Cosmos discovery: Quantitative assessment of Cosmos blockchain

Daria Smuseva, Carla Piazza
Università degli Studi di Udine
Udine, Italia
{daria.smuseva,carla.piazza}@uniud.it

Ivan Malakhov, Andrea Marin, Sabina Rossi
Università Ca' Foscari Venezia
Venezia, Italia
{ivan.malakhov, marin, sabina.rossi}@unive.it

***Abstract*—Blockchain technology has experienced significant advancements, with Proof-of-Stake emerging as a notable alternative to traditional Proof-of-Work blockchains. Among various PoS blockchain systems, Cosmos stands out as a prominent example due to its ecosystem designed to facilitate interoperability between different blockchains built on their platform through the Inter-Blockchain Communication protocol. What is more, Cosmos is operated by the unique consensus mechanism, namely CosmosBFT that supports multiple rounds for an agreement on block of the same height. This study examines the current state of blockchains within the Cosmos ecosystem, highlighting two major issues. First, we observe the multi-round performance in Cosmos blockchain using the process algebra tool to create our model featured non-homogeneous proposers. Second, we propose a method for determining optimal timeouts for the Propose step in any network within the ecosystem. In addition, we identify a skewed distribution of voting power among validators, favouring top-ranked members. This concentration of VP threatens the network's decentralisation and immutability, as it allows a small group of members to potentially corrupt the consensus process. Our models, although parameterised for a particular Cosmos instance, are applicable to any blockchain that use the CometBFT protocol, offering valuable insights for enhancing efficiency of consensus mechanisms in the decentralised networks.**

***Index Terms*—blockchain, Proof-of-Stake, performance evaluation.**

I. Introduction

Blockchains are distributed networks of users with an intentional lack of central authority that store data across a network in the form of linked blocks. Despite the variability of potential use cases such as logistics, healthcare, and IoT, the most dominant application remains cryptocurrency. In such networks, users are usually incentivised to participate and extend the network as they are rewarded with the system's internal currency, e.g., BTC in Bitcoin or ETH in Ethereum.

In today's landscape of blockchain technology, two primary types of consensus algorithms are employed: Proof-of-Work (PoW) and Proof-of-Stake (PoS). The former, dating back to the earliest days of blockchain, relies on miners to validate and extend the network by spending significant computational resources, as outlined in Satoshi Nakamoto's seminal paper [1]. In contrast, PoS blockchains emerged post-Bitcoin, offering a compelling alternative where validators stake their cryptocurrency to secure and validate transactions without the need for extensive computation.

Among the variety of PoS blockchain systems (including Ethereum[1], Cardano[2], and Algorand[3]), Cosmos[4] stands out as a prominent example. Firstly, it is not merely a single blockchain network but an entire ecosystem designed to facilitate interoperability between different blockchains by using Inter-Blockchain Communication (IBC) protocol [2]. The ecosystem refers to the interconnection of blockchain networks, allowing native and secure communication between any network instances within Cosmos. Moreover, all such networks utilize a unique consensus mechanism, CometBFT [3], based on the Tendermint protocol [4], which underpins its PoS-based architecture. Thus, the Cosmos consensus requires more steps for block commitment, namely Prevote and Precommit, that follow the (block) Propose step. Each step has its default timeouts to control the block agreement time. Blockchain throughput and scalability heavily depend on the appropriate choice of timeouts: short timeouts waste network performance efficiency by interrupting the validation process and causing process repetition, while long timeouts introduce undesirable waiting times in the case of faulty validators. Thus, studying the trade-offs is vital in the context of this type of blockchain. Furthermore, depending on the type of data a network plans to share, including possibly complex smart contracts that may take hours to process, new Cosmos instances would require proper timeouts for the consensus.

In Cosmos networks, as well as in any PoS network, validators play a crucial role. They are responsible for extending the ledger by validating transactions and creating blocks, thus securing the network's integrity. The validators stake their cryptocurrency as collateral, demonstrating their commitment to the network's security and reliability. However, unlike in Ethereum, where each block is validated by a randomly selected subset of an unlimited number of validators, in Cosmos, the validators' committee is fixed in quantity and consists of members with the highest stakes. It may happen that validators do not reach a consensus about a new block. In Ethereum, this would lead to voting on some of the previous blocks, while in the Cosmos ecosystem, the validators must start a new *round* within the same block position until a block is eventually

[1]https://ethereum.org
[2]https://cardano.org
[3]https://algorandtechnologies.com
[4]https://cosmos.network

accepted. This guarantees the absence of any forks, as a trade-off for potentially lower network performance. A network with a majority of blocks approved not in a single round would experience decreased throughput, as the transactions in the blocks would not be accepted for a longer time due to the protocol's policy.

Contribution. In this paper, (i) we describe two problems related to the consensus process in Cosmos ecosystem using an eponymous blockchain instance as a viable example. Specifically, we study network performance considering the multi-round nature of the underlying process and introduce a method to determine optimal timeouts for a blockchain in Cosmos. (ii) We create an analytical model that reflects the validation process using the Performance Evaluation Process Algebra (PEPA) [5]. Finally, (iii) we perform a comprehensive assessment of the model's outcomes, providing numerical insights.

