XinFin Network-XDC Consensus Algorithm

White Paper (Updated)

Updated on May 4th, 2021
(edited for grammar, readability, consistency)
XinFin (XDC) Hybrid Blockchain R&D Team

XinFin Hybrid Blockchain Network
Email: info@XinFin.org

ABSTRACT

In this paper, we present an overview of the architectural design of XinFin Network's consensus algorithm. XinFin Network is a public EVM (Ethereum Virtual Machine)-compatible blockchain with the following advantages: low transaction fees (near zero), low energy consumption, efficient confirmation time, double validation, and randomization for security guarantees. In particular, we propose XinFin's Delegated Proof of Stake (XDPoS) consensus, a Proof-of-Stake (XDPoS)-based blockchain protocol with rigorous security guarantees and fast finality. We also present a novel reward mechanism - demonstrating that, with this mechanism, the blockchain has a low probability of forks and fast confirmation times. Additionally, the contributions and benefits of masternodes are fair and equitable, in the sense that the probability distribution function is uniform, eventually. As we go into details, explaining the motivation and double validation process as well as the finality checkpoint of the protocol, we also present the formalization of the XinFin model in a mathematical manner, demonstrating the soundness of the XinFin model and protocol. Lastly, we conduct a security analysis to examine the protocol's ability to resist different attacks. Comparing the XinFin Network with several existing blockchains, we explain why XinFin Network's consensus algorithm offers a more secure protocol.

I. INTRODUCTION

A. Context

Today, many companies use blockchain to drive greater veracity and transparency across their digital information ecosystems. And, they are boosting awareness of the blockchain industry and its accompanying infrastructure in a variety of sectors—ranging from consumer-packaged goods (CPG) to cloud computing and storage, and from infrastructure to public policy.

Building upon these developments, blockchain has surpassed initial forecasts based on its promise within the banking and cryptocurrency arenas. Worth noting: Funding to blockchain-based companies, despite dipping from 2018's highs, more than doubled in 2020 as compared to 2017.[1] Moreover, forecasts predict that annual spending on blockchain solutions will exceed $16B in 2023—thanks to the increased adoption of blockchain-based ecosystems.

To add, 2020 has branded its mark in history as a year of crypto institutionalization, with large multinationals (e.g., JPMorgan, PayPal, DBS) starting to offer crypto-based services.

While blockchain was invented in 2008 by Satoshi Nakomoto, developments in blockchain ecosystems, today, have a striking semblance to the early years of the internet. During the inception years pioneers, dreamers, and early adopters came together to build the internet space we experience today.

In the blockchain space, many pioneers, dreamers, and early adopters have introduced ground-breaking projects. XinFin is among these novel organizations. The XinFin Network is a pioneer in today's establishment of blockchain ecosystem architecture. Leveraging the power of cryptographic tokens, XinFin interconnects an ecosystem of applications via a

[1] CB Insights. "58 Big Industries Blockchain Could Disrupt." CB Insights Research.

unique blockchain infrastructure that allows fast, frictionless and secure payment, and ensures reliable storage of value.

Powered by XDC Protocol, XinFin's XDPoS Hybrid Network is a highly interoperable blockchain network supporting global trade and finance. Thanks to the XDPoS' interoperability, the network permits digitization, tokenization, and swift settlement of trade transactions, increasing efficiency and reducing reliance on complex foreign exchange infrastructures. Operating as a blockchain agnostic middleware, the XinFin Network ensures flexibility in liquidity management by connecting MSME originators and decentralized liquidity pools.

Recently, XinFin received recognition from the World Trade Organization (WTO) for its hybrid protocol that supports permissionless ledgers for public verification and permissioned ledgers for restricted data sharing.[2]

B. XinFin's Efficient and Secured Protocol

Satoshi Nakamoto's blockchain protocol attempted to achieve consensus within a permissionless setting—that is, anyone can join or leave the protocol execution without seeking permission from a centralized or distributed authority. Additionally, the protocol's instructions weren't dependent on the players' identities, presenting a game-changing protocol architecture. Later, Ethereum and the Ethereum Virtual Machine (EVM), which were released in July 2015, proposed notable enhancements compared to the Bitcoin protocol. The Ethereum protocol introduced the smart contract functionality.

Whilst Bitcoin and Ethereum are game-changing technologies, they present a myriad of issues, especially with transaction processing performance. That said, to create an efficient and secured consensus protocol for the XinFin Network, the novel network tackles the following bottlenecks of classic blockchains.

- **Transaction In-Efficiency**

  Consortium blockchains, employed by leading cryptocurrencies like Bitcoin and Ethereum, experience a blockchain scaling problem. More specifically, they don't scale well to handle large transaction volumes. Put into context: the fixed block size in the Bitcoin blockchain and gas prices in Ethereum cap their transactions per second (TPS) to 7 and 15, respectively. This small throughput severely hinders the wide-spread adoption of such cryptocurrencies.

- **Confirmation Times**

  Network latency—the time required to generate an additional block of transaction in a chain or the time it takes for a transaction to appear on a blockchain—is a major issue of concern. For instance, Bitcoin's 10-minute block-time is significantly longer than the average network latency. To add, Bitcoin blocks require 5 subsequent blocks to ensure confirmation. Thus, it takes close to an hour for transactions to receive confirmation. On the other hand, Ethereum, which has lower latency, has a relatively

[2] Ganne Emmanuelle, and Deepesh Patel. "Blockchain & DLT In Trade: Where Do We Stand?" Trade Finance Global

high block-time, around 13 minutes. These long confirmation times hinder the wide scale adoption of these classic blockchains in many smart contract applications.

- **Fork Generation**

  Fork generation—whether hard or a soft fork—is time-consuming, creates potential vulnerabilities, and consumes significant computational energy. In regards to vulnerabilities, fork generation exposes blockchain networks to costly attacks like Peer-to-Peer Network-based, Smart Contract-based, Consensus & Ledger-based, and Wallet-based attacks. In the past, Ethereum Classic, Bitcoin gold, Feathercoin, Vertcoin, Grin, and Verge blockchain networks suffered 51% attacks. (A 51% attack refers to an attack on Proof-of-Work (PoW) blockchain where attackers gain control of 51% or more of a networks' hash rate). Given the vulnerabilities therein, private/permissioned blockchains are usually very resistant to possible forks of their blockchain.

- **High Energy Consumption**

  With classic blockchain networks, the Proof-of-Work (POW) consensus mechanism requires mining (computational power) to do proof-of-work computations. These transactions consume alarmingly large amounts of energy. To clarify, the current annual estimated energy consumption of Bitcoin mining activities is 87.17 terawatt-hours (TWh)[3] while the energy consumption of Ethereum is 29.41 TWh.[4] Put into further perspective, as of 2017, Bitcoin mining activities consumed energy levels equivalent to Denmark's energy consumption. With projections indicating that energy consumption in Bitcoin mining will soar by the year 2022, the POW consensus remains unsustainable.

- **Anonymous Network Node**

  A transaction hash is a unique string of code that's given to transactions that have been verified and added to a blockchain. With blockchain nodes playing an important part in transaction hashing, the anonymity of nodes—which store copies of the distributed ledger and maintain the reliability of the stored data—poses a problem. With governments seeking control of sensitive transactions, lack of government control, lack of regulatory authority, and maintaining pseudo-anonymity is a challenge. Notably, anonymous transactions may lead to misuse of blockchain technology, undermining government and regulatory activities.

Motivated by the above-mentioned challenges, the XinFin Network proposes a consensus protocol that focuses on the following key strategies:

- Double Validation to strengthen security and reduce likelihood of forks.
- Randomization to guarantee fairness and prevent handshaking attacks.
- Fast confirmation time and efficient checkpoints for finality or rebase.
- Self-KYC layer while setting up Network Node.

To start dealing with these problems, this whitepaper presents an overview of the architectural design of XinFin Network's masternodes. In particular, the paper proposes XinFin Delegated Proof of Stake (XDPoS) consensus, a Proof-of-Stake (PoS)-based blockchain protocol with rigorous security guarantees and fast finality. The white paper also

[3] Digiconomist "Bitcoin Energy Consumption Index."
[4] Digiconomist "Ethereum Energy Consumption Index (Beta)."

presents a novel reward mechanism and demonstrates that with the POS mechanism, the blockchain can provide efficient confirmation times, and a low probability of forks. Additionally, the contributions and benefits of masternodes are equitable, in regards to the eventual uniformity of the probability distribution function.

C. Structure of The Paper.

Section II-A Goes over the architectural design of the XinFin Network's masternodes, framework, and background protocols that help mass readers (e.g., investors, traders, participants) absent the technical education to understand the network's masternodes.

Section II-B presents the XinFin masternode stakeholder policy, masternode committee, and reward and slashing mechanism.

Section II-C explains the motivation and double validation process as well as the finality checkpoint of the protocol.

Section II-D presents the formalization of XinFin Network's architecture in a mathematical model to show the soundness of the model and the protocol.

Section III discusses the security analysis and the systems resistance to potential attacks.

Section IV discusses and compares the XinFin Network with several existing blockchains.

Section V concludes the paper.

