cosmos Multi-Token Proof of Stake Token Model

Sankalp Aggarwal
Tendermint, Berkeley, CA, USA
http://www.sunnya97.com
sunny@tendermint.com

Abstract

This paper presents a novel token model for Proof of Stake based blockchains being pioneered by the Cosmos Hub. We argue that general purpose currencies should not be used as staking tokens, but rather we should use tokens whose primary utility is to be used for staking. We also present mechanisms for allowing the blockchain's transactions fees to be paid in a multitude of fee tokens, which will improve the user experience of using the blockchain.

2012 ACM Subject Classification Theory of computation-Algorithmic mechanism design; Law, social and behavioral sciences-Economics

Keywords and phrases Proof of Stake, Staking Tokens, Cosmos, Tendermint

Digital Object Identifier 10.4230/LIPIcs..2019.

1 Introduction

Proof of Stake refers to the concept of using ownership of tokens in a blockchain as a sybil resistance mechanism for determining participation in its consensus protocol. The Cosmos Hub is a Proof of Stake based blockchain that uses Tendermint consensus and serves as a backbone to the Cosmos ecosystem. The Cosmos Hub provides a number of innovations to the field of Proof of Stake and cryptoeconomic design, including features such as skin-in-the-game delegation, instant redelegation, delegation commitments, and efficient automatic reward distribution. [1] This paper, however, will focus particularly on the Cosmos Hub's token model for Proof of Stake systems, which differs from most existing Proof of Stake proposals. While this model being presented was designed for the Cosmos Hub, it is meant to serve as an example for other Proof of Stake blockchains, both inside and outside of the Cosmos ecosystem to emulate.

Tendermint Consensus

The Cosmos Hub uses Tendermint as its consensus algorithm.[12] Tendermint is a byzantine fault tolerant consensus algorithm developed by Jae Kwon in 2014 to address the speed, scalability, and environmental concerns of Proof of Work. [10] The economics of Cosmos Proof of Stake are designed with a classical BFT algorithm (such as Tendermint) in mind due to its design of involving all validators in the production of each block, a substantial difference from Nakamoto consensus. [16] Unlike PBFT, from which it is heavily inspired, Tendermint also has the ability to seemlessly change the leader on a per-block basis. This means validators in Tendermint take turns producing blocks in a weighted round robin fashion. Furthermore, its strict accountability for byzantine faults allows us to punish misbehaving validators and provide economic security for the network. The safety and liveness guarantees of Tendermint hold under that assumption that less than 1/3 of voting power is byzantine. Description of Tendermint consensus algorithm and the proof of its correctness can be found in [4].

The Cosmos Hub uses Tendermint in its public Proof of Stake context. This works in the following method as described by the original Tendermint whitepaper [10]:

Tokenomics, International Conference on Blockchain Economics, Security and Protocols.
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

> "Validators are users with accounts that have coins locked in a bond deposit by posting a bond transaction. We say that a validator has voting power equal to the amount of the bonded coins."

In such a system, the validator set is open and permissionless, which means that anyone who owns some of the *staking token* in the system can bond their coins and become a validator. In the Cosmos Hub, this staking token is known as an Atom. The limited resource of Atoms acts as a sybil prevention mechanism. A single actor cannot create multiple validator nodes in order to increase their voting power, as their voting power is determined by their total number of atoms owned, not the number of validator nodes they control. Because all that determines a validator's voting power is their bonded stake and not reputation or real world identity, validators can choose to be either anonymous or public.

One of the key innovations of the Tendermint whitepaper was the idea of using validator "security deposits" which can be seized and burned by the protocol in a process known as "slashing" if a validator is caught creating attributable byzantine faults that harm to the well-functioning of the system, thus solving the Nothing-at-Stake problem suffered by first generation Proof of Stake blockchains like NXT and BitShares 1.0 [2] [14]. These security deposits are locked in a bonded account and when a staker wishes to unbond their staked tokens, they are still bonded for the period of an "unbonding period". During this unbonding period, the validator doesn't have any voting power, but proof of malicious activity from past byzantine faults can still be submitted, and the validator can still be slashed.

The details of the slashing conditions which lay out the possible byzantine faults and their respective punishments are out of scope of this paper.