Our study sheds light on the current state of blockchains within the Cosmos ecosystem, highlighting the following:

- We study the performance of the network, evaluating single- and multi-round behavior. We show that, since the delays associated with the block consolidation phases depend both on the block content and the speed of the validators, the appropriate setting of the timeout has a strong impact on the system's throughput.
- We propose a model-based method for finding optimal timeouts for the Propose step for any network within the ecosystem. Alternatively, using the timeouts of existing Cosmos instances, we can determine their average time spent at each consensus step.
- Our work can be used to study any blockchain instance in the Cosmos ecosystem. Although our model was parameterized and studied with respect to the Cosmos blockchain, it is applicable to any blockchain using the underlying CometBFT protocol.
- Additionally, we show that the current state of Cosmos blockchains suffers from unfairness, i.e., the skewness of the validators' voting power (VP) towards the top-ranked members. Such a high concentration of VP in a small group threatens the decentralization and immutability of the network, as they can easily corrupt the consensus process and modify the network.

Paper structure: Section II discusses related work. Section III delves into the background of Cosmos and PoS, and provides a brief analytical overview of the Cosmos system based on blockchain data. Section IV introduces our model based on the PEPA tool. Finally, Section V concludes the paper.

II. Related Work

The evaluation of blockchain network performance remains a vibrant area of interest within the academic community [6], [7]. In their study, Fan et al. [6] categorise evaluation methodologies into empirical analysis and analytical modeling. Empirical analysis encompasses benchmarking, monitoring, experimental analysis, and simulation, while analytical modeling delves into stochastic models for assessing mainstream blockchain consensus algorithms.

Recent research has focused on Inter-Blockchain Communication within the Cosmos ecosystem [8], [9], [10]. Wu et al. [8] introduce a queuing model that utilizes a three-dimensional continuous-time Markov process to optimize cross-blockchain transaction performance. The model's effectiveness is validated through experimental simulations, suggesting its potential applicability to other cross-blockchain systems. Similarly, Han et al.[9] propose a Cosmos blockchain-based Central Bank Digital Currency that leverages the IBC protocol for blockchain communication. Their study encompasses system requirements, design, and implementation, introducing a Group Key Management system for user privilege allocation and privacy preservation in key generation. In contrast, our work focuses on evaluating performance metrics of Cosmos networks with respect to optimal timeout of consensus steps and throughput dynamics across various numbers of rounds.

In addition, Smuseva et al. [11], [12] leverage Markovian Process Algebra to explore variants of the Verifier's Dilemma in PoS Ethereum, proposing a countermeasure involving the injection of invalid blocks through quantitative model-based analysis. This paper utilizes the same framework but primarily focuses the blockchains within the Cosmos ecosystem.

III. Background and Motivation

In this section, we provide a brief description of the Cosmos network delving into the protocol underlying it.

A. Proof-of-Stake

The core of any blockchain network is its consensus algorithm. Currently, a variety of consensus mechanisms underly blockchain protocols, from the well-known PoW and PoS to the more exotic Proof-of-Burn implemented in Slimcoin[5]. In this study, we focus on the Cosmos network with its underlying PoS consensus mechanism.

Proof-of-Stake is a type of consensus algorithm that has gained popularity due to its efficiency and security. Unlike PoW, which requires nodes to perform complex computations to add new blocks, PoS generally selects the creator of a new block based on their stake - the fraction of deposited or possessed tokens in the network. This method significantly reduces energy consumption, making PoS a more sustainable option for blockchain realization[6]. In PoS blockchains, validators hold or sometimes lock part of their funds, representing a stake. The size of the stake determines the likelihood of a node being chosen to propose the next block, as well as the validator's voting power. This setup naturally stimulates good behavior and keeps the network more secure. Validators are motivated to act honestly, as they have a financial stake in the network and stand to lose their stake if they attempt to misbehave.

[5]https://slimcoin.info

[6]See https://ccaf.io/cbnsi/cbeci and https://ccaf.io/cbnsi/ethereum for comparison.

B. *Cosmos blockchain*

Recall that Cosmos ecosystem can be seen as a platform of blockchain networks within. All such blockchains take part of the system as they can interact with each other using IBC protocol, and can have their own currency but remain rather independent networks. The Inter-Blockchain Communication protocol enables secure and trustless data exchange between different blockchains in Cosmos ecosystem. It facilitates interoperability, allowing various blockchains to communicate and transfer tokens and data seamlessly.

Cosmos blockchains utilise a single PoS protocol, namely CometBFT, that might be adjusted depending on the system requirements. Furthermore, Cosmos ecosystem includes an eponymous blockchain (sometimes called Cosmos Hub) that we use as a main example since it reflects the most common protocol configuration in the ecosystem, referring to it as a network instance rather than an ecosystem.

In the Cosmos protocol used by all blockchains in the ecosystem, only the validators with the largest stakes are involved in securing the network by validating transactions and producing new blocks [13]. Essentially, anyone can become a validator, but they must stake (or bond, in Cosmos terms) a significant amount of tokens to compete with the highest stakeholders. For example, the top validator controls over 22 million tokens, worth around 220 million USD. Validators can achieve this stake using their own balance or by having delegators contribute funds in exchange for a share of the rewards. In this instance, the top validator personally holds only two million tokens, with the rest delegated by others.

Note that the current set of active validators in the Cosmos blockchain consists of the top 180 validators. This number has increased multiple times and may continue to grow as the network expands [14]. If a validator drops out of this top list, they lose their validator status.

a) Consensus algorithm: The consensus process of CometBFT consists of several steps (see [15]) that can be visualised as follows:

$$\texttt{NewHeight} \rightarrow (\texttt{Propose} \rightarrow \texttt{Prevote} \rightarrow \texttt{Precommit})^{\geq 1} \rightarrow \texttt{Commit} \rightarrow \ldots$$

In turn, three special steps, namely `Propose`, `Prevote`, and `Precommit` form a single `Round`, as described below.