II. XINFIN NETWORK Masternode DESIGN

A. The XinFin Network Architecture

The XinFin Network consensus protocol or XinFin Delegated Proof of Stake (XDPOS) Consensus shown in Fig. 1 below, regulates the consistent production and maintenance of masternodes in the XinFin Blockchain.

Masternode Parameters

Suggested parameter values from requirements:

MIN_STAKE: 10000000 XDC

VALIDATOR_REWARD: NOS_masternode / TOTAL_REWARD

REWARDS_TRANSFER: Every next block of epoch

VALIDATOR_SET_SIZE: 108

WITHDRAWAL_PERIOD: Set of Epoch (1 Epoch == 900 Blocks)

(Worth noting, there's a proposal to increase the VALIDATOR_SET size to 144 after approval from the governance committee.)

Each XinFin masternode is a full node that holds XDC. For coin-holders to operate a masternode, 8 (eight) key requirements must be satisfied. These include:

- More than 10,000,000 XDC held by the new masternode holder, helping them perform random delegated proof of stake consensus, seamlessly.
- A suitable wallet to store XDC tokens. Preferably in hardware form.
- A dedicated and stable hardware environment.
- A dedicated Static Public IP address.
- 100% network uptime by IDC network.
- A minimum of tier 3+ IDC environment.
- Virtual Private Server (VPS). Though optional, this option is highly recommended.
- When using cloud-based services like Amazon EC2 M3, large virtual machine (VM) sizes are appropriate. Similar configurations are applicable for the Microsoft Azure Cloud network users.

Given that the XinFin masternode is a full node, it stores a copy of the blockchain, produces blocks and keeps the chain consistent. These nodes are controlled by consortium members and come with a number of caveats. As previously noted, full nodes must purchase (and hold) a fixed equity of XDC (more than 10,000,000) to be able to host the full XDC protocol. This design introduces the advantage that no Full Node or groups of Full Nodes have control over the network. For node holders to possess more than 51% hashing power, a full node must acquire more and more XDC. With increased demand, XDC prices rise, making it financially impossible for a Full Node cartel to control the XinFin Network.

The XinFin Network also employs Double Validation complemented with a Randomization mechanism to prevent unscrupulous characters from gaining control of the network. With these techniques, the network reduces the probability of having invalid blocks in the blockchain, ultimately reducing its effectiveness. These enhancements and the components of the XinFin Network are explained in subsequent sections.

B. Stakeholders.

Coin Holders, Masternodes

Coin-holders are as simple as the name suggests: users who join the network and who own and transfer the required amount of XDC. It's worth noting that the XinFin Network doesn’t have miners as is with Proof-of-Work-based blockchain systems like Bitcoin and Ethereum. The XinFin Network employs a Proof-of-Stake (PoS)-based protocol.

On the XDPoS, only masternodes can produce and validate blocks. Once coin holders deposit 10 million XDC to the Smart Contract, they are listed as masternode candidates in the DApp. Masternodes that work consistently within the system creating and verifying blocks are incentivized with XDC. XinFin Network engineers take responsibility for designing this fair, explicit, automated, and accountable reward mechanism.

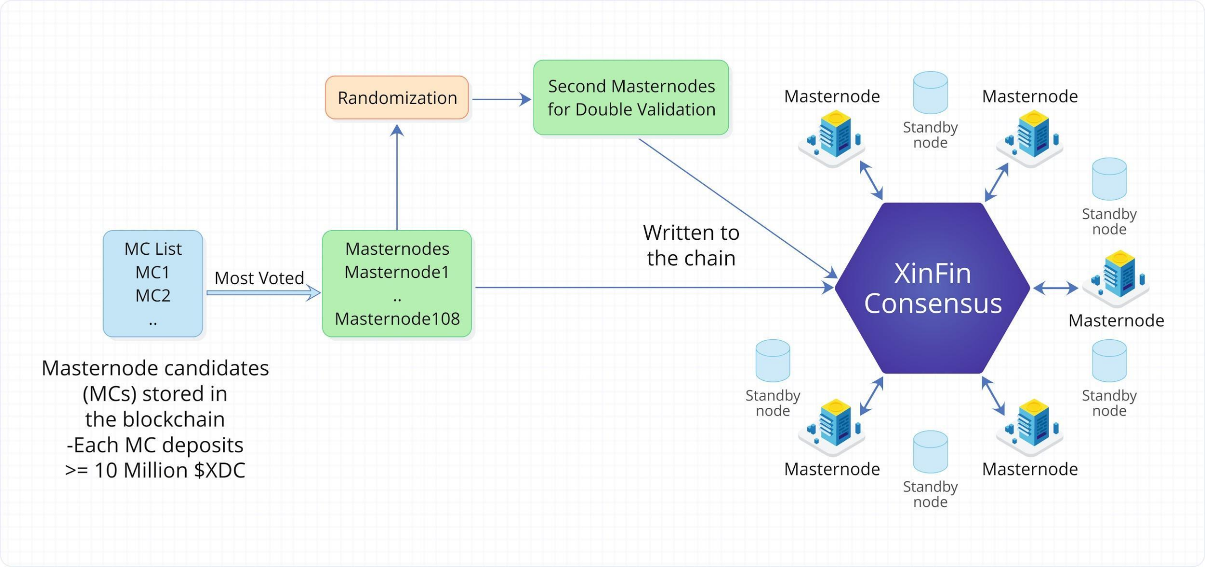


Fig. 1. XinFin Network Architecture

Since 108 masternodes are the maximum number in the masternode committee, masternode holders must deposit 10 million XDC to be considered for positions in the masternode committee. The amount (10,000,000 XDC) is locked in a smart contract. In the event that a masternode is demoted or intentionally resigns from the masternode, the candidate's deposits are locked for 30 days and can be accessed thereafter.

Reward Mechanism

For each iteration of 900 blocks (called an epoch), a checkpoint block is created, which implements only reward works. The checkpoint block is referred to as the block signer. Tasked with storing all block signatures, block signers count the number of signatures sent to the block signer smart contract during the epoch. Rewards are based on the number of signatures linked to a masternode in an epoch.

In addition, block creators are selected in a circular and sequential order, allowing each masternode holder an equal opportunity to create and sign a block. In XDPoS's current implementation, failure of a masternode to create a block causes a 10-second delay before the next masternode in the sequence takes its turn to create the next block.

Further, there is a reward-sharing ratio among coin-holders and masternodes that have been elected via the support of coin-holders. Specifically, each epoch consists of 900 blocks, which receive a 250 XDC reward in the first 2 (two) years. The 250 XDC reward is divided among masternodes based on the number of signatures associated with the node in each epoch. Thereafter, masternode rewards are divided into three portions, namely:

- **Infrastructure Reward:** This reward comprises the first portion of 40%. The reward goes to the masternode.

- **Staking Reward:** This reward accounts for 50% of the reward. The reward is shared proportionately amongst the pool of voters for a specific masternode. Token stake is the criteria used to share the reward amongst voters.

- **Foundation Reward:** The foundation reward accounts for the last 10%. This reward is channelled into a special account that's controlled by the masternode foundation. The foundation is initially run by the XinFin Network founding company.

Worth noting, coin-holders that un-vote prior to the check-out block don’t receive any rewards in the staking reward portion.

Slashing Mechanism

Slashing is an inbuilt mechanism for proof of stake blockchain protocols that seeks to discourage validator misbehaviour. Slashing incentivises node security, availability, and network participation.

On the XDPoS, underperforming masternodes dramatically decrease the performance capabilities of the entire network and cause instabilities. In XinFin’s Hybrid Network, failure by a masternode to create a block causes a 10 second delay. If a masternode has several turns (say ***M*** times) to produce a block within an epoch, the masternode causes more delays (that is ***M*10 sec***). To mitigate the aforementioned problem, adoption of the slashing mechanism is key. XinFin’s slashing mechanism entails as follows:

First, if a masternode fails to create a block during an epoch, the masternode is slashed for the 4 (four) subsequent epochs. Second, if more than one masternodes underperform, several changes occur in the treatment of the masternodes list for the next epoch. To specify, a masternode that’s considered to have underperformed within the past 4 epochs is considered as “kicked-out” and it loses the right to create blocks. At such times, the number of masternodes taxed with creating blocks falls below 108. At the same time, active masternodes do not wait the 10 seconds for the underperforming masternodes.

Once kicked out of XinFin’s masternode list, underperforming masternodes can still verify and sign blocks. This caveat is used to enable underperforming masternodes to notify other masternodes of their liveness. Nonetheless, underperforming masternodes don't receive rewards for verifying and signing off blocks after being slashed out.

To make the slashing process more effective, the XinFin Network employs both off-chain and on-chain slashing mechanisms. With the off-chain slashing, detection of misbehaviour is easier to implement, and it can be used for crafty characters. Within the contract is a ***reportBenign*** method (part of the Validator Set Contract) which only Validators can call, pass a message and a block-number. Slashing is executed if more than 2/3 of the Validators agree on the misbehaviour. Misbehaviours might include: validators consistently propagating blocks late or validators being offline for more than 24 hours. The slashing can be executed on portion of the stake—say 4%.