Role in the Cosmos Ecosystem

The Cosmos Ecosystem is a partially-ordered network of blockchains called zones that run concurrently and interoperate using light client-style proofs to create a multi-blockchain ecosystem. It is termed an Internet of Blockchains as much like the traditional Internet which connects otherwise isolated networks, Cosmos connects otherwise isolated blockchain networks. These multiple blockchains can communicate using a protocol called Inter Blockchain Communication (IBC), which can be thought of as the TCP/IP of Cosmos. [8] IBC enables zones to create two-pegs between them and transfer coins between them, essentially making them sidechains of each other. Using IBC, you could move a native token of Zone A onto Zone B for use within the state machine of Zone B. Much like how Internet Service Providers (ISPs) help route IP packets, Hubs are special blockchains that help zones route IBC packets. [12]

The Cosmos Hub is one such hub to which zones can connect to and is secured by an open and globally decentralized set of validators. Along with just providing routing for packets, the Cosmos Hub also preserves the global invariance of the total amount of each token across the zones, preventing zones from double spending other zones. [12]

Many ISPs, along with providing Internet connectivity, also offer a variety of additional services such as web hosting. Similarly, the Cosmos Hub also provides hosted consensus. While zones can connect to the Cosmos Hub with their own independent validator sets, zones also have the opportunity to be validated by the Cosmos Hub validators and benefit from the shared security of all the chains operated by the Cosmos Hub validators. [11]

Because the Cosmos Hub is responsible for the operation and consensus of many chains, the security and cryptoeconomic design of the Hub is of paramount importance.

2 Token Economics and Incentives

Let us examine some of the fundamental economics and incentives of blockchain systems. To do so, let us first define the two primary classes of actors in a blockchain network, the *keepers* and the *users*. The term keepers refers to a class of actors that act as the operators and maintainers of a blockchain network. In the context of blockchain consensus, keepers refers to miners in a Proof of Work setting and stakers in a Proof of Stake setting [21]. The users of a blockchain network are the people who are deriving some utility from the network. In any public blockchain, the keepers of the chain are essentially offering the service of providing *BFT compute* to the users of the blockchain, and in order for them to do this, they need to be incentivized.

In order to explore incentivization, let us first set up some basic premises of the concept of value. Blockchain ecosystem tokens can be considered a type of commodity. To borrow from Marxian economic terminology, commodities have two identified types of value: use value and exchange value [15]. Use value is defined as the "want satisfying power" of a commodity. It is based on the utility of the commodity. Meanwhile, the exchange value of a commodity is defined as "the amount of goods and services which we may obtain in the market in exchange for a particular thing" [15]. We may commonly think of this as the price of the commodity.

We can define the economic security of a blockchain consensus system as the amount of money needed to be spent in order to attack the system, which is proportional to the exchange value of the commodity needed to be consumed in order to pull off the attack. In Proof of Work, the commodity needed to be spent is electricity, and thus the economic security is the number of joules needed to attack times the exchange value (price) of joules. Similarly, in Proof of Stake, the economic security is based on the number of staking tokens needed to have more than 33% of stake times the exchange value of the staking token.

For many different reasons, the exchange value of a commodity might be greater than the use value of the commodity. Trying to reason about and predict the exchange value of Atoms is very difficult, especially due to the lack of market data before the public launch of the Cosmos Hub. However, in theory, the exchange value should be effectively lower bounded by the use value of the commodity. Thus, we can use this lower bound in order to reason about the economic properties of our system.

In most Proof of Work blockchains, the miners are paid by the users through transaction fees and sometimes block rewards which need to be paid in some token that has exchange value. In these Proof of Work blockchains, there is usually a native token whose primary purpose is to pay transaction fees in the network. In some cases, this same token also has some other desired utility as well. For example, in Bitcoin, the token is used to pay transaction fees but also acts as a generic monetary currency. But at the very minimum, the use value of the token is its utility in affording you to use the blockchain due to its necessity in paying transaction fees. Because this token has a use value, this gives it a lower bounded exchange value, thus allowing it be used to compensate miners for their role in providing consensus.

3 Staking Tokens

Single Token Model

The naive approach to creating a Proof of Stake blockchain is to try and extend this single token model and use it for both fees and staking. Because the single token already has

some minimum exchange value enforced by its use value, it provides economic security to the Proof of Stake by making sure there is real economic value at stake. This is done by many Proof of Stake systems. For example, when switching to Casper, Ethereum plans to use its existing fee token, Ether, as its staking token. [17] While there is some merit to the conceptual simplicity of this model, it does lead to potential risks and issues.