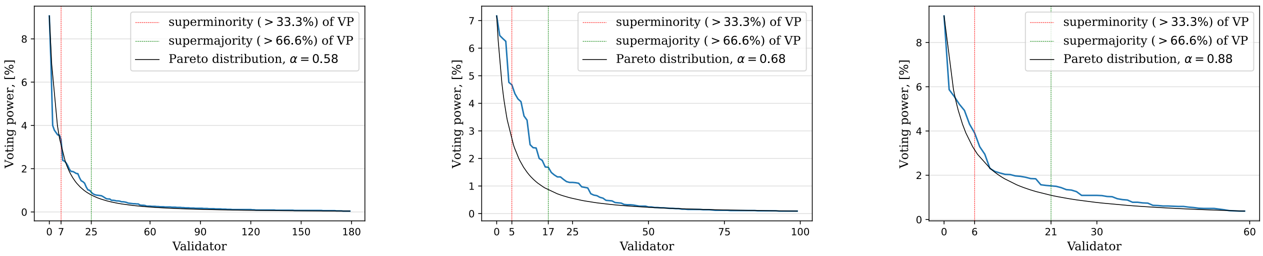
1) *Propose*: A validator is selected as the proposer using a weighted round-robin algorithm. They form and verify a block of pending transactions, then broadcast it to other validators. The block includes a hash of the previous block, a timestamp, the proposer's signature, and may contain proofs of misbehavior for penalties, such as voting for multiple blocks or introducing multiple blocks simultaneously.
2) *Prevote*: Each validator checks the proposed block's validity and broadcasts a vote for or against it. Validators can see each other's votes. A vote is a signed message with essential data about the validator and the block. Validators can also cast a nil vote if they didn't receive a valid block proposal in time or found the block invalid.
3) *Precommit*: If over $\frac{2}{3}$ of validators prevote for a block, they precommit to it; otherwise, they vote *nil*. Validators then wait for enough votes to secure that supermajority of the VP belongs to those voters.

The round steps are repeated until the validators holding at least $\frac{2}{3}$ of the total voting power precommit for the same block, which is then committed and added to the blockchain. If this does not occur within a timeout period, the next round is started with a new proposer and increased round timeouts but for the same block height (block number). The CometBFT algorithm (and any PoS protocols, in general) ensures that the validators can reach consensus on a unique block in a finite number of rounds, as long as up to $\frac{1}{3}$ of the voting power is controlled by malicious or faulty validators inherent from [16]. Next, `Commit` step is needed to perform a block commit and move again to `NewHeight` repeating the validation process.

b) Fairness: One of the common concerns in blockchain networks is the lack of *fairness*, which can put validators following the protocol in a disadvantageous position as it becomes more rewarding violating it like in the case of Verifier's Dilemma in PoS [12]. Fairness means that participants receive a fair share of network rewards based on their contribution, typically measured by their voting power, determined by their token holdings. Currently, Cosmos network utilises only this type of fairness. While this notion is legitimate, it encompasses only one aspect focusing on the *user's perspective* that is the reward for any single validator. Nevertheless, from a *system perspective*, fairness refers to a perfectly balanced network where no validator holds a disproportionately large stake. In an ideal scenario, the network features a validator committee with evenly distributed voting power, ensuring that the majority of votes align with the majority of voting power, with each vote carrying equal weight.

Recall that in a PoS network, the core security assumption dictates that the network can withstand at most $\frac{1}{3}$ of adversarial activity. This means that a single validator or a group with voting power exceeding $\frac{1}{3}$ could compromise the network's integrity, rendering consensus unattainable. Therefore, balancing the network becomes paramount.

Fig. 1 shows the voting power distribution among the top three blockchains in the Cosmos ecosystem. A consistent pattern emerges, resembling a bounded Pareto distribution of stakes across these networks. Note that the parameter α shown in the figures is obtained with a maximum likelihood estimation. In all cases α lies lower than 1 intending rather heavy-tailed nature of the corresponding (unbounded) Pareto. Thus, we can safely assume that a bounded Pareto distribution is a viable approximation of the stake distributions in Cosmos ecosystem's blockchain instances. Consequently, a supermajority of voting power, over $\frac{2}{3}$, is held by a small group of top validators, leaving the rest of the network with minimal voting power, almost zero. This concentration means that if even a few members of this dominant group are unavailable, as seen when the top seven validators in Cosmos, holding a superminority of voting power (more than $\frac{1}{3}$ of the total),



(a) Cosmos with 180 validators. (b) Celestia with 100 validators. (c) Injective with 60 validators.

Fig. 1: Distribution of the validators' VP in the top three blockchains in Cosmos ecosystem.

become unavailable, the network can't achieve consensus, resulting in paralysis.

A potential solution could involve setting an upper limit on delegated power, ensuring it does not exceed a fair share of the network's voting power. However, a thorough examination of mitigation strategies is beyond the scope of this paper.

c) Block creation time and transaction throughput: Fig. 2a demonstrates a probability density function of block inter-arrival time for Cosmos blockchain. The data was retrieved from a blockchain data service[7] and contains information about more than $70,000$ blocks collected during 5 days from 2024-03-23 to 2024-03-28. It is clear that the majority of blocks appear approximately every 6 seconds which quite well matches the protocol settings. Interestingly, it shows that apart from the standard block creation time there is a notable interval within 2 seconds.