For the on-chain slashing, the process occurs when a validator signs-off two blocks with the same step—a condition known as equivocation. Once a validator node enters the wrong KYC detail, the contract includes a ***reportMalicious*** method. With the reportMalicious method only Validators can call, pass a message and a block-number. If more than 2/3 of the validators agree on the ***reportMalicious***, a slashing will be executed. The process can slash a portion of the entire 100% stake of a Validator Node.

C. XinFin Network Consensus Protocol

Motivation for XinFin’s Double Validation

On the XDPoS, Double Validation is motivated by the need for an additional trust-less validation layer that enhances security via a provable uniform distribution. The trust-less validation layer is achieved through decentralized randomization. To clarify, once a masternode creates a block, and before it's added to XinFin's blockchain, it's mandatory that such a masternode is verified by a randomly selected set of masternodes.

Leveraging Double Validation, XinFin's hybrid protocol strengthens the network, improves XDPoS security, reduces possibilities for nothing-at-stake and fork attacks, and makes the hybrid blockchain unique among Delegated Proof-of-Stake-based blockchains.

Double Validation Process

Within the XinFin Network, masternodes have equal responsibility in running the network and keeping it stable. The network's Full nodes (in this case masternodes) run on powerful hardware configurations and high-speed network connectivity, ensuring the required block time (estimated to be two seconds).

Moreover, only masternodes can produce and seal blocks. For blocks to be produced and sealed, the XinFin Network consensus relies on the concept of Double Validation to improve the reliability of existing consensus mechanisms, namely Single Validation.

In the next section, we will cover XinFin's Double Validation process, then analyse the differences and improvements of Double Validation compared to Single Validation.

XinFin's Double Validation (DV) is similar to some existing PoS-based blockchains such as Cardano. In the process, each block is created by a block producer. A block producer is primarily a masternode that takes its block creation permission turn following a pre-determined and circular sequence of masternodes in each epoch.[5]

However, the DV process in the XinFin Network differs from the process in other blockchains. It's worth noting that DV in the XinFin Network requires the signatures of two masternodes on a block for the block to be published on its blockchain. The block creator (first masternode) provides the first signature. The second masternode, which is the block verifier, is randomly selected among a set of voted masternodes to verify and sign a new block.

In subsequent sections—for a more convenient read—the terms 'block creator' and 'block verifier' are used interchangeably for masternode 1 (block producer) and the randomly selected masternode 2, respectively. The process of randomly selecting the block verifiers is detailed in the next paragraphs. Additionally, there is no mining in the block creation as demonstrated in Proof-of-Work-based blockchains (e.g., Ethereum and Bitcoin). This means that a created block is valid if, and only if, it is sealed by two signatures from a block creator and a corresponding block verifier that confirms its accuracy.

At XinFin, there is a strong belief that the unique DV technique enhances the stability of the blockchain by diminishing the probability of producing 'garbage' blocks, while simultaneously maintaining the system security and consistency. Randomization of block verifiers in DV is a key factor in reducing risks associated with paired masternodes trying to commit malicious blocks.

Additionally, comparing XinFin's blockchain architecture with other blockchain setups indicates that there are significant advantages to the unique DV technique. Specifically, DV introduces improvements in block time by only requiring two signatures per block. In the interest of demonstrating how the XinFin Network has undergone enhancements, compared

[5] Hoskinson, Charles. "Why We Are Building Cardano --a Subjective Approach."

to existing PoS-based blockchains, the following figures analyze the differences between Double Validation and Single Validation mechanisms. The limitations in some existing blockchains serve to highlight the improvement introduced by Double Validation over Single Validation.

In Fig. 2 and Fig. 3 below, benefits introduced by DV as compared to SV are compared by analyzing some attack scenarios.

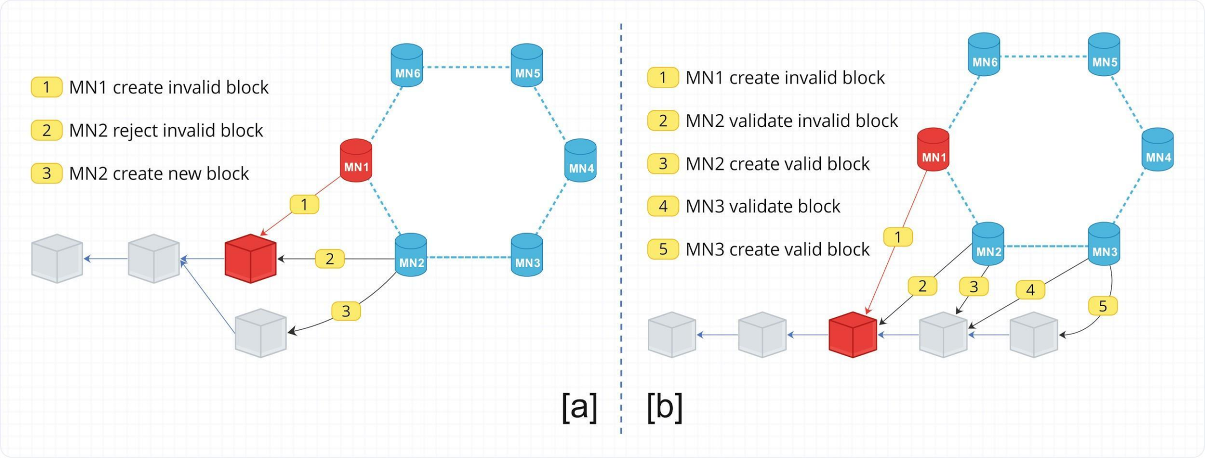


Fig. 2. Single Validation (SV): (a) SV with block creation masternode as an attacker and (b) SV with two consecutive block creation masternodes as attackers

- **Single Validation**

In Single Validation and in an epoch, each masternode (e.g., M1 in figure 2.), sequentially takes its turn to create a block (e.g., block100). The next masternode (e.g., M2) in the sequence then validates the new block100. In the event that block100 is invalid (implying that M1 is potentially an attacker) and that the new block contains a transaction that invalidly benefits M1, an honest M2 rejects block 100 and creates a valid block 100 next to block 99 (see Fig. 2 [a].)

However, if M2 is also an attacker (see Fig. 2 [b]) that cooperates with M1, M2 ignores the invalidation of block100, signs it, and creates the next block, namely block101 that is valid. Subsequently, if masternode M3 verifies the validity of block101, M3 signs block101 and creates a block102. This way, Single Validation potentially leaves the blockchain with ‘garbage’ or invalid blocks which require a ‘rebase’ to restore the validity of the blockchain.

- **Double Validation (DV)**

With XinFin's DV technique, the likelihood of having garbage blocks in the blockchain is significantly reduced. In this case let’s assume that M1 and M2 are the block creator and block verifier of block100, respectively.

If block100 is invalid and M2 is honest (see Fig. 3 [a]), the next block creator M3, when creating block101, notes that block100 doesn't have the required number of signatures (two signatures in XinFin’s case), and thus, rejects block100 and creates another block100 next to block99. In the event that M2 is also an attacker pairing/handshaking with M1 (see Fig. 3 [b]), M2 signs block100 despite its invalidity. Worth noting is that XinFin’s block verifiers or M2 are randomly selected, therefore there is little chance of successfully pairing M1 and M2. This limits the possibility of invalid blocks being added to the blockchain.

Next, even after M3 verifies that block100 has two valid signatures, M3 still rejects it because block100 is invalidated by M3 that will create another valid block100. To break the stability and consistency of XinFin's blockchain, in this case, M3 should be an attacker together with M1 and M2. This scenario has a low probability of occurring since block verifiers are randomly selected. That said, DV strengthens the consistency of the blockchain and makes it hard to disrupt.

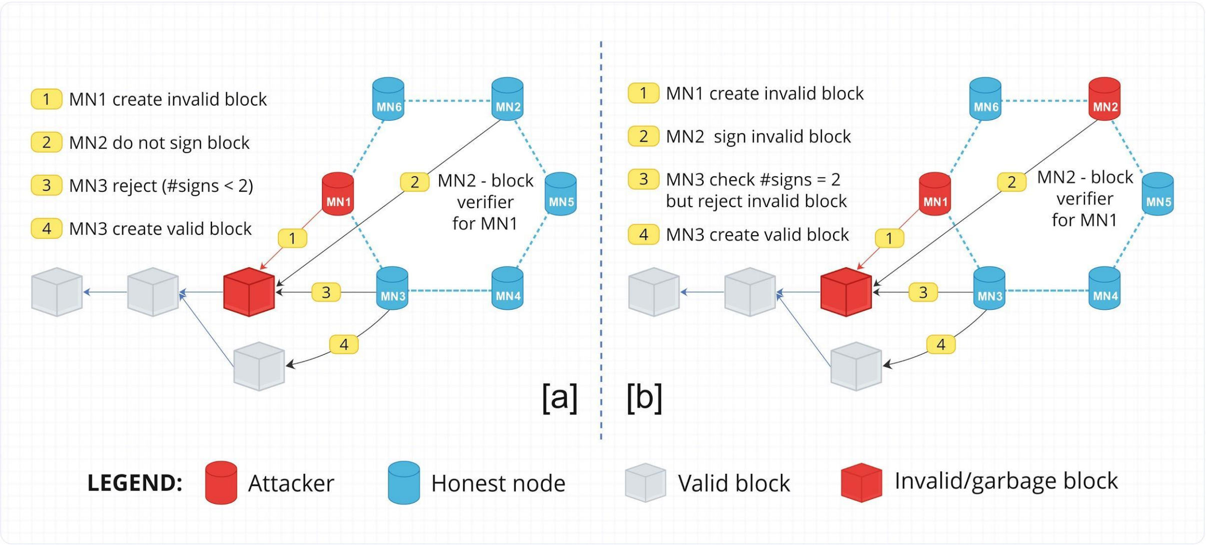


Fig. 3. Double Validation (DV): (a) DV with block creator as an attacker and (b) DV with both block creator and block verifiers as attackers