In the single token model, the single token has utility beyond mere staking, thus there will be a reasonably liquid amount of tokens that are not staked. The more utility that the token has beyond staking, the more liquidity of that token is required for those purposes, meaning a larger percentage of the tokens will not be staked. This weakens the security of the system as it makes it easier to stealthily obtain enough tokens required to attack the system.

For example, in a Proof of Work network, because electricity and computational power have so many alternate utilities other than mining, the vast majority of these resources are not used for mining and thus are "unaccounted for". For a Proof of Work blockchain using an ASIC-resistant hashing, it would be possible for a large government to suddenly shift a lot of its electric and computational resources to mining to attack the system. However, because SHA-256 ASICs have little utility beyond Bitcoin mining (which is largely dominated by ASICs), it is reasonable to assume that close to all existing SHA-256 ASICs are already currently mining, and thus it is highly unlikely that any entity could suddenly start mining and have over 50% of the hashrate, which is needed to take over the system [19]. Any attempt to try to mass produce or buy up a majority of ASICs would take time and would also likely send detectable signals into ASIC markets, allowing governance mechanisms to be able to preemptively respond i.e. changing the hash function.

Let us imagine a hypothetical situation in which a large token holder (in industry terms, a whale) owns 5% of the total supply of Ether in the Serenity Ethereum blockchain with Casper. This whale could single handedly pull off a 33% attack if less than 15% of all Ether was staked. But because Ether has so much utility outside of staking [17], it will likely be very difficult to get 15% of Ether to be staked, causing an economic security risk.

Staking Only Tokens

So instead, the Cosmos Hub uses a multi-token model in which the Atom, is used primarily only for staking, and is not intended to be used to pay fees or used as a currency. Essentially the optimal utility of the token should be to stake it. But if this token no longer has use value derived from its function as a fee token, where does it get enough exchange value to give economic security to the system?

The utility of the Atom is that it is necessary for staking and thus gives you the ability to earn the transaction fees and the block rewards of chains validated by the Cosmos Hub validators, which includes the Cosmos Hub itself as well as any hosted chains. Atoms can basically be seen thus as somewhat analogous to ASICs in Proof of Work; it is a piece of virtualized hardware (economic capital) that you need to obtain in order to participate as a keeper in the network. Because partaking as a keeper in consensus allows you to earn transaction fees (paid in a token that has economic value as described above), the value of the staking token thus becomes the expected profit to be earned from the transaction fees and block rewards in the network, which are paid in a fee token. This can be observed as price fluctuations of SHA-256 ASICs are correlated with the expected earnings from mining Bitcoin. [20]

If you have a certain percentage of the staked Atoms, you have the ability to earn that percentage of the transaction fees and block rewards. There are two ways you can earn

more value. First, an increase in the value or amount of transaction fees going through the network means a larger pool of fee rewards for all validators, thus increasing the value of the transaction fees that are allocated to you. Second, you can increase your percentage share of the staked Atoms, thus affording you a larger percentage share of the transaction fees and block rewards. This can be done through buying more Atoms and staking them or by decreasing the share of other validators by getting them slashed.

In the Cosmos Hub, there are block provisions given in Atoms. These are technically analogous to block rewards but serve a different purpose. They are there in order to incentivize Atom holders to stake. Because Atoms are given as block provisions to stakers, there is a continuous inflation rate of Atoms. Atoms holders who do not stake are thus punished through inflation as their percentage share of the total Atom supply decreases as they do not receive any of the newly minted Atoms. This creates a significant incentive for Atom holders to stake their Atoms as there is a continuous transfer of wealth from non-stakers to stakers. The inflation of Atoms is denominated as a percentage of the total supply, and thus the block provisions per block increases as the total supply of Atoms increases. We use the term provisions rather than rewards because the function of the provisions is primarily to punish non-stakers, not to reward stakers. The inflation rate targeting mechanism is based on the percentage of staked Atoms. As the percentage of Atoms that are staked decreases, the inflation rate increases, thus further incentivizing Atom holders to bond their stake.

Note that this effect could have been achieved with demurrage, in which Atoms that are not staked begin to "decay" away (get burned). However, demurrage is more difficult to implement technically and would likely have a non-negligible negative impact around user experience, especially in the context of interchain transfers. For this reason, we choose the route of using inflation rather than demurrage.

Because of inflation going to only stakers and the lack of alternate designed utility for the Atom, it is reasonable to expect a large percentage of the total Atom supply will be staked. This makes it far harder for the situation described at the beginning of this section to occur in which a large whale stakes and gets over 33% of the bonded tokens. If anyone tries to buy the amount of Atoms required to get 33% of the stake on the open market, the market for Atoms would get progressively more illiquid causing the price of each successive Atom more expensive, greatly increasing the cost required to make an attack feasible.

4 Fee Tokens

Transaction Fees

In a blockchain, transaction fees serve two primary purposes: spam-prevention and a payment to keepers for their work in operating the blockchain. Blocks need to have some sort of scarce resource that limits the amount of work need to validate a block. A validator's mempool is the set of all valid transactions they have seen so far that haven't been inserted into a block yet. If a validator tried to put the entire pending mempool into the next block, a user could spam the mempool with many transactions causing the block to take too long to validate. In Bitcoin, this scarce resource is the block size and in Ethereum it is the gas limit. Because there is limited space in a block for a limited number of transactions, in order to determine which transactions from the mempool get added into a block, each transaction is accompanied by some sort of transaction fee that gets paid to keepers of the blockchain. The proposer of a block then uses these transaction fees to order the transactions by fee, thereby maximizing their earnings from proposing that block as the transactions with the highest fees get added to the block before the limit is reached. Different transaction use

differing amounts of resources, so instead of ordering by absolute amount of transaction fee of a transaction, we instead order by transaction fee per resource i.e. BTC per byte in Bitcoin or "gas price" (ETH per gas) in Ethereum.

The Cosmos Hub, having a limited number of types of transaction, will not use gas counting per opcode like most turing-complete blockchains, but rather assigns each transaction type a gas cost based on its estimated computational, storage, and network bandwidth costs. For example, because an IBC transaction is more computationally complex than a token transfer within Cosmos, the gas cost for an IBC Tx will be greater than that for a SendTx.

A chain can also implement a *MinimumGasCost*, the minimum resource cost that a transaction needs to declare in order to get included into a block. For example, currently Bitcoin transactions are required to pay a minimum of 1 satoshi per byte. This can be used for spam prevention in the case that a blockchain's blocks are not full. Furthermore, modifiable minimum fees can be used for more complex fee models than first price auctions, such as the Fixed Price Sale mechanism proposed by Vitalik Buterin. [5]

Many Fee Tokens

Another novel feature of the Cosmos Hub is that unlike most other public blockchain platforms, transaction fees can be paid in any *whitelisted fee token*. Most other platforms only allow fees to be paid in a single token, such as ETH in Ethereum, BTC in Bitcoin, and GAS in Neo. In line with the "multi-token multi-chain ecosystem" mindset of Cosmos, the Cosmos Hub, instead of forcing users to use a specific token in order to pay transaction fees in the system, allows users to select from a number of possible tokens. We will discuss how this works and then the benefits of such a system.

Let's delve into how to make this multi-fee token model work and allow for validators to be able to compare transaction fees paid in differing tokens/currencies in order to do transaction ordering. In the Hub, there exists a whitelist of tokens that are allowed to be used to pay fees in. The validators of the Cosmos Hub can add and remove tokens to the whitelist through *on-chain governance*, a mechanism for staked Atomholders to vote on changes to certain parameters within the protocol (such as the set of tokens in the fee token whitelist). The reason for needing to go through governance to add new tokens to the whitelist is that in the Cosmos Hub fees from every block are shared amongst validators, and thus it is necessary that there is consensus amongst the validators as to which tokens they are willing to accept as a group. As more chains and tokens join the Cosmos ecosystem, it can be expected that more of these tokens will become whitelisted fee tokens.

Transaction Ordering

In order to allow for validators to choose what order to run transactions, we need a mechanism to compare the value of the transaction fees being paid in different tokens. To do this, we need to know the *relative values* of each fee token. For the following example, assume we had a perfect oracle that informed us that the relative value of BTC is 5000 and ETH is 0.2 (note that these values are unitless).

Let us walk through an example of deciding the order of two transactions. Let's say we have two SendTx of equal gas cost and the tx fees on them are 0.5 BTC and 13000 ETH respectively. The first one's transaction fee is worth 2500 (5000×0.5) while the second one's is worth 2600 (0.2×13000). Thus the second transaction will have priority to go into the block over the first one.

So if given the relative values of each fee token, it is trivial to order them by fee. However,

how would we get these values without the use of centralized oracle? One potential option is to pull prices from an on-chain decentralized exchange, even one on a different chain by using IBC. However, while this may be a possible strategy in the future, currently DEXs suffer from a lack of liquidity allowing malicious attackers to cheaply game orderbooks causing reported prices to not match the prices on off-chain exchanges (likely closer to the true price of the asset). [18]

Thus we propose 2 possible mechanisms for determining the relative token valuations.

Option 1 – Localized Validator Config

Option 1 is that we completely forgo trying to come to consensus on the relative values, but rather let each block's proposer use their localized view of how much they value each token to order transactions when it's their turn to propose. Each validator node has a personal config file which lists their personal weighting of how much they value each token. These weights are what the validator will use to determine transaction ordering when it is their turn to be a proposer.

Let's imagine a config file for the Hub in which two tokens, Dogecoin (DOGE) and Photons (PHO), are whitelisted as fee tokens.

```
DOGE: 1
PHO: 1
```

Each validator can manually adjust these values to their liking (these values were just chosen randomly as an example):

```
DOGE: 5
PHO: 3
```

As more tokens get whitelisted, they get added to the config file as such:

```
DOGE: 5
PHO: 3
BTC: 0
ETH: 0
DAI: 0
```

Then each validator can also adjust these new values as well:

```
DOGE: 5
PHO: 3
BTC: 5000
ETH: 0.2
DAI: 1
```

As these numbers are simply ratios, these values can be normalized to any value. However, a common choice might be to normalize them to the the dollar value of the token.

Having such a config file allows validators to run a separate process like a cron job that live updates the configuration file based on live market data pulled from their favorite exchange's API (or combining the data of multiple!). If the exchange API is unreachable, the validator can set the config to fallback to the last available values, pull data from an on-chain exchange, or even resort to a backup default config (such as one in which Atoms have a weight of 1 and everything else has a weight of 0).

Note that a validator can also choose to set a MinimumGasCost in the config file, and choose to drop transactions from the mempool that do not meet their MinimumGasCost.

Option 2 – In Consensus Weighted Median

The second option is to employ a mechanism that still allows validators to come to consensus on the relative values of the tokens. Like in option 1, each validator continues to maintain a local view of their valuation of each token. However, instead of using their local view to order transaction, only when they are a proposer, they instead submit their "votes" for the value of each token on-chain as a transaction.

Let's say for example, there are 4 validators, {A, B, C, D, E}, with powers {0.1, 0.1, 0.2, 0.3, 0.3} respectively.

They submit the following votes for their personal views of each token:

- **A:** DOGE = 100, PHO = 0.4
- **B:** DOGE = 90, PHO = 0.45
- **C:** DOGE = 102, PHO = 0.39
- **D:** DOGE = 101, PHO = 0.41
- **E:** DOGE = 100000, PHO = 0

In the chain, we store each of the values in an ordered list, along with their validator that placed that vote.

- **DOGE:** $[90_B, 100_A, 101_D, 102_C, 100000_E]$
- **PHO:** $[0_E, 0.39_C, 0.4_A, 0.41_D, 0.45_B]$

Then, when a validator wishes to propose a block, they take the **weighted median** (weighted by stake) of the votes for each fee token, to get the in-consensus value of each token.

- **DOGE:** $WeightedMedian([(90, 0.1), (100, 0.1), (101, 0.3), (102, 0.2), (100000, 0.3)]) = \mathbf{101.5}$
- **PHO:** $WeightedMedian([(0, 0.3), (0.39, 0.2), (0.4, 0.1), (0.41, 0.3), (0.45, 0.1)] = \mathbf{0.395}$

At any point, any validator can send a transaction to update their vote. For example, let's say validator E, updates his vote:

- **E:** DOGE = 99, PHO = 0.43

The new ordered lists, thus become:

- **DOGE:** $[90_B, 99_E, 100_A, 101_D, 102_C]$
- **PHO:** $[0.39_C, 0.4_A, 0.41_D, 0.43_E, 0.45_B]$

and the new median values become:

- **DOGE:** $WeightedMedian([(90, 0.1), (99, 0.3), (100, 0.1), (101, 0.3), (102, 0.2)]) = \mathbf{100.5}$
- **PHO:** $WeightedMedian([(0.39, 0.2), (0.4, 0.1), (0.41, 0.3), (0.43, 0.3), (0.45, 0.1)] = \mathbf{0.41}$

The medians would likewise be recalculated upon changing of a validators stake/voting power.

The purpose of using weighted median, rather than a metric such as weighted average, is it is not affected by outliers. Assuming that $< 1/3$ of the stake is voting in a byzantine manner, the median value will always be between the values sent by correct processes.

As updating a vote requires sending an on-chain transaction, it is likely not feasible for a validator to send an update on an overly-frequent basis. One possible option is to send an update, only when your perceived value of a token changes by more than a certain percent from your last vote.

It is also important to note that this mechanism does require validators to agree upon a unit to normalize around, as if they were all normalizing their weights around different values, this mechanism would not work. One suggested proposal is that validators treat the ATOM token as a value of 1, and the relative value of the fee tokens are normalized around that.

A similar weighted median mechanism can also be used to gain consensus on a global MinimumGasCost.

Comparing the Two Mechanisms

These two mechanisms both achieve a similar end goal in that they allow validators to have some knowledge of relative values of the different tokens without reliance on a centralized oracle. However, each mechanism has it's own sets of pros and cons. Here is a brief rundown of some of the pros of each design:

- **Localized Validator Config:**
  - Is far more simplistic and easier to understand / reason about.
  - Requires no additions to the blockchain state machine and reduces computational overhead
  - Does not require validators to send on-chain txs which will also mean that the system is more likely to be able to quickly react to changes in market prices.
  - Validators aren't expected to track every token, they can choose to only track the whitelisted tokens they care about, and put 0 for the rest in the config file.
- **In Consensus Weighted Median:**
  - Provides a better UX, as there is a consistent global view for fee token values for clients/users to refer to, to make informed decisions when attaching fees to their transactions.
  - Tendermint provides a feature called CheckTx, which allows nodes in the network to do some basic verification of transactions when they first recieve them before deciding to propagate them or add them to their mempool. If MinimumGasCost was in global consensus, all nodes can use this to decide to drop transactions that do not meet in-consensus MinimumGasCost. This can be used to mitigate network and mempool spam for non-consensus nodes.
  - In Nakamoto consensus chains, because all the transaction fees of a block go to the block's proposer, a proposer can frontrun (insert their own transactions in front of other target transactions) for essentially no cost because they can attach a high transaction fee which will be sent back to themselves. For this reason, most Nakamoto chains do not even bother to enforce ordering of transactions by gas price.
    However, in Cosmos's Proof of Stake design, because fees are split amongst all validators, any transaction fee that the proposer attaches to their transactions, will be partially distributed to the other validators, not just going back to themselves. However, let's assume that there was whitelisted fee token, MEH, that does not see too much usage. If there was no global ordering of relative values, a block proposer who wished to frontrun, could do so for near 0 cost by attaching a fee of 1 MEH for their entire transaction, but then claiming that their local value of MEH token was MAX_VALUE. If there was global ordering of relative values, part of the consensus rules for block proposals could be to make sure that all transactions in a block are actually ordered by *fee value*. Because of this, an individual proposer cannot front run for near zero cost as they would have to attach a non-negligible amount of fees to be able to put

their transaction at the front of the block, and their non-negligible fee will be split amongst all the validators.

The Cosmos Hub will initially use mechanism 1, but will likely shift towards mechanism 2 upon development of the software and approval from governance. It is encouraged that chains looking to use Multi-Fee Tokens use the mechanism that is best suited for their use case.

Advantages of Multi Fee Tokens

Having many tokens usable as fee tokens heavily reduces the friction involved in using the Cosmos Hub. Because the primary use of the Cosmos is for users to move their tokens from one chain to another chain, users are likely to own the token that will be moved and want to pay fees in that token. In a system with only a singular fee token, it creates a poor user experience because now users would have to obtain the fee token just to make their transactions. This will likely increase the usage of the hub, as the friction in using the hub is greatly reduced. Regarding the complexity of choosing which token to pay with from the tokens the user owns, this choice will likely be dealt with by wallet software so that the average user does not have to think about it, much like how *amount of fee to pay* is already done so.

Along with providing a better experience to users, this multiple fee token model also provides a better experience for validators. Unlike other protocols that impose one token as the first class token in the chain, the Cosmos Hub lets validators have a sovereign view on what is valuable. The validator set of Atom holders will inevitably have multiple and diverse preferences about what tokens they find valuable in the cryptocurrency ecosystem, and so we can allow them to accept and be paid in a variety of these different tokens. It is more work on the part of the validators to maintain a relative weighting of different tokens over time, but it is to give them more freedom in how they want to be paid.

5 Related Work

The majority of Proof of Stake based blockchains/proposals such as Ethereum 2.0, Tezos, and Cardano all use a *single token model* as discussed earlier [17] [9] [3]. However, a few other Proof of Stake projects also seem to be experimenting with novel token models. Here, we will explain some of these systems and how their differ from the model proposed by the Cosmos Hub.

NEO

A common question received is how this staking token model differs than that employed by the NEO blockchain. According to the NEO whitepaper [7]:

> "NEO network has two tokens, NEO representing the right to manage NEO blockchain and GAS representing the right to use the NEO Blockchain."

At the surface, the NEO token seems similar to the *Staking Only Token* with GAS serving as a separate *Fee Token* to pay fees in. However, upon further reading of the NEO documentation, it becomes clear that the NEO model is very different.

Firstly, NEO is not actually required to be staked to become a validator. Rather it is used as a mechanism to determine how many votes each NEO account gets. Each account

can vote for as many validator candidates as they wish, and each validator candidate they vote for recieves the number of votes equivalent to the amount of NEO in the voter's account. For example, if a voting account has 10 NEO, and votes for 4 validator candidates, each candidate recieves 10 votes. According to the documentation, when declaring a validator candidate, 1000 GAS will be charged to the candidate. This seems to be the only charge to the candidate; at no point are any NEO tokens required to be staked. Therefore it does not seem comparable to the staking token model proposed by the Cosmos Hub.

With regards to fee tokens, as mentioned earlier, the NEO blockchain only supports a single Fee Token, GAS. Furthermore, the NEO blockchain uses a global MinimumGasCost for transactions, however, there is no mechanism for validators to adjust this MinimumGasCost, as this price is fixed at 1 GAS = 1 resource accounting unit (gas). A transaction that costs 5 *gas*, also costs 5 GAS tokens to be paid as its transaction fees. This means there is no mechanism for transaction costs to adjust to changes in the market price of GAS. If the market price for GAS was $0.01, a transaction that costs 100 GAS costs $1.00 in USD value. However, if the market price of GAS increased to $1.00, that same transaction still has a MinimumGasCost of 500 GAS but has a cost of $500.00 in USD terms. Without a mechanism for validators to adjust the global MinimumGasCost, the volatility in USD value of GAS token will cause massive user experience problems.

As can be seen, the NEO model differs significantly from the Cosmos Hub's staking token and multi-fee token model.

EOS

Another blockchain that seems to have a somewhat similar staking token model to the Cosmos Hub model is EOS [13]. In EOS, each EOS token holder can stake their EOS tokens in order to vote for block producers as well as receive resources such as CPU, NET, and RAM, resource units that are used for making transactions on the blockchain.

However, like in NEO, the staking token EOS is not actually staked by the block producers, and it is not slashable in the case of misbehavior. It is only used to distribute votes to "staked" token holders. In EOS, the term staking does not necessarily imply being a validator or contributing to consensus. Rather, staking your tokens means you're putting your tokens in a lockup period, during which they can be earning you either CPU, NET, or RAM resources which can be used to pay for computational capacity, network capacity, and storage capacity respectively. These resource units are not exactly tokens, as they are not transferable between users. CPU and NET are only spendable by the original receiver of them, while RAM can be bought and sold to other users, but only through the chains built-in Bancor-style marketplace. [6]

These resources, went spend to make transactions, are not given to the block producers, but are rather burned. But if validators are no longer being paid in transaction fees for the contribution in maintaining the network, how are they being compensated? The premise of the compensation of block producers is in that they are rewarded with block rewards in EOS tokens, which have use value because of their necessity to earn resources which afford usage of the EOS blockchain. Despite having more nuances and steps than the traditional single token model, the EOS token seems to be more similar to a single token model that a staking only token model.

6 Conclusion

The paper explains the token model of the Cosmos Hub, especially with regards to the incentives and utilities of different tokens in the staking and fees mechanisms. The Atom token acts as a pure staking token with high inflation and minimal utility outside of staking, thus reducing liquidity of free Atoms on the open market and improving the security of the staking system. We also present two designs for constructing a multi fee-token system, which allows users to pay their transaction fees in a variety of whitelisted fee-tokens, which will greatly improve user experience. As mentioned previously, while this token model was originally designed for the Cosmos Hub, we believe that similar designs can be used in other blockchain systems.

References

1. Sunny Aggarwal. Cosmos proof of stake. Crypto Economics Security Conference 2018, 2018. URL: https://github.com/cosmos/cosmos-academy/blob/master/presentations/2018-10-10-cesc/cosmos-pos.pdf.
2. Et al. Whitepaper:nxt. 2013. URL: https://nxtwiki.org/wiki/Whitepaper:Nxt.
3. Lars Brünjes and Et al. Incentives and staking in cardano. 2018. URL: https://staking.cardano.org/.
4. Ethan Buchman, Jae Kwon, and Zarko Milosevic. The latest gossip on BFT consensus. *CoRR*, abs/1807.04938, 2018. URL: http://arxiv.org/abs/1807.04938, arXiv:1807.04938.
5. Vitalik Buterin. Evaluate alternative transaction fee market mechanisms. 8 2018. URL: https://github.com/zcash/zcash/issues/3473.
6. EOSReal. Eos ram 101: Non-technical guidebook for beginners. 7 2018. URL: https://medium.com/coinmonks/eos-ram-101-non-technical-guidebook-for-beginners-6f971322042e.
7. NEO Foundation. Neo whitepaper. URL: https://docs.neo.org/en-us/whitepaper.html.
8. Ethan Frey, Christopher Goes, and Others. The latest gossip on BFT consensus. 2018. URL: https://cosmos.network/docs/spec/ibc/.
9. L.M Goodman. Tezos — a self-amending crypto-ledger white paper. 9 2014. URL: https://tezos.com/static/papers/white_paper.pdf.
10. Jae Kwon. Tendermint: Consensus without mining. 2014. URL: https://github.com/cosmos/cosmos/blob/master/tendermint/main.pdf.
11. Jae Kwon and Sunny Aggarwal. Interchain scaling security models. Crypto Economics Security Conference 2018, 2018. URL: https://github.com/cosmos/cosmos-academy/blob/master/presentations/2018-10-10-cesc/interchain-scaling-models.pdf.
12. Jae Kwon and Ethan Buchman. Cosmos - a network of distributed ledgers. 2017. URL: https://github.com/cosmos/cosmos/blob/master/WHITEPAPER.md.
13. Dan Larimer and Et al. Eos.io technical white paper v2. 3 2018. URL: https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md.
14. Daniel Larimer, Charles Hoskinson, and Stan Larimer. Bitshares: A peertopeer polymorphic digital asset exchange. 2013. URL: https://blog.bitmex.com/wp-content/uploads/2018/06/173481633-BitShares-White-Paper.pdf.
15. Karl Marx. *Das Kapital: A Critque of Political Economy*. 1867.
16. Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008. URL: https://bitcoin.org/bitcoin.pdf.
17. Anthony Sassano. Why ether is valuable. 2019. URL: https://medium.com/ethhub/why-ether-is-valuable-2b4e39e01eb3.
18. Sam Town. The final frontier of crypto decentralization: Dexs and the liquidity problem. 2018. URL: https://cryptoslate.com/the-final-frontier-of-crypto-decentralization-dexs-the-liquidity-problem/.

19 David Vorick. Choosing asics for sia. 6 2017. URL: https://blog.sia.tech/choosing-asics-for-sia-b318505b5b51.

20 Josiah Wilmoth. Bitcoin miners are selling old asics for scrap metal as price decline hastens obsolescence. 11 2018. URL: https://www.ccn.com/bitcoin-miners-are-selling-old-asics-for-scrap-metal-as-price-decline-hastens-obsolescence/.

21 Ryan Zurrer. Keepers—workers that maintain blockchain networks. 2017. URL: https://medium.com/@rzurrer/keepers-workers-that-maintain-blockchain-networks-a40182615b66.