Fig. 2b reflects the transaction throughput during the same time interval of 5 days. While, in general, it tends to stay on average around 5 tx/s there are few evident spikes happening during the days. Notably, in the period between 2024-03-24 and 2024-03-25 we can see the rapid growth of transaction activity with the corresponding tenfold increase in their throughput from 5 to 50 tx/s. This signalises us that, despite the overall stability, the network can suffer from exceptional transaction arrivals affecting the consensus process.

IV. Model description and assessment

In this section, we introduce a PEPA model that describes the consensus process utilised in Cosmos ecosystem. All the results presented are obtained by solving the Markov chain underlying the PEPA model using the PEPA Eclipse plug-in.

A. Model Description

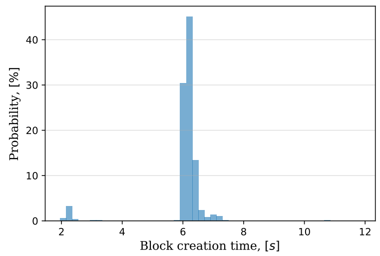
Table I describes the model that reflects the consensus protocol implemented in any network within Cosmos ecosystem. To study a blockchain within the ecosystem, we only need to parameterize the model with real system variables. We configure it using key step parameters from Table II. We assume that all validators can be categorised by their possessed voting power into three groups derived from Fig. 1a. The first category comprises validators with the largest stakes, holding a superminority of VP ($\frac{1}{3}$ of the total), this category encompasses the top seven validators of Cosmos blockchain. We assume that this set of validators are, in general, *superfast* in performing their block proposals than all others as they possess higher stakes, thus committing to be always reliable and prompt to response with all necessary resources to execute the consensus. The second category includes validators ranked 8 to 25, collectively holding another third of VP. We believe that such validators are slower in proposals, however, they are able to execute the Propose step by their timeout. We call them the *fast* performers. Finally, the remaining numerous validators constitute the last category, collectively holding the last third of the total VP. In this context, we assume that these validators are rather incapable of doing their proposals in time and they are always quite *slow* such that an extra round is needed to try to reach the consensus.

The block creation process starts with the component $NewHeight$, where the system immediately starts round $Round_i$ (always initiating from $i = 1$ upon transiting from $NewHeight$) and with equal probability moves to one of the Propose steps, i.e., superfast ($Propose_{i_{FF}}$), fast ($Propose_{i_F}$), and slow ($Propose_{i_S}$). The transition to any of these proposal steps implies equal probabilities applied to the Round's rates, i.e., $p_{FF} = p_F = p_S = \frac{1}{3}$. If a block has been successfully created with the rate $w_{1_{i_{FF/F/S}}} \gamma_{i_{FF/F/S}}$ the system moves to the component $Prevote_i$. Otherwise, it moves to $NilPrevote_i$ with rate $(1 - w_{1_{i_{FF/F/S}}}) \gamma_{i_{FF/F/S}}$. $w_{1_{i_{FF/F/S}}}$ denotes the probabilities of completing the step before the timeout expires.

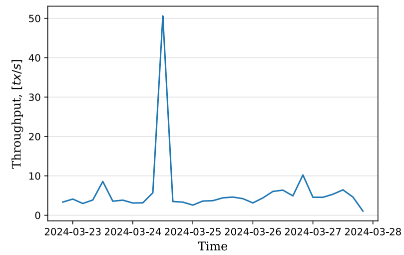
The Prevote step $Prevote_i$ has two outcomes as well: with the probability w_{2_i} validators successfully find an agreement and move to precommit state $Precommit_i$; with the probability $(1 - w_{2_i})$ the prevote is unsuccessful and it goes further to precommit stage that we call $Unsuccess_i$. By the unsuccessful prevote stage we assume two conditions: (i) the system has not collected votes from the validators with at least $\frac{2}{3}$ before the timeout of this step; (ii) the validators could not find an agreement. Thus, for simplicity we call the state after the unsuccessful Prevote $Unsuccess_i$, which always leads the model to a new round $Round_j$, where the round counter j is limited by R such that $j = min(i + 1, R)$.

The component $Precommit_i$ behaves with rate δ_i and successfully moves to $Commit_i$ with probability w_3. Then,

[7] https://flipsidecrypto.xyz



(a) Probability Density Function of block generation time.



(b) Transaction throughput with 4 hours granularity.

Fig. 2: Cosmos blockchain data visualisation.

TABLE I: PEPA model of consensus process with the non-homogeneous proposers in Cosmos ecosystem.

$$\begin{aligned}
\mathit{NewHeight} &\overset{def}{=} (nh, n).\mathit{Round_1} \\
\mathit{Round_i} &\overset{def}{=} (r, p_{FF}\, n).\mathit{Propose}_{i_{FF}} + (r, p_F\, n).\mathit{Propose}_{i_F} + (r, p_S\, n).\mathit{Propose}_{i_S} \\
\mathit{Propose}_{i_{FF}} &\overset{def}{=} (p, w_{1_{i_{FF}}} \gamma_{i_{FF}}).\mathit{Prevote_i} + (p, (1 - w_{1_{i_{FF}}})\gamma_{i_{FF}}).\mathit{NilPrevote_i} \\
\mathit{Propose}_{i_F} &\overset{def}{=} (p, w_{1_{i_F}} \gamma_{i_F}).\mathit{Prevote_i} + (p, (1 - w_{1_{i_F}})\gamma_{i_F}).\mathit{NilPrevote_i} \\
\mathit{Propose}_{i_S} &\overset{def}{=} (p, w_{1_{i_S}} \gamma_{i_S}).\mathit{Prevote_i} + (p, (1 - w_{1_{i_S}})\gamma_{i_S}).\mathit{NilPrevote_i} \\
\mathit{Prevote_i} &\overset{def}{=} (pv, w_{2_i}\beta_i).\mathit{Precommit_i} + (pv, (1 - w_{2_i})\beta_i).\mathit{Unsuccess_i} \\
\mathit{NilPrevote_i} &\overset{def}{=} (pv, \beta_i).\mathit{Unsuccess_i} \\
\mathit{Unsuccess_i} &\overset{def}{=} (pc, \delta_i).\mathit{Round_j} \\
\mathit{Precommit_i} &\overset{def}{=} (pc, w_3\delta_i).\mathit{Commit_i} + (pc, (1 - w_3)\delta_i).\mathit{Round_j} \\
\mathit{Commit_i} &\overset{def}{=} (c_i, \eta).\mathit{NewHeight}
\end{aligned}$$