Randomization for Block Verifiers for Double Validation

The First masternode or Block Creator: Within a given epoch, the first masternode/block creator(v_1) is selected by a round-turn game and it can be formally defined as an array.[6] The array's formula is as follows:

$$\begin{bmatrix} v_1 \end{bmatrix} = \begin{bmatrix} v^{e}_{1.1} \\ v^{e}_{1.2} \\ . \\ . \\ . \\ v^{e}_{1.n-1} \\ v^{e}_{1.n} \end{bmatrix}$$

(equation: 1)

Random Matrix and Smart Contract

To select random verifiers for a subsequent epoch (***e+1***), 3 (three) steps are followed. To understand the three steps explained below, let ***m*** be the number of masternodes, ***n*** be the number of slots in an epoch.

[6] EMPTY

Step 1: Random Numbers Generation and Commitment Phase

First, at the beginning of epoch ***e***, each masternode ***Vi*** creates an array of ***n + 1*** special random numbers ***Recommendi* = [ri.1, ri.2, ..., ri. n, θi]**, where **ri. k ∈ [1, ..., m]** indicates the recommendation of an ordered list of block verifiers for the next epoch of ***Vi***, and **θi ∈ {−1, 0, 1}** is used for increasing the unpredictability of the random numbers.

Second, each masternode ***Vi*** has to encrypt the array ***Recommendi*** using a secret key SKi, say **Secreti = Encrypt (*Recommendi*, SKi)** as the encrypted array. Next, each masternode forms a “lock” message that contains encrypted array Secreti, signs off this message with its blockchain’s private key through the Elliptic Curve Digital Signature Algorithm (ECDSA) scheme that’s currently used in Ethereum and Bitcoin along with the corresponding epoch index and its public key generated from its private key. After forming a “lock” message and signing off the message via the ECDSA verifiable key, every masternode can check who created this lock message through ECDSA verification scheme and which epoch it relates to. Thereafter, each node ***Vi*** sends their lock message with its signature and public key to a Smart Contract stored in the blockchain. The process enables each masternode to collect and know the locks from all other masternodes.[7]

Step 2: Recovery Phase

The recovery phase is for every node to reveal its previous lock message so other nodes can get to know the secret array it has sent before. A masternode only starts revealing its lock message if all masternodes have sent their lock messages to the smart contract or a certain timeout event occurs. Each masternode then opens its lock message by sending an “unlock” message to the smart contract for other masternodes to open the corresponding lock. Let’s imagine a commitment-like scheme, in this case, where a lock message is a commitment message locking its contained recommendation array ***Recommendi*** so that no one can open or guess the contained array, and the unlock message gives the key for other masternodes to decrypt the box and retrieve the values of ***Recommendi***. Eventually, a masternode has both locks and unlocks to other masternodes. If some elector is an adversary which might publish its lock but not intend to send the corresponding unlock, other masternodes can ignore the adversary’s lock and set all its random values be 1, by default. The idea is simple: the network can keep working successfully even if some masternodes are adversaries.

Step 3: Assembled Matrix and Computation Phase

At the point of the slot n^{th} of the epoch ***e***, the secret arrays Secreti in the smart contract will be decrypted by each masternode and return the plain version of Recommendi. Each tuple of the first n numbers of each ***Vi*** will be assembled as the i^{th} column of an ***n × m*** matrix. All the last number ***θi*** forms a ***m × 1*** matrix. Then each node will compute the block verifiers ordered list by some mathematical operations as explained below. The resulting output is a matrix ***n × 1*** indicating the order of block verifiers for the next epoch ***e + 1***.

For the second masternode or block verifier, each node computes the common array v_2 for the order of the block verifiers by the following steps as in Equation 1.

[7]

$$\begin{bmatrix} v_2' \end{bmatrix} = \begin{bmatrix} v_{2.1}^{e+1} \\ v_{2.2}^{e+1} \\ \vdots \\ v_{2.n}^{e+1} \end{bmatrix} = \begin{bmatrix} r_{1.1} & r_{2.1} & \cdots & r_{m.1} \\ r_{1.2} & r_{2.2} & \ddots & \vdots \\ r_{1.3} & \ddots & \ddots & r_{m.3} \\ \vdots & & r_{m—1.n—1} & r_{m.n—1} \\ r_{1.n} & \cdots & r_{m—1.n} & r_{m.n} \end{bmatrix} \begin{bmatrix} o_1 \\ o_2 \\ o_3 \\ \vdots \\ o_m \end{bmatrix}$$

(***equation: 2***)

$$\begin{bmatrix} v_2 \end{bmatrix} = \begin{bmatrix} v_2' \quad mod \quad m \end{bmatrix} = \begin{bmatrix} \left| v_{2.1}^{e+1} \right| & mod & m \\ \left| v_{2.2}^{e+1} \right| & mod & m \\ \vdots & & \\ \left| v_{2.n}^{e+1} \right| & mod & m \end{bmatrix}$$

(***equation: 3***)

Then, $\mathbf{v_2}$ is obtained by modulo operation of element values of v 0 2 as in Equation 2:

Finality Analysis

A standard definition of “total economic finality”: A phenomenon occurring when ¾ (three quarters) of all masternodes make maximum-odds bets that a given block or state will be finalized. This condition offers very strong incentives for masternodes to never attempt colluding to revert a block. When masternodes make such a maximum odds bet, in any blockchain where that block or state is not present, the masternodes lose their entire deposit.[8]

XinFin Network maintains this standardization in the design so that one block is considered as irreversible if it collects up to three-quarters of the signatures of all members in the masternodes committee. The time-line of the blockchain creation process, checking finality, and marking the block as immutable are described in Figure 4 below.

[8]

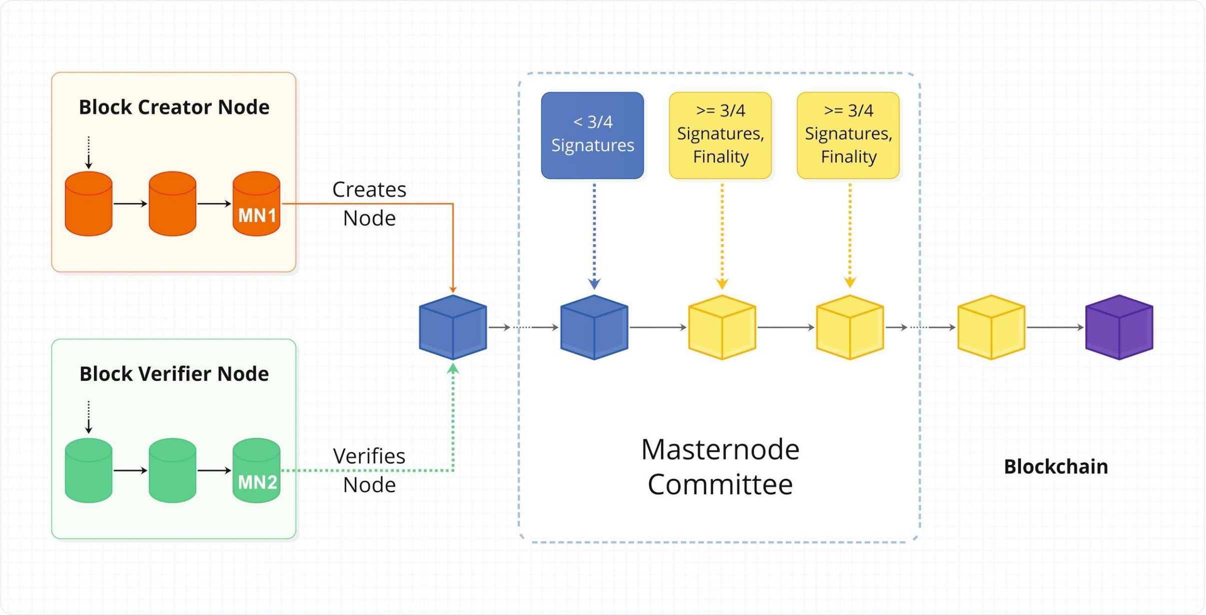


Fig. 4. Timeline of Blockchain Making Process

D. Consensus Protocol: Formalization

Basic Concepts & Protocol Description

To provide a solid educational foundation and to prove that the XinFin Network can achieve its claims, in the following section, we will present a preliminary examination of the concepts discussed in our yellow paper and an overview of XinFin Delegated Proof of Stake (XDPoS). To start, we will provide a presentation of XinFin's proof of stake consensus algorithm. The formalization follows that of notable tokens, such as Cardano and Thunder, in recent literature. More specifically, XinFin places emphasis on the following concepts and definitions that were presented in literature for Cardano and Thunder tokens and adapts them to the context of XinFin Network.

Time, Slots, Epoch

Ideally, each epoch is divided into 900 block times. Each of these block times is referred to as a block slot. Only one block can be created in a slot. The main assumption is that there is a roughly synchronized clock that allows for masternodes to learn the current slot. This simplification effectively permits masternodes to execute the signing and validation process of the XDPoS consensus, where each masternode must collectively create a block to the current slot. Simplified further, each slot SLR is accessed by an integer $\mathbf{r} \in \mathbf{\{1, 2, ...\}}$, and is supposed that the real-time window that corresponds to each slot has the following properties, which are similar to what is specified in Cardano.[6]

A. Every masternode can determine the index of the current slot based on the current time. And, any discrepancies between parties' local time are insignificant in comparison with the length of time represented by a slot.

B. The amount of a slot time is sufficient to guarantee that any message transmitted by an honest party at the beginning of the time window will be received by any other honest party by the end of that time window. While similar to Cardono's assumption, the XinFin Network adopts the assumption to ensure that block creators seamlessly propagate their created blocks to the corresponding block verifiers. This guarantees a block is signed by both the masternodes before the next block creator builds another block on top of it.

As mentioned in Section II-A, in XinFin's setting, it's assumed that the fixed set of ***m*** (150) masternodes **V1, V2, ..., Vm** interacts throughout the protocol to reach the consensus. For each ***Vi***, a public/private key pair **(PKI, ski)** for a prescribed signature scheme, ideally ECDSA, is generated.

Additionally, XinFin's protocol adopts the assumption that the public keys **pk_1, ..., pk_m** of the masternodes are distributed and known by all masternodes in the protocol (that means a masternode knows all public keys of all other nodes). Some notable definitions of the blockchain concepts are defined following the notation.

Definition 1 (State):

A state is an encoded string **st ∈ {0, 1}**

Definition 2 (Block):

A block B generated at a slot sli contains the current state **st ∈ {0, 1} λ, data d ∈ {0, 1}** ∗, the slot number ***i*** and a signature **Σ = Signski (st, d, sli)** computed under ski corresponding to the masternode Vi generating the block

Algorithm 1: The algorithm illustrated the consensus protocol

Input: ***m*** - Number of masternodes, ***n*** number of slots in an epoch
Output: The complete ledger of the blockchain C

To create the complete ledger for block C, several steps must be completed. These are as follows: **(a)** Creating the empty blockchain (stack) C **(b)** Commencing an Initial Coin Offering (ICO) to raise funds to support the provision of cryptocurrencies and blockchain-related products and services **(c)** Issuance of tokens/coins to holders. These tokens do not provide equity stake, rather they deliver their owners some stake in a product or service created by the company and **(d)** Voting for the masternode committee (masternodes) **VC** ← **{V1; V2; ..., Vm}**.

Thereafter, **(e)** Initiate the first epoch **e_1** ← **{sl_1, sl_2, ..., sl_n}**;**(f)** Randomly generate the array of second masternodes for the first epoch **SV1** ← **[$v_{1\,2.1}$, $v_{1\,2.2}$, ..., $v_{1\,2.\,n}$]**; **(g)** Create the genesis block B_0; **(h)** Update the blockchain **C** ← **C. push(B_0)**; while true do while j is less than n to create block B_j by the first masternode; Update the blockchain C ← **C. push (B_j)**;

Then, step **(i)** validate the block Bj by the second masternode; **(j)** broadcast and validate the block B_j by VCi; if Bj has more than 3/4 masternode committee members' signature then FINALITY(Bj .ID) = true; if j = n then j ← 1; else j++; if len(C) mod n = 0 then doCheckpoint(); Voting for the masternode committee for the next epoch **VC** ← **{V1; V2; ..., Vm}**; Random generate the array of verifier masternodes for the next epoch **(i + 1)th; SVi+1** ← **[v i+1 2.1 , vi+1 2.2 , ..., vi+1 2.n]; ei+1** ← **i ∗ n ∗ 2 + e1; i++**;

Here's a pictorial summary of the process:

Algorithm 1: Algorithm illustrated the consensus protocol

```
Input: $m$ - Number of masternodes, $n$ number of slots in an epoch
Output: The ledger of the blockchain $C$
begin
    Create the empty blockchain (stack) $C$;
    Initiate ICO; coinholders;
    Voting for the masternode committee (master nodes) $VC \leftarrow \{V_1; V_2; ..., V_m\}$;
    Initiate the first epoch $e_1 \leftarrow \{sl_1, sl_2, ..., sl_n\}$;
    Randomly generate the array of second masternodes for the first epoch
    $SV_1 \leftarrow [v^1_{2.1}, v^1_{2.2}, ..., v^1_{2.n}]$;
    Create the genesis block $B_0$;
    Update the blockchain $C \leftarrow C.push(B_0)$;
    while true do
        while j is less than n do
            Create block $B_j$ by the first masternode;
            Update the blockchain $C \leftarrow C.push(B_j)$;
            Validate the block $B_j$ by the second masternode;
            Broadcast and validate the block $B_j$ by $Vc_i$;
            if $B_j$ has more than 3/4 masternode committee members sign then
                FINALITY($B_j.ID$) = true;
            if j = n then
                $j \leftarrow 1$;
            else
                $j$++;
        if len(C) mod $n$ = 0 then
            doCheckpoint();
            Voting for the masternode committee for the next epoch $VC \leftarrow \{V_1; V_2; ..., V_m\}$;
            Random generate the array of verifier masternodes for the next epoch $(i + 1)^{th}$;
            $SV_{1+1} \leftarrow [v^{i+1}_{2.1}, v^{i+1}_{2.2}, ..., v^{i+1}_{2.n}]$;
            $e_i+1 \leftarrow i * n * 2 + e_1$;
            i++;
```

Definition 3 (Blockchain):

A blockchain C is a sequence of blocks $\mathbf{B_1, ..., B_n}$ associated with a strictly increasing sequence of slots for which the state sti of $\mathbf{B_i}$ is equal to H(Bi−1), where H is a collision-resistant cryptography hash function. To add, a blockchain has a number of properties, including the length of a chain len(C) = n, which is its number of blocks, and the block B_n is the head of the chain, denoted head(C).

As mentioned earlier, in the XinFin Network model, each time slot ***sli*** is set as 2 seconds and an epoch is a set as ***R*** of 900 slots $\{sl_1, sl_2, ..., sl_{900}\}$. The duration of an epoch equals 1800 seconds. In summary, the consensus protocol of XinFin Network consensus can be formalized in Algorithm 1. Algorithm 1 is simulated and explained as a process shown in Fig. 5