where $\gamma_{i_{FF/F/S}} = max\left(\frac{1}{t_{1_{FF/F/S}}}, \frac{1}{T_1 + (i-1)g}\right)$, $\beta_i = max\left(\frac{1}{t_2}, \frac{1}{T_2 + (i-1)g}\right)$,
$\delta_i = \frac{1}{T_3 + (i-1)g}$, $i \in \{1, \ldots, R\}$ and $j = min(i+1, R)$, and $\eta = \frac{1}{T_4}$ while $p_{FF} + p_F + p_S = 1$

the system moves to the initial state $\mathit{NewHeight}$. Probability $(1 - w_3)$ indicates that agreement was not reached during the precommit step, prompting the system to initiate a new round.

Prevote step after the missed block $\mathit{NilPrevote_i}$ always moves the system to $\mathit{Unsuccess_i}$. Such round will never transit the system to $\mathit{NewHeight}$, but only to the new round $\mathit{Round_j}$.

In real systems, the maximum number of rounds is typically unbounded, and our model reflects this. However, we approximate the system's behavior for any round $i > R$ by using the same rates of $R = 6$.

Bear in mind that each step in the confirmation process has its own timeouts after which every next step starts. Table II shows the timeouts implemented in Tendermint Core and Cosmos blockchain such that a typical round should lasts 6 seconds [17]. Moreover, should multiple rounds be needed for reaching a new height each phase is extended on the value g, its timeout increase for any new round within the same block height, where $g = 0.5$s in Cosmos blockchain. Thus, augmenting chances of finding agreement in the next rounds. Below, we will closer examine the effect of different round augmentation. Note that the n possesses a super high rate due to the properties of the tool to demonstrate the immediate transition from *NewHeight* to *Round* component.

TABLE II: Time constraints of Cosmos blockchain consensus process.

Name	Duration	Description
Propose timeout, (T_1)	3s	Time duration for a proposal block before prevoting *nil*
Prevote timeout, (T_2)	1s	Time duration after receiving supermajority of prevotes for any block(s) or *nil*
Precommit timeout, (T_3)	1s	Time duration after receiving supermajority of precommits for any block(s) or *nil*
Timeout increase, (g)	0.5s	Propose, Prevote, and Precommit timeout increases per new round within the same block height
Commit timeout, (T_4)	1s	Time duration after committing a block, before starting on the new height

Furthermore, the Propose and Prevote rates are computed differently depending on how small are the actual service times of these steps, namely $t_{1_{FF/F/S}}$ and t_2. The respective rates are always the maximum between the processing and timeout rates. The reasoning behind this is fairly straightforward - the actual performance of any of these steps depends on how well the steps are executed by the validators and in case of poor execution they are limited by the given timeout for the steps.

Regarding $w_{1_{i_{FF/F/S}}}$ and w_{2_i} we assume that they are derived as the probability that an independent exponential random variable with mean value $t_{1_{FF/F/S}}$ (t_2) to be lower than the corresponding timeout $T_1 + (i-1)g$ $\left(T_2 + (i-1)g\right)$. For the first round, we have:

$$w_{1_{FF/F/S}} = Pr[X_{1_{FF/F/S}} \leq T_1] = 1 - e^{-\frac{1}{t_{1_{FF/F/S}}}T_1} \tag{1}$$

$$w_2 = Pr[X_2 \leq T_2] = 1 - e^{-\frac{1}{t_2}T_2}\,, \tag{2}$$

where T_1 and T_2 stand for default timeouts set for the first two steps in a Round, e.g., in Cosmos $T_1 = 3$s for Propose and $T_2 = 1$s for Prevote. We intentionally leave the probability of success for the Precommit step (w_{3_i}) unchanged, as the last step of a round depends solely on the network performance - specifically, how quickly node votes are propagated through the network, rather than on the processing of the block itself. Thus, although it is possible to configure the probability of failing on the Precommit step, to avoid complications, given its negligible nature we assume that $w_3 \to 1$.

B. Studying the throughput of the system

In this experiment, we employ the model detailed in Table I to evaluate the network throughput as a function of the validators' speed. To this aim, let us fix the timeouts T_1 and T_2 following the values from Table II for Propose and Prevote steps, respectively. We measure the speed of the validators in terms of these timeouts, i.e., $t_{1_{FF/F/S}} = kT_1$ and $t_2 = kT_2$. Thus, at $k < 1$ the average time to finish the steps is lower than the one of timeout implying that the validators perform the consensus steps faster while starting from $k > 1$ we can observe the opposite case.

Moreover, we aim to compare the performance results of the model with various types of the Propose step, that we can call the model with *non-homogeneous* proposers, with the one in which all the proposers act equally having the same rates for block proposal such that they are *homogeneous*.

Note that in the scenarios we assume that the network is completely fair which implies no malicious actors or parties are present in the system that can deviate from the protocol.

1) Non-homogeneous proposers: In order to determine the actual processing speeds of the proposers we refer again to Fig. 1a. We assume that the validators' VP for each of the categories strongly correlates with their Propose rates. First, we calculate the average VP per validator such that

$$\overline{VP}_{FF} = 0.0483, \quad \overline{VP}_F = 0.0184, \quad \overline{VP}_S = 0.0021\,.$$

Next, normalising our values with respect to the fast category ($\overline{VP}_F$) we then obtain the corresponding actual times assuming that the Prevote time of the fast one equals to their timeout (3s). Finally, the actual Prevote rates for three categories of the validators are:

$$t^{-1}_{1_{FF}} = \frac{\overline{VP}_{FF}}{\overline{VP}_F}\frac{1}{3}, \quad t^{-1}_{1_F} = \frac{\overline{VP}_F}{\overline{VP}_F}\frac{1}{3}, \quad t^{-1}_{1_S} = \frac{\overline{VP}_S}{\overline{VP}_F}\frac{1}{3}\,. \tag{3}$$

Fig. 3a and 3b show the network throughput as functions of coefficient k for Prevote only and for Propose and Prevote steps together. Interestingly, the former figure shows that despite generally faster service times, around 40% of the total throughput is achieved using more than one round (indicated by the green line). This is because encountering the slow Propose step, which occurs with equal probability, ultimately necessitates the use of multiple steps. Regarding the latter figure, unlike the case of Prevote step only parameterisation, we observe that at $k \ll 1$ the multi-round performance is approaching zero while starting from its peak at $k = 1$ it overcomes the single round throughput and gradually decreases reaching close-to-zero values along with the total throughput that at this moment almost fully consists of multi-round performance.

2) Homogeneous proposers: Fig. 3c illustrates the system throughput variation concerning Prevote duration w_2 as a function of coefficient k. We fix success probability of $w_{1_h} = 0.63$ that corresponds to the situation when $t_{1_h} = T_1$. It is evident that shorter verification times enhance network throughput. Conversely, delays in executing votes - such as in cases where blocks contain numerous large smart contracts requiring re-execution - can cause throughput to approach zero. In general, its performance patterns greatly mirror the results of the non-homogeneous model. Additionally, starting from approximately $k = 1.5$ the multi-round throughput, i.e., when a network needs more than 1 round for the consensus, surpasses that of a single-round consensus, indicating potential struggles among validators to reach agreement on about half of the new blocks. Similarly, Fig. 3d shows the throughput as a function of changing parameter k this time for Propose and Prevote steps at the same time. When Propose and Prevote timeouts reach 3s and 1s respectively at $k = 1$, the network experiences nearly a 90% decrease in performance that would suggest the presence of exceedingly complex pending transactions that even block proposers cannot include efficiently and in a timely manner.

C. How to parameterise the model estimations

In the realm of blockchain development, crafting a new blockchain requires careful consideration of various factors, with block verification speed being particularly crucial. For instance, when designing a blockchain intended for financial transactions, minimising the processing time of blocks is crucial due to the relatively low computational demands expected from validators. Conversely, a blockchain geared towards executing heavy computations may require a different approach, potentially necessitating longer waiting times to accommodate the intricate processing involved.

One pertinent question arises: *"How does one determine the optimal processing rate?"* In this context, presume we have a known average block processing time which is suitable for both the block proposer and other validators, considering their requirement to re-execute all transactions within a block. Initially, one might set a timeout for a step to an arbitrary time, but this approach may not optimize performance. Hence, a more sophisticated method is needed.

When we assess an existing blockchain, all timeouts are already known and implemented like in the case of the Cosmos blockchain. Now, let us assume that we are about to introduce a new blockchain within Cosmos ecosystem. For simplicity, it would share most of the parameters values with the Cosmos blockchain, however, Propose and Prevote timeouts would remain unknown. Moreover, suppose that we are already aware of the average processing time for Propose and Prevote steps, such that in case of non-homogeneous proposers, using (3) we define $t^*_{1_{FF}} \approx 1.14$s, $t^*_{1_F} = 3$s, and $t^*_{1_S} \approx 25.73$s. For homogeneous proposers, the times are $t^*_{1_h} = 3$s and $t^*_{2_h} = 1$s. Our goal in this scenario would be to find the optimal timeouts for corresponding steps $(T^*_1, T^*_{1_h}, T^*_{2_h})$, i.e. those that provide the best reachable network performance for given cases.

We use the same model as in the previous experiment, but this time we fix the proposers' speeds, such as $\frac{1}{t^*_{1_{FF}}}$, $\frac{1}{t^*_{1_F}}$, and $\frac{1}{t^*_{1_S}}$ for non-homogeneous proposers, and $\frac{1}{t^*_1}$, $\frac{1}{t^*_2}$ for homogeneous ones, for the Propose or Propose and Prevote steps together, respectively. The timeout rates are measured accordingly: $T^*_1 = \sigma t^*_{1_F}$ (non-homogeneous Propose step), $T^*_{1_h} = \sigma t^*_{1_h}$ and $T^*_{2_h} = \sigma t^*_{2_h}$ (homogeneous Propose step), such that $w^*_1 = w^*_{1_h} = w^*_2 = 1 - e^{-\sigma}$.

Fig. 3e shows the impact of different timeout for the Propose step on the network throughput with the non-homogeneous proposers. The black vertical line at $\sigma = 1$ represents the step timeout that equals to its actual processing time. It shows that the maximum network performance is reached at $\sigma_{max} = 2.1$. Thus, in this case the optimal timeouts would be $T^*_1 = t^*_{1_F}\sigma_{max} = 3 * 2.1 = 6.3$s.

Fig. 3f and 3g show network throughputs as functions of the changing coefficient σ applied to Prevote only and Propose and Prevote timeouts with the homogeneous proposers. The plots demonstrate that the throughput of the network increases as the timeouts are extended. Setting the timeouts equivalent to their actual times is not enough to approach the maximum throughput. This indicates that it is more effective for the system to allow the network additional time to reach an agreement within Prevote phase of the current round rather than starting a new round. Shorter timeouts compared to actual service times lead to poor network performance.

D. Effect of Round timeouts' augmentation

In our final experiment, we aim to observe the effect of varying timeout augmentations on subsequent blocks. Recall that Cosmos blockchain uses the standard timeout increase $g = 0.5$s for each step except Commit, resulting in a total increase of 1.5s per round (see Table II).

Fig. 3h and 3i illustrate the network throughput as functions of timeout increase that have range of $g \in (0, 1)$s for both homogeneous and non-homogeneous scenarios. They share the same parameters, i.e., we fix the actual times for the Propose and Prevote steps to $t^*_{1_{FF}}$, $t^*_{1_F}$, and $t^*_{1_S}$ for the non-homogeneous scenario as we did in sec IV-C and to $t_1 = 3$s in the homogeneous case. Contrary to the assumption that increasing the timeout would benefit by allowing more time for consensus, our models show that the highest throughput occurs when there are no step time increases. However, keeping a non-zero timeout increase might still be reasonable for real situations with prolonged processing times, despite the approximately 25% decrease in performance.

V. Conclusion

In this paper, we assess the operational dynamics of Cosmos blockchain ecosystem. The key takeaways encompass the following: first, we utilise the process algebra tool to introduce two analytical models of the consensus process for Cosmos ecosystem, w.r.t. the homogeneous and non-homogeneous proposers. Next, we study the multi-round performance in Cosmos blockchain that is a well-known network within the ecosystem. Additionally, we propose a method for determining optimal timeouts for the Propose step applicable to any instance driven by the protocol CometBFT. Our models are broadly applicable to any blockchain with the same protocol, offering valuable insights for improving efficiency in decentralised networks. Being our models formalized in a process algebra they allow further analysis in terms of behavioural equivalences and security properties (see, e.g., [18], [19]).

Additionally, we reveal that the Cosmos blockchains suffer from unfairness due to the skewed distribution of voting power among validators, with a high concentration of VP among top-ranked members. Mitigation solution for this and comparative analysis of practical validation of the proposed methods in real-world settings are subjects to future work.

Acknowledgment

This study was carried out within the PE0000014 - Security and Rights in the CyberSpace (SERICS) and received funding from the European Union Next-GenerationEU - National Recovery and Resilience Plan (NRRP) – MISSION 4 COMPONENT 2, INVESTIMENT 1.3 – CUP N. H73C22000890001. This work has been also partially supported by the Research Project INDAM GNCS 2024 - CUP E53C23001670001 "Modelli composizionali per l'analisi di sistemi reversivili distribuiti (MARVEL)" and by the Project PRIN 2020 20202FCJMH - CUP G23C22000400005 "Nirvana - Noninterference and Reversibility Analysis in Private Blockchains". This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: http://www.bitcoin.org/bitcoin.pdf

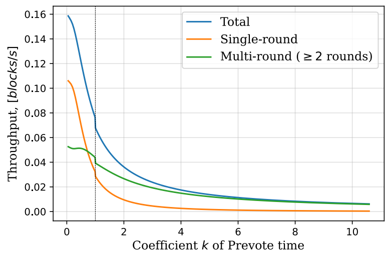
[2] "Inter-Blockchain Communication Protocol documentation," https://ibc.cosmos.network/main, accessed: 2024-04-15.

[3] "CometBFT Documentation," https://cometbft.com, accessed: 2024-04-15.

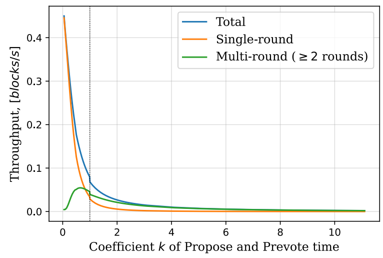
[4] E. Buchman, J. Kwon, and Z. Milosevic, "The latest gossip on BFT consensus," *arXiv preprint arXiv:1807.04938*, 2018.

[5] J. Hillston, *A Compositional Approach to Performance Modelling*. Cambridge University Press, 1996.

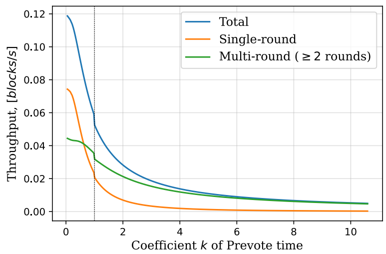
[6] C. Fan, S. Ghaemi, H. Khazaei, and P. Musilek, "Performance evaluation of blockchain systems: A systematic survey," *IEEE Access*, vol. 8, pp. 126 927–126 950, 2020.



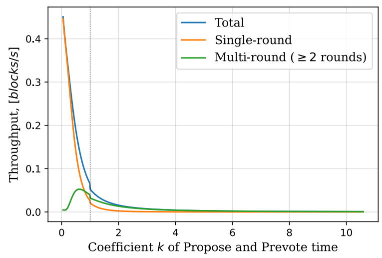
(a) Network throughput as a function of coefficient k for Prevote time with the non-homogeneous proposers.



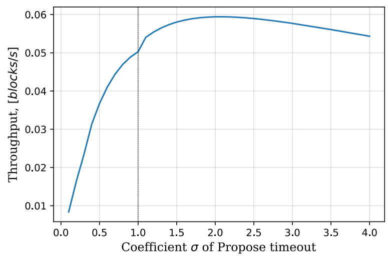
(b) Network throughput as a function of coefficient k for Propose and Prevote times with the non-homogeneous proposers.



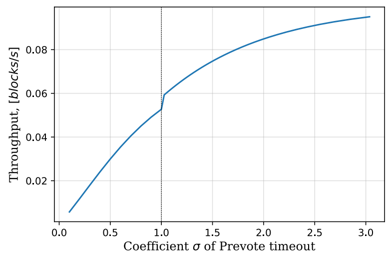
(c) Network throughput as a function of coefficient k for Prevote time fixed at $w_1 = 0.63$ with the homogeneous proposers.



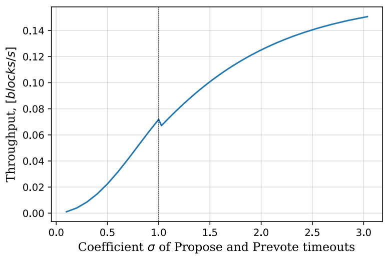
(d) Network throughput as a function of coefficient k for Propose and Prevote times with the homogeneous proposers.



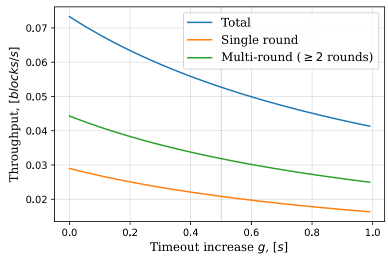
(e) Network throughput as a function of coefficient σ of Propose timeout with the non-homogeneous proposers.



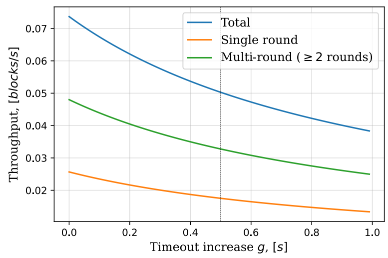
(f) Network throughput as a function of coefficient σ of Prevote timeout fixed at $w_1 = 0.63$ with the homogeneous proposers.



(g) Network throughput as a function of coefficient σ of Propose and Prevote timeouts with the homogeneous proposers.



(h) Network throughput as a function of different timeout increases for homogeneous proposers.



(i) Network throughput as a function of different timeout increases for non-homogeneous proposers.

Fig. 3: Numerical results of the experiments.

[7] Y. Supreet, P. Vasudev, H. Pavitra, M. Naravani, and D. Narayan, “Performance evaluation of consensus algorithms in private blockchain networks,” in *2020 International Conference on Advances in Computing, Communication & Materials (ICACCM)*. IEEE, 2020, pp. 449–453.

[8] O. Wu, S. Li, Y. Wang, H. Li, and H. Zhang, “Modeling cross-blockchain process using queueing theory: The case of Cosmos,” in *2022 IEEE 28th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, 2023, pp. 274–281.

[9] J. Han, J. Kim, A. Youn, J. Lee, Y. Chun, J. Woo, and J. W.-K. Hong, “Cos-CBDC: design and implementation of CBDC on cosmos blockchain,” in *2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, 2021, pp. 303–308.

[10] I. A. Qasse, M. Abu Talib, and Q. Nasir, “Inter blockchain communication: A survey,” in *Proceedings of the ArabWIC 6th Annual International Conference Research Track (ArabWIC)*, 2019, pp. 1–6.

[11] D. Smuseva, I. Malakhov, A. Marin, A. van Moorsel, and S. Rossi, “Verifier’s dilemma in Ethereum blockchain: A quantitative analysis,” in *International Conference on Quantitative Evaluation of Systems (QEST)*. Springer, 2022, pp. 317–336.

[12] D. Smuseva, I. Malakhov, A. Marin, and S. Rossi, “Crisis of trust: Analyzing the Verifier’s Dilemma in Ethereum’s Proof-of-Stake blockchain,” in *2023 IEEE International Conference on Blockchain (IEEE Blockchain)*. IEEE, 2023, pp. 332–339.

[13] “Cosmos Whitepaper,” https://cosmos.network/whitepaper, accessed: 2024-04-15.

[14] “Cosmos Network: Validator Overview,” https://hub.cosmos.network/validators/overview#introduction, accessed: 2024-04-15.

[15] “CometBFT: Consensus Algorithm,” https://docs.cometbft.com/main/spec/consensus/consensus, accessed: 2024-04-15.

[16] M. Castro and B. Liskov, “Practical byzantine fault tolerance,” in *Operating Systems’ Design and Implementation (OsDI)*, vol. 99, no. 1999, 1999, pp. 173–186.

[17] “Configuration of Tendermint protocol,” https://docs.tendermint.com/v0.34/tendermint-core/configuration.html, accessed: 2024-04-15.

[18] A. Marin, C. Piazza, and S. Rossi, “Proportional lumpability,” in *International Conference on Formal Modeling and Analysis of Timed Systems, FORMATS 2019*, ser. LNCS. Springer, 2019, pp. 265–281.

[19] J. Hillston, A. Marin, C. Piazza, and S. Rossi, “Persistent stochastic non-interference,” *Fundamenta Informaticae*, vol. 181, no. 1, pp. 1–35, 2021.