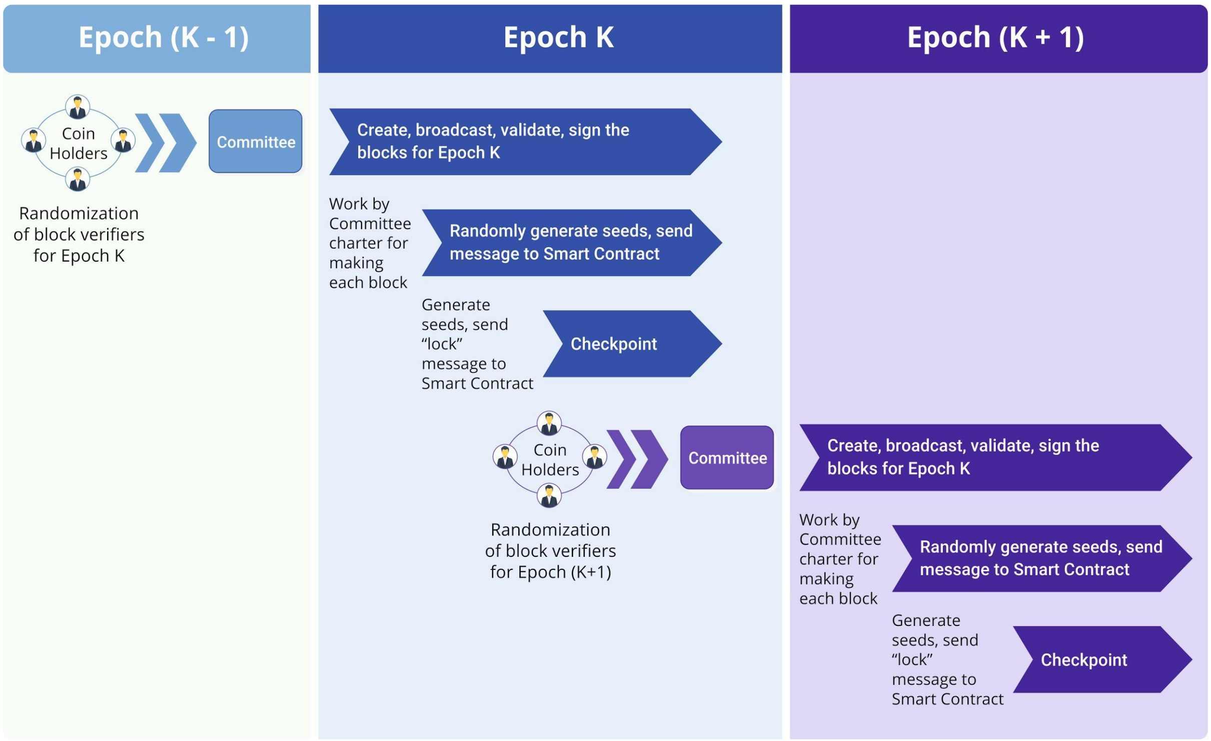

Epoch (K - 1)
Epoch K
Epoch (K + 1)
Coin Holders
Committee
Randomization of block verifiers for Epoch K
Create, broadcast, validate, sign the blocks for Epoch K
Work by Committee charter for making each block
Randomly generate seeds, send message to Smart Contract
Generate seeds, send "lock" message to Smart Contract
Checkpoint
Coin Holders
Committee
Randomization of block verifiers for Epoch (K+1)
Create, broadcast, validate, sign the blocks for Epoch K
Work by Committee charter for making each block
Randomly generate seeds, send message to Smart Contract
Generate seeds, send "lock" message to Smart Contract
Checkpoint

Fig. 5. Randomization of Block Verifiers, Creating and Validating Blocks in Each Epoch

III. SECURITY ANALYSIS

A. Nothing-at-stake

Nothing-at-stake is a well-known problem in PoS-based blockchain, just like the 51% attack in PoW algorithms. For PoW-based miners, it's mandatory to have CapEx (capital expenditures) for buying mining equipment such as ASICs. Similarly, there's a need for OpEx (operation expenditures) such as electricity to solve mathematical puzzles securing the network. That means, there is always an intrinsic cost for miners in mining regardless of its success. In case of a fork, miners therefore always allocate their resource (equipment) to the chain that they believe is correct in order to get incentives for compensating the intrinsic costs in mining.

On the contrary, PoS-based systems don't rely on mining. During an ideal execution creating a fork, the only costs incurred relate to block validation and signing. That is because masternodes do not incur intrinsic costs. In this case, there's an inherent problem of the masternode having no downside to staking both forks. Therefore, there are actually two issues in the original design of PoS. On one hand, for any masternode, the optimal strategy is to validate every chain/fork, so that the masternode receives its rewards no matter which fork wins. On the other hand, for attackers/malicious masternodes, they can easily create a fork for double spending.

The XinFin Network handles these two problems exceptionally. (Note: Through the XinFin Network consensus protocol, the XinFin Network maintains a certain order of masternodes in creating and sealing blocks during each epoch).

For the first issue, random/arbitrary forks never happen because block creation by the masternodes is predetermined in each epoch. For the second issue, the Double Validation mechanism ensures that only one block can be validated by the second randomly selected masternode. That's even when one malicious masternode creates two blocks at its turn.

B. Long-range attack

Within the XinFin Network, a block is valid only if it collects Double Validation and is finalized once 3/4 of masternodes verify. Therefore, as long as the number of attackers or malicious nodes and/or fail-stop nodes is less than or equal to 1/4 the number of masternodes, the number of masternodes signing a block is at least 3/4 the total number of masternodes, which makes the block finalized.

Thus, there is no chance for one malicious masternode to create a longer valid chain on the XinFin Network because other masternodes will reject the new block.

C. Censorship Attack

In the event that there are more than 3/4 malicious masternodes in the XinFin Network, censorship attacks may occur. For example, if the malicious masternodes refuse valid blocks or simply become inactive, the chain is stuck. To avoid censorship attacks, masternodes are paid for their effort of correctly working so that the chain is actively updated in a consistent manner.

More importantly, becoming a masternode means a certain number of coins is locked —10,000,000 XDC in this case. Therefore, to control more than 3/4 masternodes, attackers must hold a considerable amount of XDC and gain substantial support from coin-holders. Given the inhibiting cost, the attackers do not have incentives to engage in any malicious activity that could harm the chain

However, in the worst-case scenario, XinFin Network can conduct a soft fork to reduce the number of masternodes, keeping the chain running and figuring out slasher mechanisms to weed out the malicious masternodes.

D. Relay Attack

The XinFin Network supports EIP155. The EIP-155 provides unique identifiers to a blockchain helping it overcome relay attacks. With EIP-155, two conditions are met: (a) definition of an integer for Chain-id for a blockchain and (b) signing of a ***chain-id*** into a transaction data. This prompts attackers to send the same transaction to different blockchains. With specifications in the EIP-155, blockchains have to define a ***chain-id*** and register the *chain-id* in a public repository.[9]

[9] Zoltu, Micah. "Ethereum/EIPs." GitHub, September 29, 2020. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-155.md.

A challenge for using an integer for chain-id is that it’s not broad enough to cover all blockchains and it doesn't prevent the use of the same ***chain-id*** by different blockchains. Furthermore, using an integer fails to address issues introduced by two forked blockchains having the same ***chain-id***. In this context, the XinFin Network has adopted a more robust blockchain identifier that overcomes these drawbacks, especially for cross chain operations where multiple chains are involved thus providing extensive protection against relay attacks. With the XinFin Network, the process through which a transaction id is made unique is as illustrated with the following example:

Consider a transaction with: **nonce = 9, gas price = 20 * 10**9, start gas = 21000, to = xdc3535353535353535353535353535353535353535, value = 10**18, data='' (empty).**

Once signed the “signing data" becomes:

0xec098504a817c800825208943535353535353535353535353535353535353535880de0b6b3a764000080018080

And, the "signing hash" becomes:

0xdaf5a779ae972f972197303d7b574746c7ef83eadac0f2791ad23db92e4c8e53

In the event that a transaction within the XinFin Network is signed with a private key like **0x4646464646464646464646464646464646464646464646464646464646464646**, then the v, r, s values would be:

(37, 18515461264373351373200002665853028612451056578545711640558177340181847433846, 46948507304638947509940763649030358759909902576025900602547168820602576006531)

With the use of 37 instead 27 in the v, r, s values, the signed Tx would become:

0xf86c098504a817c800825208943535353535353535353535353535353535353535880de0b6b3a76400008025a028ef61340bd939bc2195fe537567866003e1a15d3c71ff63e1590620aa636276a067cbe9d8997f761aecb703304b3800ccf555c9f3dc64214b297fb1966a3b6d83

Within the XinFin Network, a cross chain-id can be used to present a relay attack. Notably, applications handling cross chain transactions can verify cross ***chain-id*** via their block hash and decide whether the transaction is valid or not. Transactions without a verifiable cross *chain-id* are rejected. In effect, EIP-155 specifications provide a robust approach to preventing relay attacks.

Table 1 shows chains and ***chain-ids*** recognized on the network.

TABLE 1
CHAINS AND CHAIN_ID

CHAIN ID	Chain(s)
1	Ethereum mainnet
2	Morden (disused), Expanse mainnet
3	Ropsten
4	Rinkeby
30	Rootstock mainnet
31	Rockstock testnet
42	Kovan
61	Ethereum Classic mainnet
62	Ethereum Classic testnet
1337	Geth private chains (default)
77	Sokol, the public POA Network testnet
99	Core, the public POA Network main network
50	XinFin Mainnet
51	XinFin Testnet

E. **Safety and Liveness**

A consensus protocol is considered live if it can eventually propagate and make valid transactions onto the blockchain. A liveness fault occurs when transaction omission, information withholding, or message reordering, among a number of violations are observed. This type of fault is unlikely to happen in XinFin Network because the block creation masternodes list is ordered in a predetermined way for each epoch, thus if even an attacking masternode omits some transactions, the latter will be processed and validated by the next honest masternode in the next block.

Furthermore, safety implies having a single agreed upon chain where there are not two or more competing chains with valid transactions in either chain. As such, consensus protocols are safe when blocks have settlement finality, or else probabilistic finality. The XinFin Network provides safety because it has a settlement finality.

To note, XinFin Network has implemented the Istanbul Byzantine Fault Tolerant (IBFT) consensus mechanism. The IBFT consensus mechanism ensures instant finality, higher throughput, manageable validator set, and a high Transaction Per Second (TPS) rate.[10]

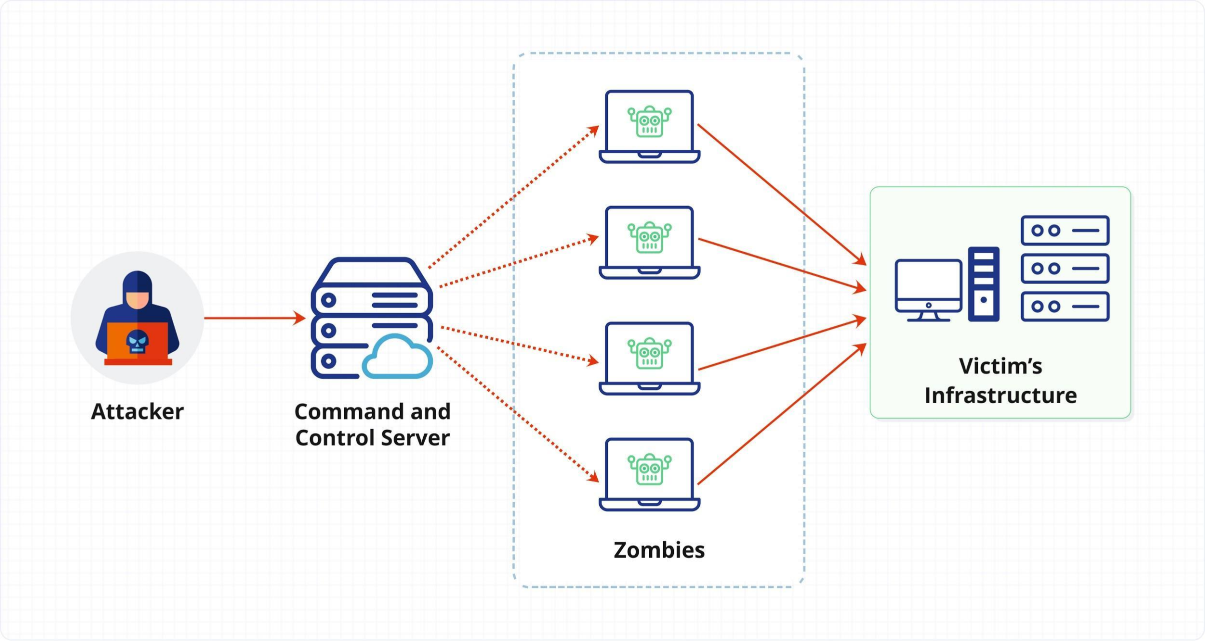
[10] Yutelin. “Istanbul Byzantine Fault Tolerance · Issue #650 · Ethereum/EIPs.” GitHub, June 22, 2017. https://github.com/ethereum/EIPs/issues/650

With the IBFT consensus mechanism, the XinFin Network introduces several benefits guaranteeing the network's safety and liveness. First, the XinFin Network—via the IBFT—guarantees immediate block finality. That is because only 1 block is proposed at a specific chain height. Thus, the single chain removes forking, prevents uncle blocks, and the risks that a transaction may be “undone” once on the chain at a later date. It’s worth noting that XinFin’s MyContract—a next generation smart-contract platform—will be IBFT compliant, enabling the consensus to scale up to 2500 TPS.

Second, the IBFT consensus mechanism reduces times between blocks. This occurs by effectively reducing efforts needed to construct and validate blocks, increasing the throughput of the network. Third, with the IBFT consensus, the XinFin Network ensures high data integrity and fault tolerance. To clarify, the IBFT employs a group of validators to ensure the integrity of each block being proposed. Plus, a super majority (66%) of the validators are required to sign a block Byzantine, which is inserted to the chain, making block forgery very difficult. Thirdly, the IBFT consensus mechanism guarantees operational flexibility. Notably, the ‘leadership’ of the network's validators rotates over time, preventing faulty nodes from exerting long-term influence over the chain, introducing undesirable liveness and safety issues.[11]

F. DDOS Attack

Distributed denial of service (DDoS) attacks occur when malicious characters overwhelm the target or the related infrastructure with malicious traffic. Employing networks of malware compromised computers, bots, and other devices, an attacker remotely controls the target infrastructure. DDOS adversary affects the bandwidth and connectivity leading to the disruption of services on a network. To add, cloud-based ecosystems suffer significant losses since DDOS causes service degradation and in some cases complete service denial.



[11] Yutelin. “Istanbul Byzantine Fault Tolerance · Issue #650 · Ethereum/EIPs.”

Fig.6.Distributed denial of service (DDoS) attack.[12]

In the context of the XinFin Network, masternodes are required to run on reputable public cloud providers like AWS, Microsoft Azure or Google Cloud, which provide multiple DDOS prevention mechanisms. When some nodes are attacked or fail-stop, the network still operates correctly as long as the number of failing and/or attacked nodes remains less than 1/4 of the number of masternodes.

G. Spam Attack

XinFin Network keeps the same transaction fee mechanism as Ethereum which employs gas prices.[13](Gas refers to the unit that measures the amount of computational effort required to execute specific operations on the Ethereum network).[14] However, the XinFin Network supports a minimum transaction fee (1 wei--approx. 1/100 the gas price of Ethereum). Concerns have been raised on the likelihood of spamming given that attackers try to broadcast a huge amount of low fee transactions to the system. To deter spamming attacks, however, the XinFin Network's masternodes always sort transactions and pick up only high fee transactions into the proposing block. Thus, spammers have little chance of harming the system.

IV. RELATED WORK

Consensus plays an important role in guaranteeing the success of distributed and decentralized systems. Bitcoin's core consensus protocol, often referred to as the Nakamoto consensus, realizes a "replicated state machine" abstraction. In the abstraction, nodes in a permissionless network reach agreement about a set of transactions committed as well as their ordering.[15] However, known permissionless consensus protocols such as Bitcoin's Nakamoto consensus come at a cost. Bitcoin and Ethereum rely on PoW to enforce, albeit roughly, the idea of "one vote per hash power" and to defend against Sybil attacks.

Unfortunately, PoW-based Bitcoin and Ethereum are known to have terrible performance (Bitcoin's transaction processing performance is at peak of around 7 transactions per second as previously mentioned). Moreover, PoW is much criticized because it costs a lot in terms of electric energy consumed.

XinFin's XDC Network Consensus is able to achieve 99% less energy consumption compared to Proof of Work based networks like Bitcoin or Ethereum. Data generated from XinFin's 140 Validators nodes power usage is compared to that of Bitcoin and Ethereum in Figure 7 below.

[12] Wani, Sharyar, Mohammed Imthiyas, Hamad Almohamedh, KhalidM Al Hamed, Sultan Almotairi, and Yonis Gulzar. "Distributed Denial of Service (DDoS) Mitigation Using ..."

[13] Buterin, Vitalik. "Ethereum Whitepaper." ethereum.org, 2013. https://ethereum.org/en/whitepaper/.

[14] Richards, Sam. "Gas and Fees." ethereum.org, January 15, 2021. https://ethereum.org/en/developers/docs/gas/.

[15] Nakamoto, Satoshi. "Bitcoin: A Peer-to-Peer Electronic Cash System." bitcoin.org, 2008. https://bitcoin.org/bitcoin.pdf.

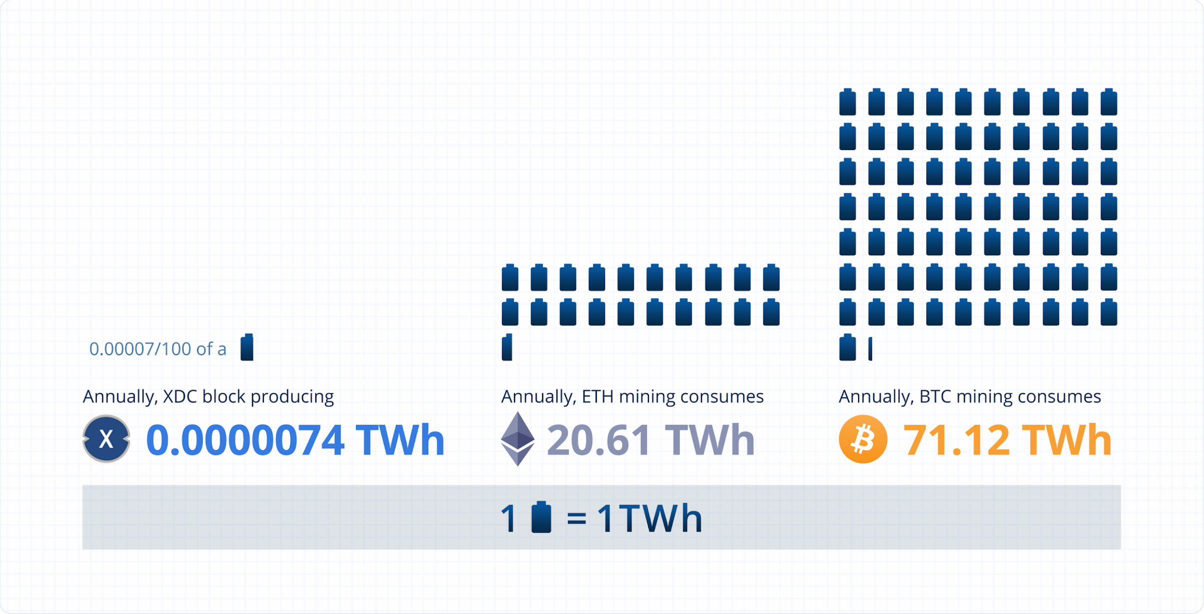


Fig.7. Comparison of Energy Consumption between Bitcoin, Ethereum and XinFin XDC Network.

To design an efficient and cost-effective consensus protocol in the permissionless model, PoS has been discussed extensively in the Bitcoin and Ethereum forum. A PoS blockchain can substitute the costly PoW in Nakamoto's blockchain while still providing similar guarantees in terms of transaction processing in the presence of a dishonest minority of users (this "minority" is to be understood here in the context of stake rather than computational power.) The Ethereum design, Casper, published by Buterin & Griffith, provides as its initial version a PoW/PoS hybrid consensus protocol, which might eventually switch to a pure XDPoS system. As with the XinFin Network, Ethereum Casper requires that validators (similar to block creators) deposit a specified amount of ETH — a concept similar to that used in the XinFin Network.

XinFin's Delegated Proof of Stake (XDPoS) consensus protocol, proposed in this paper, can be seen as a hybrid model. In particular, XinFin applied XDPoS with Double Validation to create and verify blocks smoothly and efficiently. Whenever potential fork branches are detected, XinFin employs the concept PoW to select the longest branch and then discard the other branches. With this hybrid approach, XDPoS does not only increase the performance and security of the blockchain, but it also reduces forks in an efficient and practical manner.

Recently, there are several consensus protocol research works that are closely related to XinFin Network such as EOS and Ouroboros of Cardano. The mechanism of relying on masternodes to reach consensus is utilized by Bitshares and EOS, whose consensus protocol is termed Delegated Proof-of-Stake (DPoS). This DPoS is similar to the XinFin Delegated Proof of Stake consensus in the XinFin Network. Both systems are similar in the sense that masternodes (block creators or witnesses in DPoS) are elected through a voting system.[16] However, XinFin Network requires that masternodes need to deposit a required minimum amount of XDC to become a masternode candidate, which puts more pressure on masternodes to work honestly. Furthermore, the Double Validation mechanism of XinFin Network lowers the probability of handshaking attacks and having invalid blocks, as

[16] Lee, Greg. "EOSIO/Documentation." GitHub, April 28, 2018. https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md.

previously analyzed. EOS also has a maximum of 21 block producers for each epoch, which is less decentralized than XinFin Network with a maximum of 108 masternodes.[17,18]

The research-backed Cardano blockchain solution, namely Ouroboros, with the ADA coin, which is purely based on Proof-of-Stake, promises to provide rigorous security guarantees. Similar to the XinFin Network, Ouroboros has a set of block producers for each epoch creating blocks in a given sequence. Additionally, each block producer candidate needs to deposit a minimum amount of stake (an amount of ADA).[19] Worth noting, however, is that Ouroboros only provides Single Validation, while the XinFin Network carries out Double Validation providing several advantages over Single Validation, as previously analyzed. In Ouroboros, the order of block producers, selected among stakers, is based on a biased randomization, while the XinFin Network randomization for block verifiers is potentially uniform and based on smart contracts. Furthermore, the use of voting as in XinFin Network and DPoS enables more incentive equality between stakers.

V. CONCLUSION AND PERSPECTIVES

In this paper, the XinFin Network proposed a XDPoS blockchain protocol which has heuristic and rigorous security guarantees, in addition to fast finality. Also presented is a novel reward mechanism, showing a capability to lower the probability of having forks, and guaranteeing fast confirmation time. Plus, the reward mechanism indicates that the contributions and benefits of masternodes are fair in the sense that the probability distribution function is uniform, eventually.

Perspectives

- Future work: The XinFin team is currently working on the implementation of the XDPoS, which will be released on schedule as stated in its roadmap. Furthermore, in line with the network's novel consensus protocol, its team will investigate the Sharding mechanism in order to provide even better transaction processing and performance. The XinFin team believes that the Sharding technique with a stable number of masternodes will provide better stability and efficiency to the blockchain. At the same time, the XinFin team commits to keeping EVM-compatible smart contracts within the network's masternode Sharding framework.

- Economic sustainability is also an important concept for a blockchain based decentralized network. That means to maintain the network in a sustainable condition, an equilibrium needs to be achieved, in which the cost of running the network infrastructure could be offset by the revenues generated. In this context, the cost of network infrastructure consists of two parts: the physical cost of having hardware such as servers, memories that pass the network technical requirements; and the capital cost of having XDC locked into smart-contracts. The revenues for masternodes would primarily come from Reward Engine emission, and later on from service revenues such as token exchange fees provided by applications running on

[17] Lee, Greg. "EOSIO/Documentation.

[18] Lee , Greg. "EOSIO/Documentation.

[19] Hoskinson, Charles. "Why We Are Building Cardano --a Subjective Approach."

top of the XinFin Network. We will publish a XinFin Network economic analysis and proposal, separate from this technical paper, at a later date.

VI. Updated Note as on 17th march, 2021

Consensus level issue:

XDPoS Network is stable for the last 2 years but one small Side Chain has been created as of 14th march, 2021

Here is the event description and how fixed by the masternode holders:
Incident date as on: 14-03-2021
Total Node affected: 67 out of 143
Side Chain Created at Blocks No: 27307800

Byzantine Fault/Problem: Two separate networks started after block 27307800 where one original network and 27 nodes started synching as an additional side network within a few minutes 67 nodes became part of the side network. Issue also described as Byzantine fault.

Quick fix: Got Validators support (side chain validators) and resolve additional network issues by restarting/syncing nodes with original nodes network.

XinFin Network reference to understand issue:
XinFin Network is Fork of Ethereum Network and change Network Consensus to XDPoS
Reference to GitHub code for consensus:
https://github.com/XinFinOrg/XDPoSChain/tree/master/consensus

Current network consensus details:
Validators Node have to stake 10Millions XDC to become a part of validators Network.
Nos of masternoder/validators: 108
Backup Node: 32
Transaction process: ⅔ Node Validation to Write Block of the transaction
Block time: 2 Seconds.
Block confirmation time to sync: 32 Blocks.
Current size of data: 40GB+
More details on Consensus at: https://XinFin.org/XinFin-consensus

Network Bounty: Technical as well as Research team is still looking to resolve the above consensus level issues. Foundation announced a large bounty to prevent the above-mentioned network consensus issue. XinFin Network invites the technical/research team to welcome suggestions to resolve the above issue. Please create a GitHub pull request and claim XDC bounty if you would like to collaborate with a solution.

This whitepaper evolved on a time-to-time basis after Livenet and testing various situations and errors faced by the network. Please feel free to add/suggest your feedback on the GitHub page.

Endnotes:

1. CB Insights. “58 Big Industries Blockchain Could Disrupt.” CB Insights Research. CB Insights, March 19, 2021. https://www.cbinsights.com/research/industries-disrupted-blockchain/.
2. Ganne Emmanuelle, and Deepesh Patel. “Blockchain & DLT In Trade: Where Do We Stand?” Trade Finance Global. © Trade Finance Global, October 1, 2020. https://www.tradefinanceglobal.com/wp-content/uploads/2020/11/2020_TFG_BLOCKCHAIN-_-DLT-IN-TRADE_Compressed.pdf.
3. Digiconomist “Bitcoin Energy Consumption Index.” Digiconomist, March 10, 2021. https://digiconomist.net/bitcoin-energy-consumption/.
4. Digiconomist “Ethereum Energy Consumption Index (Beta).” Digiconomist, March 5, 2021. https://digiconomist.net/ethereum-energy-consumption.
5. XinFin. 2021. "How to Deploy a XinFin Public masternode". XinFin.Org. https://docs.XinFin.org/docs/raw/masternodes.
6. Hoskinson, Charles. “Why We Are Building Cardano --a Subjective Approach.” https://whitepaper.io/search/result?q=Cardano. Cardano. Accessed March 28, 2021. https://assets.whitepaper.io/documents/HkUIhFWhL/HkUIhFWhL.pdf?AWSAccessKeyId=AKIAI7X7HAN3A7HLG22A&Expires=1616935771&Signature=42S80pEKu3c1R5yA%2Bh6p1i5srrA%3D
7. Zoltu , Micah. “Ethereum/EIPs.” GitHub, September 29, 2020. https://github.com/ethereum/EIPs/blob/master/EIPS/eip-155.md.
8. Yutelin. “Istanbul Byzantine Fault Tolerance · Issue #650 · Ethereum/EIPs.” GitHub, June 22, 2017. https://github.com/ethereum/EIPs/issues/650
9. Wani, Sharyar, Mohammed Imthiyas, Hamad Almohamedh, KhalidM Alhamed, Sultan Almotairi, and Yonis Gulzar. “Distributed Denial of Service (DDoS) Mitigation Using ...” Multidisciplinary Digital Publishing Institute. Accessed March 29, 2021. https://www.mdpi.com/2073-8994/13/2/227/pdf.
10. Buterin, Vitalik. “Ethereum Whitepaper.” ethereum.org, 2013. https://ethereum.org/en/whitepaper/.
11. Richards, Sam. “Gas and Fees.” ethereum.org, January 15, 2021. https://ethereum.org/en/developers/docs/gas/.
12. Nakamoto, Satoshi. “Bitcoin: A Peer-to-Peer Electronic Cash System.” bitcoin.org, 2008. https://bitcoin.org/bitcoin.pdf.
13. Lee, Greg. “EOSIO/Documentation.” GitHub, April 28, 2018. https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md.

Figures List: