BOSAGORA White Paper

January 16, 2020

Abstract. The BOSAGORA platform is a decentralized self-evolving cryptocurrency that is built on Trust Contracts and an embedded decision-making system called the Congress Network. (1) Trust Contracts are securely executable contracts based on a protocol layer. We intend to provide an efficient, safely designed smart contract engine and provide an easy-to-develop language with many tools and popularity for easy adoption by developers. (2) The Congress Network is the decision making body in the BOSAGORA platform which solves governance issues arising in decentralized organizations. Through a clearly defined and automated governance system, we aim to continuously develop the community and software into a more anti-fragile ecosystem.

Background

The blockchain was first conceptualized in Satoshi Nakamoto's white paper "Bitcoin: A Peer-to-Peer Electronic Cash System" in 2008[1]. The technology was implemented the following year as the central technology behind Bitcoin. Bitcoin uses blockchain technology as a financial transaction ledger where individuals publicly record transfers of currency. Bitcoin was the first of its kind to use the blockchain to successfully solve the double-spending problem. Despite the absence of a centralized administrator, Bitcoin has successfully supported over 180 million peer-to-peer transactions and now has a market capitalization of more than 100 billion dollars.

Following the success of Bitcoin, there have been numerous systems leveraging blockchain technology. There are hundreds of competing cryptocurrencies and according to a recent IBM report, more than 90% of banks are investing in blockchain technology. Currency transactions are the most common applications of blockchain technology[2]. However, some groups are also attempting to transfer and manage other kinds of digital assets using this technology, such as financial products and services, logistics information, property ownership, identity etc.

The cryptocurrency Ethereum gained a lot of traction in 2016 and aims to provide smart contracts on the blockchain: "A blockchain with a built-in fully fledged Turing-complete programming language that can be used to create 'contracts' that can be used to encode arbitrary state transition functions."[3]

The goal is to allow users to write any kind of program (or contract) onto the blockchain. Similar to Bitcoin, Ethereum uses the blockchain and a consensus mechanism to ensure that if a malicious node attempts to forge the content of the contract, the forged contract will eventually be removed from the blockchain. As Bitcoin ensures the integrity of the amount of Bitcoin being transferred between accounts, Ethereum must similarly ensure the integrity of the contract being executed.

Smart contracts have the potential to be a paradigm shift in the development of decentralized applications. Programs that are not held on a centralized server, yet can run the same logic anywhere. Smart contracts can be used to develop: decentralized marketplaces, currency exchange platforms, and projects like Golem[4] which aim to create a decentralized worldwide super-computer.

However, the freedom and flexibility provided by the Turing-complete language which Ethereum is based on is the cause for several serious problems. We believe that using a turing-complete language may be inappropriate for writing smart contracts as they are

[1] Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/bitcoin.pdf
[2] Leading the Pack in Blockchain Banking: Trailblazers Set the Pace, https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBP03467USEN&
[3] Vitalik Buterin, Ethereum Whitepaper, https://github.com/ethereum/wiki/wiki/White-Paper
[4] Golem, https://golem.network

inherently undecidable.[5] Due to this undecidability issue, a smart contract based on a Turing-complete language will make it difficult to know what a smart contract will do before running it. Ethereum attempts to overcome this issue by applying a cost to computational work (gas), however the inherent issue of the language used to program and execute smart contracts has inevitably led to a series of security vulnerabilities[6] and outright failed projects such as The DAO.[7]

Vision

Contribute to making a better world with blockchain technology as a project enabler.

Mission Statement

Building an open decentralized blockchain protocol that ensures the transparency of consensus algorithm and the clarity of contract, thereby enriching the blockchain ecosystem through enabling the meaningful projects with the expression of the collective intelligence by an advanced democratic decision-making process.

Core Values and Key Attributes

Forward Thinking

Pioneering future realization: We aim to develop a first full-node Proof of Stake and Federated Byzantine Agreement consensus algorithm blockchain platform with innovative technology development that anyone can experience speed and trust.

Fair

Mature democracy: Everyone can embody democracy that guarantees the highest level of fairness through free and inclusive decision-making with the advanced deliberative democratic decision-making tool.

Dependable

Clear transparency: To make it easier for anyone to see the entire project through transparency and to make decisions based on established procedures. (Community update, Technical advisory board, Github, Congress voting process)

[5] Hodges, Andrew, Alan Turing: the enigma, London: Burnett Books, p. 111
[6] N. Atzei, M. Bartoletti, T. Cimoli, A survey of attacks on Ethereum smart contracts, https://eprint.iacr.org/2016/1007.pdf
[7] The DAO, https://slock.it/dao.html

ICO and the Original White Paper

BOSAGORA received a surprising response from 95 countries in May 2017 to achieve the 6902 BTC hard cap in just 17 hours. The result was achieved by the diverse technological and ecological blueprints pursued by the existing white paper. However, many similar projects have been announced over the past 2 years, and it has become difficult to gain exclusive status with technology development plans and ecosystem blueprints alone. Besides, competition in the blockchain platform market is getting even more intensive as the global giant is also signaling the launch of the blockchain platform. Under these circumstances, BOSAGORA should try to both pioneer new areas, where it could gain a more exclusive status to survive, and retain the framework and spirit of existing white papers to keep the promise with the participants of the Initial Coin Offering.

Since the ICO, regulations have changed along with numerous technological advancements. BOSAGORA team focuses on delivery adhering to the original white paper but at the same time, we must make amendments to reflect the changes in policies, technology and methodologies.
As a result, our team will create a more robust and up-to-date platform while keeping the promise of the value and vision found in the original white paper.

The promise of the value and vision found in the original white paper should be maintained. In other words, fundamentals such as the formation of the Congress Network which all nodes participate in the decision-making, the provision of the Commons Budgets that can be utilized if the congress wants to, and the functions as a mainnet platform that supports various dapps and business partners should remain as it was written.

A distinct aspect of BOSAGORA's operating principles is that it can unleash collective intelligence because all nodes are involved in the decision-making process. In particular, thanks to the advanced form of mature decision-making capabilities of the BOSAGORA, various opinions will be aggregated into harmonized forms. Through this harmonious process of collective intelligence, it is ultimately what BOSAGORA seeks to improve its ecosystem.

Proposal

Anti-centralizing Consensus Algorithm. Cryptocurrencies like Bitcoin, that only use a proof-of-work(PoW) type consensus protocol, are affected by issues arising from the non-separation of economic and political incentives. By buying up more mining hardware, a user can attain more control of the blockchain(political) and also increase their mining income(economic). BOSAGORA overcomes this issue by using a consensus mechanism(explained in more detail below) that separates economic incentives from political ones. Attaining either political power or economical wealth requires an investment into the system. A user can either acquire more votes by increasing the number of nodes(one operational node equals one congressional vote) or a user can invest in confirmation rewards(rewards relative to the amount of coins locked away in a node) to maximize mining income.

Governance. Decentralized systems lack a systematic decision making process. There have been several cases in the cryptocurrency space, where this led to confusion and substantial financial losses. BOSAGORA constitutes a governance system whereby node operators referred to as the Congress Network can participate in creating and voting on proposals in order to continuously improve the software and ecosystem. System changing proposals that are voted on the Congress Network and are accepted, are considered to have reached a social consensus, and the changes in the proposal are automatically applied to the network.
Another type of proposal is a funding proposal. These proposals are requests for funds from the Commons Budget and they are also voted upon by the Congress Network. BOSAGORA sets aside a large public budget specifically for the development of the BOSAGORA ecosystem through these proposals. We will explain further later in this paper.

Trust Contracts. BOSAGORA team aims to implement Trust Contracts, which enable a safe, accurate, programmable and executable contract as in the original intention.

Rather than continuing what is not feasible, we will redefine 'Trust Contracts' and will actively pursue the selecting the optimal direction and applying the suitable technology to improve the core protocol. This approach also considers adopting a methodology that uses flexible programming language on top of virtual machines and we are currently exploring WebAssembly as other industry players do. In the end, we intend to provide an efficient, safely designed smart contract engine and provide an easy-to-develop language with many tools and popularity for easy adoption by developers.

WebAssembly is a new type of code that can be run in a modern web browser. It provides new capabilities and offers significant performance benefits. “WebAssembly is a binary instruction format for a stack-based virtual machine. WebAssembly is designed as a portable target for compilation of high-level languages like C/C++/Rust, enabling deployment on the web for client and server applications.”

Running programs that are written in multiple languages on the web at near-native speeds using client applications was previously impossible. Running the codes on WebAssembly is similar to the actual hardware. With WebAssembly, developers can code in a variety of programming languages such as C++ and Rust, and they can expect to run the program in near-native performance. EOS also uses WebAssembly, and many blockchain platforms such as Ethereum, Tron and Cardano have already introduced or plan to introduce the virtual machines using WebAssembly.

Once the feasibility of the implementation plan has been studied and the solution has been discovered, technical and practical measures will be taken to complete the objectives and directions for the "Trust Contracts" presented in the original white paper.

Features	Bitcoin	Ethereum	BOSAGORA
Coin	BTC	ETH	BOA
Core Features	Financial Transactions (Bitcoin script)	Smart Contracts (Solidity, Serpent, etc)	Trust Contracts (WASM)
Decision Making Process	Non-systematic	Non-systematic	Democratic Congress (One node = One vote)
Consensus Algorithm	Proof of Work	Current: Proof of Work Future: Casper(?)	Modified FBA(Federated Byzantine Agreement)
Block Size	1 MB	Dynamic	Dynamic

Fig 1. Comparison of Cryptocurrencies

Consensus Algorithm

Overview

The consensus algorithm is core to any blockchain based currency or system. The algorithm attempts to answer the question, ‘How can we prove with confidence that all distributed databases hold the same set of information?’
In response to this question, BOSAGORA uses a Modified Federated Byzantine Agreement(mFBA) consensus algorithm based on Stellar’s Consensus Protocol(FBA).[8]

Consensus Algorithm	Proof of Work	Tendermint	Byzantine Agreement	FBA[1]	mFBA[2]
Decentralized Control	O	O		O	O
Low Latency		O	O	O	O
Flexible Trust			O	O	O
Asymptotic Security		O	O	O	O
Governance Features					O
Staking Features		O			O
[1] Federated Byzantine Agreement [2] Modified Federated Byzantine Agreement (BOSAGORA protocol)					

Fig 2. Comparison of Consensus Algorithms

Mazieres defines key features of the federated Byzantine Agreement Protocol:

- Decentralized control. Anyone is able to participate and no central authority dictates whose approval is required for consensus.
- Low latency. In practice, nodes can reach consensus at timescales humans expect for web or payment transactions—i.e., a few seconds at most.
- Flexible trust. Users have the freedom to trust any combination of parties they see fit. For example, a small non-profit may play a key role in keeping much larger institutions honest.

[8] David Mazieres, Stellar Consensus Protocol, https://www.stellar.org/papers/stellar-consensus-protocol.pdf

- Asymptotic security. Safety rests on digital signatures and hash families whose parameters can realistically be tuned to protect against adversaries with unimaginably vast computing power.
- Governance Features. Voting and features that are related to operating the congress are additional features embedded into the protocol.

Federated Byzantine Agreement Consensus Algorithm

Bitcoin's consensus mechanism and the traditional Byzantine agreement based protocols require a unanimous agreement by all participants of the network. However, the federated Byzantine agreement(FBA) does not require an unanimous agreement by all participants and additionally each node can choose which nodes to trust. This results in faster transactions without losing integrity of the financial network and allowing for organic growth of the network.

FBA implemented this type of non-unanimous consensus mechanism by grouping nodes into teams (also known as Quorums). When a transaction is made, the information is sent to all those in the group. Rather than waiting for the whole network to agree on the state of the data, if a node hears the same message from a sufficient number of trusted nodes, the node assumes the information is correct. The overlapping of nodes, or loose federation of nodes, results in different nodes that have different sets of teams to agree on the same transactions. This leads to a system-wide consensus, without requiring unanimous agreement for each transaction block.

In situations where nodes are in disagreement over a fraudulent transaction, there is a ballot system embedded into the system to overcome such issues. Further technical details regarding FBA can be found in Stellar's consensus protocol paper.

How is the modified federated Byzantine agreement(mFBA) algorithm different?

In addition to FBA, the BOSAGORA consensus protocol also applies a Proof of Stake feature for the maintenance of the governance system. Validators need to freeze 40,000 BOA within a node and forgo liquidity. The frozen coins in the node then act as both an economic incentive(Confirmation Rewards) to operate a node as well as collateral for the security and integrity of the information held in the node's blockchain. According to the pre-set rules, if the node is discovered to have forged the blockchain on the node, the frozen coins are forfeited to the Commons Budget.

Congress Network

Overview

The Congress Network is the decision-making body for BOSAGORA consisting of fully-synchronized node operators. It enables effective and inclusive collaboration among the various project stakeholders to continuously enhance the software and the ecosystem. For example, the development of the source-code and marketing resources can be proposed, decided, and implemented within the system.

All node operators of BOSAGORA can join the Congress Network and participate in the collective decision-making process. The BOSAGORA Congress Network enables its members to engage and contribute through proposals, discussion, voting, and reviewing issues of the project's common interest. The Congress Network adheres to the 1-node-to-1-vote rule.

The Need

Like any other product, blockchain projects must satisfy the needs of potential users. Yet, no matter how intricate the initial design be, technology, people, and the market will always change directions, and projects should be accordingly and continually adaptive to the changes. Choosing when and how to change the network is critical to sustainability and growth. Communicating every stakeholders' interests and perspectives into an agreement can be a painfully long process, resulting in centralized governance systems even for blockchain projects, which, by nature, are about decentralization.

Even with the best intentions, a centralized decision-making process will inevitably leave out the comprehensive voices of the network. If members do not have a channel to participate and make changes about their problems, they have no other choice but to leave and move to another alternative, diminishing network effects. Inclusive and collaborative governance is essential to a successful project.

Problems of collaborative decision-making

Poor decisions are caused by many reasons, including challenges that arise from the social and participatory nature of decisions. Incomplete information, power dynamics, biases, and peer pressure make teams and communities reach poor decisions that are not inclusive of the best solution.

- Incomplete information: information about the topic that requires a decision may be incomplete. This information may be concrete facts about the topic or personal experiences of groups who are directly affected by this decision.
- Power dynamics: decisions are made by a small group of people without taking into account the opinions of others who are often most vulnerable to the consequences.

Discussions are also dominated by a certain group of individuals who happen to crowd out other marginalized voices.
- Cognitive biases: subconscious (or conscious) biases prevent ideas from being evaluated on their merit
- Peer pressure: social or peer pressure prevents constructive feedback and dialogue

These problems with today's decision-making are exacerbated by the notable absence of moderation resulting in ineffective collaboration. Lack of facilitation and rules, people easily fall into the trap of ever diverging opinions, and the discussion fails to progress.

Introduction of Congress Network

We propose a decentralized collaborative decision-making system based on node deliberation and a voting mechanism, namely BOSAGORA Congress Network.

Functions

Congress Network will serve as a platform where:
- Members can actively exchange ideas and communicate together
- Decisions can be reached on proposals to implement on BOSAGORA network

In BOSAGORA, there are mainly two types of proposals.
- System Proposals are the ones that impact the BOSAGORA platform. It includes decisions to alter and improve the technical functions and features of the network. The development team's direction will be reflecting the decisions reached by the Congress Network.
- Funding Proposals are the proposals requesting funds from the Commons Budget. The usage of this budget shall be related to the expansion of the BOSAGORA's ecosystem, such as blockchain Dapp projects, non-blockchain for-profit business investments, and non-profit donations for the ecosystem.

Features

To overcome the problems of collaborative decision-making, the Congress Network is equipped with the following features.

Voice (Equitable) : Discussions are dominated by individuals with the loudest voice and self-assurance in abundance. These individuals are often a part of social groups in positions of power and can crowd out ideas from other members, leading to disappointing and non-representative outcomes (made by a few, affects many).
To encourage the principle of equal opportunities of speech, discussions on Congress Network are based on 'voice'.
'Voice' is the ability to contribute: participating in discussions, suggesting proposals, and evaluating outcomes.

It is distributed equally to all members within discussions. It is also limited to prevent inefficient and lengthy communication and trolling. You can set the number of voices for an activity when creating a new activity.
If you have exhausted your voice, you can ask other members to donate their voices to you. If you believe a member has something valuable to add to the topic at hand, you can delegate some of your voice to them. Voice enables decentralized curation, where everyone can participate in shaping valued voice without forcefully reducing other members' opportunities.

Incognito : It is the concept of privacy and eligibility guarantee of the Congress members and ecosystem members. To create a platform that fosters open participation by members while minimizing fears of disclosure, AAID technology (Authenticated Anonymous Identity) is used to generate anonymized accounts for members once they pass through eligibility verification and to derive disposable identities, or unique one-time permit for each activity to prevent users' digital footprint across different activities and groups from being traceable. This technology enables your privacy being preserved even in the unlikely scenario of our servers being compromised. It would maintain trust while ensuring the elimination of biases and promote freedom of speech during discussions.

Eligibility can be verified by the network and AAID provider, who issues the proof of verification. AAID is derived from, yet un-linkable to Proof of Eligibility (address of the node with more than 40,000 BOA) using homomorphic encryption.

This concept is implemented in discussions and reviews. For voting that determines tangible impacts of system changes or funding usage, verifiability of the records takes priority, and AAID will not be applied.

Flexible (Adaptable) : Recognizing that everyone has different ways of approaching a problem, choice, and decision, the Congress Network is designed with various stages and features that are optional according to your needs. To discover a decision flow that is suitable to your needs, the process flow can be directly customized to structure discussions and problem-solving methods. Additional activities of discussion, review, or voting can be added to existing activities to create a flexible flow that best reflects changing priorities in real-time.

Facilitated (Efficient) : To help you focus on discussions on a topic to reach an outcome, the Congress Network has activity templates and an extensive catalog of features to signpost expectations for members at each stage, whether you're looking for a lively discussion, vote on a set of proposals or evaluation on the outcome.

Archived & Interlinked (Transparent & Accountable) : Transparency and accountability are essential foundations for a good decision. The Congress Network will archive as much data as possible, but due to limitations on block sizes, the blockchain will hold hashes of records that can verify the decisions. More accessible archives of discussion, votes, reviews will be provided on a third party server to help members keep track of internal discussions and outcomes, measure performance across time, and be inspired. Archiving is different

from the conventional practice of storing your data. AAID generates a unique one-time permit for each activity to prevent a traceable digital footprint, and only the members can access their own entire activity log on the platform.

Process

Joining the Congress Network

Anyone can become a Congress member as long as you fulfill the following conditions. You will be regarded as a Congress member if you meet the following criteria:

- Run a fully-synchronized node at stable network speeds as an actual validator
- Freeze at least 40,000 BOA

A node could be a server or a personal computer that a Congress member runs. The node can be located at home or a remote location, as long as network speeds are stable. Congress Members have the choice to either invest in increasing their political influence through running more nodes or increasing their economic return through increasing the number of staked BOA token.

Creating Activity

When members feel the need to discuss and reach decisions, they can create an Activity to do so, choosing from different templates depending on the nature of the topic. Any topics from new business opportunities, systemic modification, or even the general sentiments on the project that requires inputs from other group members can be brought up. There are currently 3 templated Activities members can choose from.

- Discussion: Members can exchange opinions, discuss or brainstorm ideas.
- Voting: Members can vote on issues and comment on vote-able options.
- Review: Members can review products and decisions or participate in surveys.

Additional templates with more features can be introduced reflecting on users needs.

All members can make proposals. To ensure the quality and the accountability of the proposals, a fee structure will be introduced in the future.

Guide

For each activity created, the organizer needs to input necessary information for other members to understand and engage.

1. Links (optional): An activity can be created as an add-on to previous activity. If there was a vote to change a community rule, after a period, a review activity related to the vote could be created connected to the previous vote.
2. Name: A label, 20 characters or less
3. Description and Goal Statement: What is this topic about? Describe the purpose and context.
4. Deadline: Until when should this activity be open? Until when should a decision be reached?
5. Advanced Settings (optional): Templates have different features to set, including but not limited to the number of voices to be distributed, type of ballots, and how the result should be shared.

6. Fee (optional): Activity creation is free, but for funding requests, a fee needs to be paid first before creating a vote. (Fee structure will be introduced later)

Discussion

Members can give their opinions and comments. Using voice, members can write opinions. Opinions can be recommended, and will be listed either in highest recommendations or by most recent additions. Opinions can be edited, but history of edits can be viewed by other members but cannot be deleted. Members can comment on opinions but comments cannot be edited or deleted.

If there are opinions or comments that are problematic and go against the community rules, members can mute the opinions and comments to make them no longer viewable on the local app.

Voting

Votes are created to reach a collaborative agreement. Since the result of the vote is to have direct impacts and changes to the BOSAGORA platform, the results must be recorded to be verifiable later.

To make vote verifiable and also keep the mid voting results hidden until the deadline, we have implemented the validator hash chain scheme.

When a person becomes a validator and therefore a Congress Member, the member must pick a random number known only to itself. The member will hash the random number for n times. The random number that has been hashed for n times is represented as Hn-1. The n becomes an index for locating a value.

If a validator decides to vote when the blockheight is 500, she will give her the commitment of H1008. If the vote will close when the blockheight is 600, which is 100 block time away, the validator would automatically broadcast the preimage by 100, which would be H908. With the preimage, it is easy to verify the validity of H1008 but it is not possible to figure out H908 from H1008.
By using H1008 as a verification value in proof of stake layer, the validator will be penalized for not broadcasting it to the network. The voting system will benefit by adding an additional layer of incentive for the validators to properly participate in the vote.

The validator will have to present its commitment and the ballot, and the ballot will be encrypted with a private key derived from H908. ex) H(H908||”Congress Network”|| vote ID) When the vote is closed, the validator will present H908, and the vote server will be able to decrypt the ballot and tally it.

Tallying the Votes

At the end of the voting period, the system stores the vote results. The date and time of each vote is retained and if the same ballot is duplicated, only the last vote counts as the final

result, ensuring singularity of 1-node-1-vote. This process and the results are verifiable by the scheme explained above.

Quorum

A quorum is the minimum number of the congress members that must participate in the vote for the proposal to be recognized as legitimate. The quorum for a congress voting process will be set initially to one-third of the whole congress members, but will dynamically adapts to reflect the average participation.

Approval occurs when the number of agreement votes minus those of disagreements make up to 10% or more of the total available votes. Disapproval occurs when disagreement votes minus agreement votes count up to 10% or more of the total available votes.

Implementation

After the proposal is passed, the proposal has to be implemented. The development team will take charge of passed system proposals and give when necessary development plan, roadmap, and security tests.
Funding proposal budgets will be distributed in accordance with the written contract. As the Trust Contract is implemented, on-chain transactions could be created to manage this.

Review/Inspection

After the implementation of the proposal, it will be subject to review by Congress and the Foundation. There should be scheduled inspection according to the roadmap of the proposal. Reviewing activity templates are available to aggregate members' view on ongoing proposals.
For funding proposals, the review costs will be covered by fee paid by the proposer.

Conclusion

Congress Network is not an exclusive model for the BOSAGORA community. The system envisions the application of such inclusive, effective, and transparent collaboration to any organization and communities that requires multiple stakeholders to shape their lives, from public policies, crypto communities, to even enterprises. We believe that we can create an ecosystem to iteratively learn the best ways to structure for direct deliberative democracy.

Network Interactions

Transactions

When a transaction of digital assets is requested by a user, the request is sent to the Congress Network. For a simple transfer of BOSAGORA, when a node confirms the block the user’s transactions will be confirmed, and the BOSAGORA will be transferred to another wallet. For more complex Trust Contracts, the pre-defined logic/procedures will also be carried out. In the initial stage of BOSAGORA, transaction fees will be fixed at 0.01 BOA.

The fixed transaction rate can later be adjusted by the Congress Network through the voting process. Transaction fees act as an economic incentive for node operators and also as a defense mechanism against DoS attacks.

Proposals

Proposals are system changing plans or Commons Budget spending plans that are submitted to the Congress Network. When a proposal is made, the ‘net percentage point difference’ between the positive and negative votes must exceed 10% for the proposal to be passed. For a funding proposal if the proposal passes, the requested coins will be sent to the proposer. Under some conditions, such as when the size of the proposal is large, the system can define a contract that requires a report on how the coins were spent.

Coin Freezing

Coin Freezing is a Proof of Stake concept where if a participant wants to become a validator, he/she locks-in his/her coins. Frozen coins are used as collateral in case of attempted forgery of the blockchain. If a node attempts to forge the blockchain, a portion of the frozen coins are confiscated. Additionally the system requires two weeks prior notice to unfreeze coins, as a mechanism to promote price stability. This concept will be explained in the technology part of this paper.

Reward System

Bitcoin suffers from the hash power centralization issue, due to its reliance on a Proof of Work consensus protocol. A small number of major miners can easily buy up large amounts of mining hardware, which allows them to influence changes in code and even threaten the integrity of the blockchain. By separating the incentives of those that wish to optimize financial gain, the barriers to entry to participate in the governance process is comparatively lower than a system that equates decision making power with financial rewards.
There are two ways for Congress Members to receive BOA rewards: confirmation rewards, and transaction fees.

- **Confirmation Reward:** Confirmation rewards are given to a node when a block is confirmed. This reward is crucial in providing a financial incentive to operate a node and the reward is directly linked to the number of frozen coins in a node. The reward is issued relative to the proportion of frozen coins held in the node. Initially the block confirmation reward starts at 27 BOA per 5 seconds, and it will decrease by 6.31% year on year over roughly 128 years. The rewards will be distributed to validators when a new block is created.

- **Transaction Fee:** The transaction fee is a fixed 0.01 BOA. Congress Nodes receive 70% of the collected transactions fee in a block, and 30% is sent to the Commons Budget. Transaction fees can be adjusted through the Congress.

Commons Budget

The Commons Budget is an account where BOA are held and can only be transferred by proposals that are passed through the Congress. The main role of Commons Budget is to expedite the growth of the coin users during the early stages. Coins in the Commons Budget are mainly accumulated through two channels; the first is the direct issuance of 50 BOA coins per 5 seconds for roughly 6 years and secondly from 30% of the transaction fee. This will ensure funds are available to growth hack the adoption of BOSAGORA.

Any proposal which passes through the congress can access coins from the Commons Budget. An example of a proposal is an Airdrop proposal; geo-socially distribute free coins to users in order to increase the number of BOSAGORA users. Other examples can include funding the development of the BOSAGORA ecosystem, marketing campaigns and organizing BOSAGORA related meetings.

Token Distribution and Issuance

BOSAGORA Token Distribution

BOSAGORA has conducted an airdrop of BOA to BOS holders from Thursday, May 16th to September 30th, 2019 according to the snapshot taken on Friday, April 5th, 2019, 12:00:00 UTC. According to the snapshot, 542,130,130.1958463 BOS coins were in supply.

- 500,000,000 BOS is initial supply
- 41,420,159.8931463 BOS is BlockchainOS PF00 membership rewards issuance
- 709,970.3027000 BOS is BlockchainOS PF01 membership rewards issuance

After the finalization of BOA token airdrop, the distribution plan for BOA token will be the following:

Category			BOA	Share
Initial Supply	Airdrop for BOS holders		247,595,031	5.09%
	Unclaimed	Burn	92,130,130	
		Marketing	30,000,000	0.61%
		Remain	82,404,969	1.66%
	Original Distribution	Foundation	40,000,000	0.81%
		Members	40,000,000	0.81%
		Bounty	10,000,000	0.20%
	Initial supply total		542,130,130	
	Token burn	BCOS PF	42,130,130	
		Unclaimed	50,000,000	
	Initial supply total after burn		450,000,000	
Confirmation Rewards			2,700,000,000	54.54%
Commons Budget			1,800,000,000	36.36%
Total			4,950,000,000	100.00%

Fig 3 : BOA Coin Issuance Plan

The number of airdrop tokens for BOS holders is 247,595,031.305721. The number of unclaimed tokens after the finalization of airdrop is 204,535,098.694279.
From the total of 204,535,098.694279 of unclaimed tokens:

- 42,130,130.1958463 tokens are issued by Public Financing, which was never the intention of the BOS platform foundation, thus, it should be burned.
- 50,000,000 also will be burned. The foundation has decided to burn 50,000,000 BOA from the unclaimed tokens, which is 10% from the original issuance plan.
- 30,000,000 BOA will be reserved for marketing purposes and will be used for exchange listings and partnerships.
- 82,404,968.6942793 will remain unclaimed.

Therefore, the actual initial supply will be 450,000,000 BOA. The foundation will make a separate announcement regarding the token metrics when there are any changes.

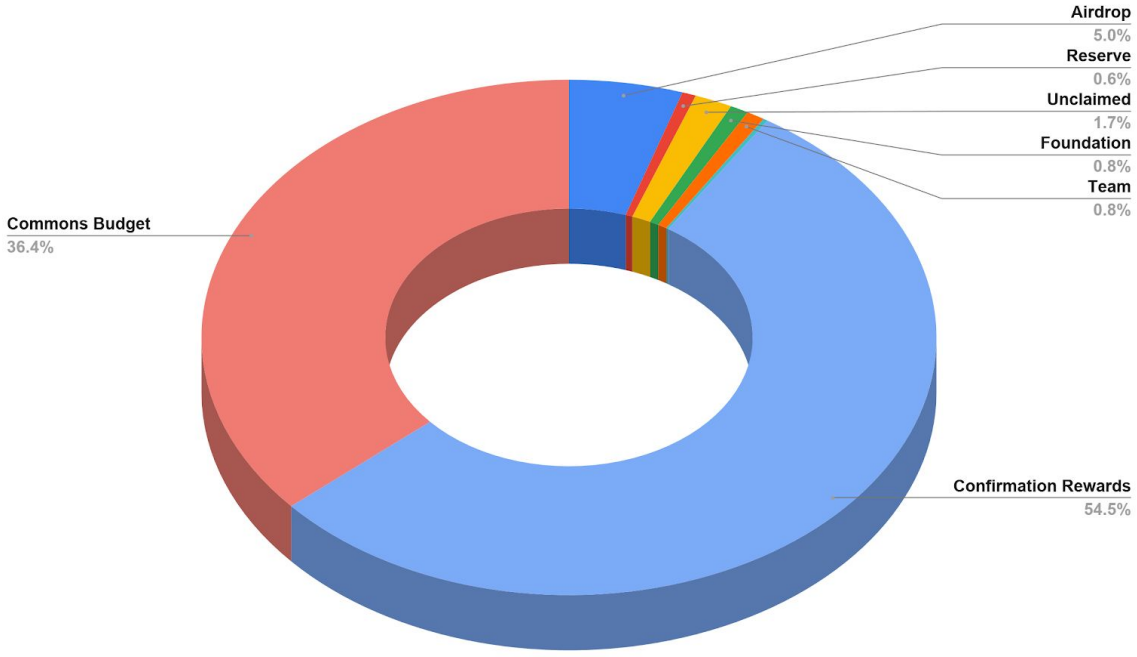

Fig 4 : BOA Coin Issuance Plan

Issuance

New coins are issued in three ways; Initial Development Budget(0.45bil, 10%), confirmation rewards(2.7bil, 54%), and the Commons Budget(1.8bil, 36%). We aim to issue a total of 4.95 billion coins over the next 100 years. These values are subject to change.

- **Initial Development Budget:** Initial development coins are coins distributed prior to the Genesis block are intended to support the final development of the software. These coins are made up of airdrops and bounties. 450 million BOA are issued with the Genesis block.
- **Confirmation Rewards:** Confirmation rewards are financial rewards issued and evenly distributed to the nodes for every confirmed block. As the reward is distributed evenly, if the number of nodes increases the probability that a node will receive a

reward decreases. This reward is relative to the number of coins frozen in a node. 2.7 billion BOA are issued through Confirmation rewards. Initially 27 BOA are issued per 5 seconds. The reward decreases every -roughly- one year by 6.31% over 128 years.

- **Commons Budget:** The Commons Budget holds BOA that can only be used by proposals that have passed the Congress Network. In order to create a sufficient budget for proposals, 50 Commons Coins are issued per 5 seconds for the first -roughly- six years. After the first six years the Commons Budget is maintained through the 30% commons fee on transactions fees.

After the mainnet launch, there will be confirmation rewards for validators, and the commons budget which will be issued.

The complete token issuance chart is attached at the end of this document.

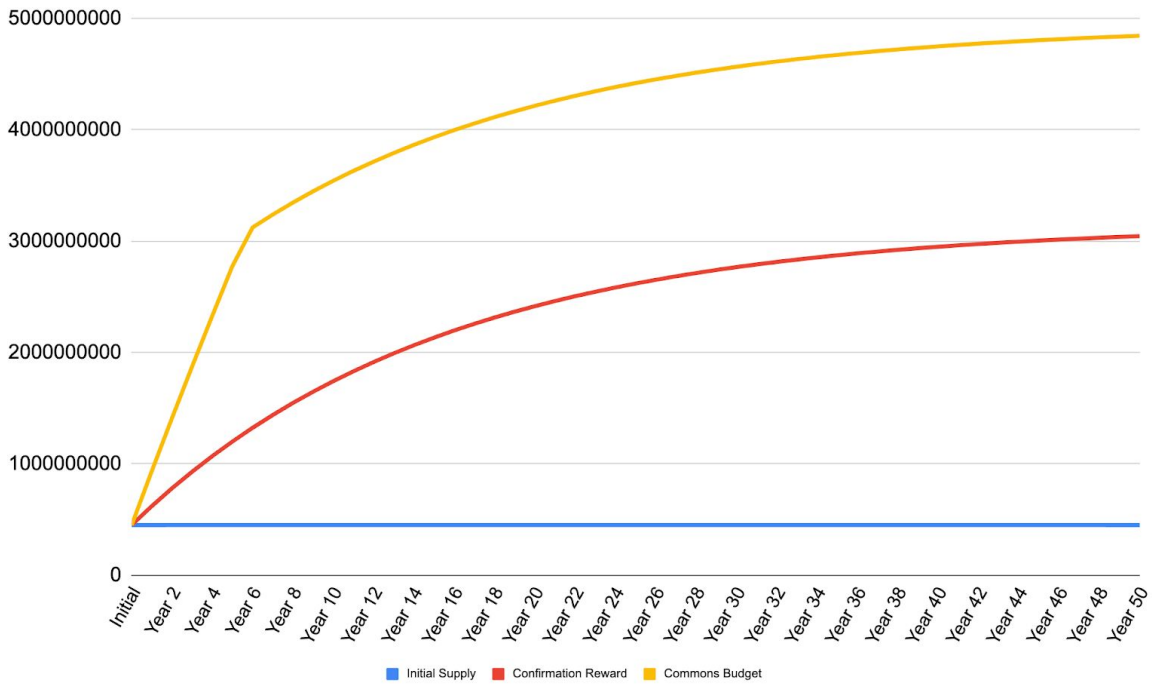

Fig 5 : BOA Coin Issuance Schedule

Technology

Abstract

Bitcoin introduced the world to the idea of digital bearer's money. By implementing a virtual equivalent to cash, it laid the foundation for a multi-billion dollar industry where the properties of money are discussed and challenged.

The approach originally used is a timestamping server where inclusion of a batch of operations is based on expanded computing power. While providing very attractive properties, the usage of Proof-of-Work (PoW) turned out to be extremely energy inefficient. Additionally, the mining approach led to the development of specific hardware, centrally hosted in areas with lower energy cost, threatening the system with greater centralization. While alternatives have been developed for the latter, the mining approach is inherently wasteful.

We propose a construction where consensus is achieved at a low cost, in a self-contained system where penalties are applied to provably misbehaving actors. Such a self-contained system is usually referred to as “Proof-of-Stake” (PoS), although we provide our own definition of what that means.

We start by exploring the assumptions under which any system should operate to be safe, then define the properties we seek and how those compare to Bitcoin. Additionally, we explore the current state of the art development on PoS (most notably Ethereum’s research) and PoS-specific attack.

I. Introduction

Proof-of-Work resource consumption

As mentioned in the Bitcoin white paper [Nak09] the main problem an electronic cash system face is double spend. While proof-of-work (PoW) was a powerful tool in democratizing the concept of electronic cash, it led to the development of special hardware and large, wasteful consumption of energy. As a result, mining operations have been largely centralized in areas offering cheap electricity. As of June 2018, it was estimated that around 74% of the hash-rate was operated by Chinese entities [KJL18]. Additionally, most of the specialized hardware (ASIC) is being developed in China, which makes the currency vulnerable to Chinese regulators.

Proof-of-Stake

There is not currently any replacement for PoW that exhibit the same properties. One contender is thought to be Proof-of-Stake (PoS). Many projects have explored the problem, starting from Peercoin in 2012 [KN12] with its “coin age” approach. Another well-established coin, NXT, uses an approach where the newly-created block data is used as a seed to decide on the next selector [NXT19]. The most prominent project working on a PoS system, Ethereum, has been planning to transition to PoS since its inception in 2014. Over the past few years, new projects have introduced the notion of “Delegated PoS” (DPoS), where nodes votes to delegate their voting power to a small subset of nodes.

One major difference between PoS and PoW protocol is that the former favors safety over liveness, resulting in protocols that can be stopped, but have instant finalization, while the latter provide guaranteed liveness and exponential safety. However, PoW does not provide

meaningful liveness: an attacker with enough resources could decide to produce empty blocks, effectively rendering the system useless. At the time of writing, the missing income would be marginal compared to the block reward, and could be easily compensated by external actors.

Nonetheless, regardless of any safety aspects any attack would likely result in the devaluation of the currency, which in many cases is a strong enough incentive to prevent attack. The blockchain community has fully embraced this, and game theory has been an essential part of the analysis of consensus protocol since day 1 [GTB19].

Scalability issues

Additionally to the inherent waste of resources that PoW system represent, blockchain scalability is a topic of active research. Even in the Bitcoin community, multiple factions have emerged: namely, Bitcoin has stayed with a 1 MB block limit (although Segregated Witness [SegWit] helped augment the capacity of the chain), while Bitcoin Cash has increased its block size to 32 MB. The argument in favor of small blocks is that only full nodes (nodes that verify the blockchains completely) are secure, while the others rely on other parts of the system (e.g. miners), and thus a personal computer should be able to run a bitcoin node. With 32 MB per block, at 1 block / 10 minutes (144 blocks per day), the amount of data that can be accepted is 4.6 GB daily, 138 GB monthly and 1,659 GB yearly.

One inherent requirement of a decentralized network is that a transaction has to be confirmed by a majority of the system to be considered accepted. Additionally, the more nodes that participate in the network, the more decentralized the network is, provided nodes are not controlled by colluding entities. Thus, a system which gains more users will put more stress on each node, leading to higher hardware and bandwidth requirements. On the other hand, when the number of nodes increases, every transaction will need to reach more nodes, increasing the time it takes for a transaction to be confirmed.

Instead of attacking this multi-objective optimization problem, we decided to follow the same track as Bitcoin core: building a layer(L2 / “Flash Layer”) on top of the blockchain layer (L1 / “Settlement Layer”), with slightly weaker rules, allowing to get most of the L1 safety while allowing transactions to be accepted by peers without needing to record them to L1 (and, by extension, forward them to all nodes). In doing so, we integrate some incentives for clients to use the scheme by default, and for nodes to accept such transactions.

The benefits of the Flash Layer solution are:

- Less data on the blockchain;
- Confirmation time is “almost instantaneous” if the protocol is followed;
- Users do not have to wait for a block for confirmation;
- Cheaper fee for the micro transactions that occur within the Flash Layer;

With the benefits, we expect that the built-in second layer solution will bring a secure and a low-cost dapp development environment. Additionally, one of the most important goal of BOSAGORA project, the separation of the political and the economic power will be realized with the Flash Layer solution.

Summary

In **Section II**, we will explore attacks on a PoS system.
In **Section III**, we introduce the foundations on our approach: the network model, source of randomness, and a scheme for validators to sign blocks in an efficient manner.
In **Section IV**, we introduce a consensus protocol where participants (“validators”) lock a specific amount (“stake”) in order to participate in the consensus protocol, and the incentive scheme is designed to encourage correct collaboration in the network.
In **Section V**, we describe our layer 2 approach and its integration with layer 1. This section will be developed further later.

II. Attacks on PoS

A few attacks and concerns on PoS have been discussed over the years. In this section, we will go over the basic definitions of such attack in an effort to enlighten the reader on challenges our protocol will face.

Short & long range attacks

We define short range attacks as attacks happening on clients that are less than N blocks behind the latest network-accepted block, while long range attacks are targeting clients more than N blocks behind the latest network-accepted block.
N is a parameter of the consensus protocol which can be explicitly chosen or derives from other factors. An example of an explicit definition of N can be found in Ethereum’s concept of weak subjectivity [VB14].

Due to the low computational cost associated with creating blocks, an adverse entity with access to past private keys could create a competing chain without much associated costs. As keys are essentially worthless after the coins they control have been moved, it would be economically viable for validators to sell their private keys after exiting the system.

Stake grinding attacks

Grinding attacks arise when part of the consensus algorithm depends on a random factor. Since the consensus protocol cannot rely on data it cannot verify (this would induce trust, and by extension a single point of failure), any randomness must be based on a known, predictable process, and data available to all participants, which is at odds with the traditional approach to randomness.
Since data is publicly available, attackers could attempt to influence it in a way that would be more favorable to them.

For example, a naive consensus protocol would have the following steps:

- Select a fixed set of n validators;
- Order this set in a predictable way (e.g. according to their public key);
- Every round, pick a validator to nominate a block;
- The validator selected has index Hash(previous_block) % n in the ordered set;

With such an approach, a validator would simply have to find a single suitable hash to be elected as the next round's validator. In the random oracle model, for n = 100, 1000 combinations would give a validator > 99.99% chances to be the next validator.

An approach that is often cited to solve such an issue is requiring blinded pre-commit. For example, validators would commit a hash during round R, and reveal the preimage of this hash on round R + 1. The random value (or seed for it) would then be the sum (or XOR, or hash of concatenation) of those preimages.

Nothing at stake attacks

Nothing at stake attacks were present in early design of PoS protocol.
When a validator is presented with two different blocks which are both valid candidates for the current chain, the most economically viable behavior would be to "vote" on both of them, since "voting" on a chain consumes no resources [VB14]. This led to consensus protocol adding penalization of such behaviors.
However, such penalties are inefficient if they are not combined with a mandatory lock-in period. If validators are able to move (sell) their stake at any time, including directly after voting on a block, it would be trivial for them to move their stake, then attempt to double spend a previously spend output from a block where they still had stake.
There would be no way to penalize such behavior, as the stake would already belong to another party. For this reason, lock-in period are introduced.

III. Fundamentals

Network model

Out of all 3 available network models (Synchronous, Asynchronous, Partially synchronous), we position ourselves in the synchronous model [DLS88], as a result of SCP's requirement, which is a synchronous protocol.
A well known result of consensus research is that no protocol can have liveness (ensuring that the network makes progress), safety (ensuring that all participants reach the same result) and fault tolerance (ensuring that the network can safely make progress if one or more nodes are not responding). This result is called the FLP impossibility [FLP85] and is heavily referenced by the SCP paper, which chooses to favor safety over liveness. Fault tolerance, on the other hand, is a requirement for any system with open membership.

Source of randomness

Some parts of this paper, such as the signature scheme, rely on pseudo random data. Because randomness is by nature unpredictable, and hence cannot be verified for correctness, ensuring randomness in a distributed system faces needs to rely on seed data provided by all participants. A resulting challenge is ensuring no participant can gain an edge over any other participant by crafting or delaying its seed data.

This is achieved by using a hash and its preimage as seed data.

Upon enrollment, validators pick a random value, hash it n times, and commit the final value as their initial seed data. Every time a new seed data is required, validators can reveal the preimage of their last-published seed data, thus ensuring true randomness without the ability to manipulate data.

However, an issue arises when a validator willingly withhold data from the network. If publishing the data leads to a worse outcome than withholding it, then a node can choose to selectively withhold its preimage, either stopping the network or skewing the result. To avoid this pitfall, validators should regularly publish (and listening validator should support) enough seed data to survive a minor outage.

If sensible intervals are introduced in the consensus protocol, validators can be guaranteed that publishing their preimage ahead of time will not result in weakening the safety guarantees, and allowing them to cope with temporary downtime.

Enrollment process

When registering as a validator, a node broadcast the following data:

- **K** (UTXO key): A public key matching a frozen UTXO;
- **X** (random seed): The nth image of their private key;
- **n** (cycle length): the number of rounds a validator will participate in (currently fixed to (**freezing period** / 2)];
- **R** (signature noise): The initial nonce used for signing (see 3. Validator signature scheme);
- **S**: A signature for the message H(***K, X, n, R***) and the key **K**, using **R**.

After its registration is recorded, a validator is expected to start signing blocks immediately, as described in IV.1.

The following are requirements the UTXO controlled by **X** must satisfy in order to qualify for enrollment:

- It has at least 40,000 coins;
- It did not default in the last **freezing period** blocks;

Validators signature scheme

Validators signal their commitment to a block by signing the hash of this block.
Signatures can be combined efficiently, so that in the best case scenario (all validators sign), the signature is the combined signature of all validators, taking O(1) space.
The scheme used is based on Schnorr signatures, and is described below.

We define the following notations:

- H() is a hash function;
- Given the pair (k, K):
- k is a value in the group G of prime order P;
- K is the exponentiation of the base point B of the elliptic curve used by k;
- The pair (k, K) is used for the private/public key pair, respectively;

- The pair (r, R) is a unique random value and its exponentiation;

IV. Layer 1 Protocol

Cycle & Consensus rounds

We consider the consensus protocol as being a succession of simultaneous **cycles**, undertaken by each participant individually. Participants are called **Validators**, while observers of the consensus protocol are called **Nodes**. While every validator is a node, not every node is a validator.

Each cycle has a length (**n**) known at the beginning of the cycle. This length is expressed in terms of consensus round, with the output of each consensus round being a block, itself being primarily defined by the set of transactions being selected. Each round is expected to last in the range of (a few)minutes(to be defined by experimentation during testnet). Each round, the value of n decreases by one, and the cycle is over when the value reaches 0.

Cycles are dependent on the freezing capability of UTXO. A malicious actor would have a strong incentive to revert blocks right after it exited its validator capacity, if it was able to immediately trade the stake that was used for validation. As a result, the stake used for validation is frozen, and the **freezing period** is fixed to 14 days.

In order for a node to become a validator, and begin a cycle, it must complete the **enrollment process**. This is done by selecting a number of round n suitable for the entity, within the bounds defined in III.3, and propagating that message to existing validators.

Once that transaction is registered, a node immediately becomes a validator, collecting and propagating transactions. However, when a node originally enroll, it is not yet assigned a quorum set and is expected to be **passive** (sign blocks only when they reach the 50% threshold) until the next quorum balancing event happens.

Quorum balancing events happens once every 1 hour. When a quorum balancing event happens, the network is re-organized in a pseudo-random but predictable manner to ensure fairness in the reward process and prevent collusion between validators.

Towards the end of every round, nodes initiate a nomination process as defined by SCP [SCP16]. The leader selects a set of transactions and elects them according to the roles defined in the SCP paper. In the future, we aim to replace this nomination protocol by a protocol based on our source of randomness.

The result of the SCP round is a block that is signed by a majority of registered participants.

Preimage availability

Enrolled validators should always make sure their preimage is available to other nodes in a timely manner. As some aspects of validation / quorum balancing are dependent on

preimages, any node that is not able to provide a preimage before it is needed (usually, the end of a consensus round). Validators can make their preimage available by broadcasting a message comprising of the preimage at a certain round and the round number, such as: (**P, Nx**), where **P** is the preimage and **Nx** is: **n - (rounds since enrollment)**.
Should the network miss a preimage, a node is said to have **defaulted**. Such node will not be able to re-enroll for consensus nor unfreeze their stake for a set period of time.

Nomination protocol

Nomination is the act of selecting a set of transactions as candidates for inclusion in the next block. Since multiple participants in the network might have a different set of transactions, this task is often relegated to a single node.
In Bitcoin, this node is the miner. In most other consensus protocol, there is a leader election who decides on the set of transactions. Currently, BOSAGORA relies on SCP's nomination protocol, which is based on a quorum leader election.
However, the presence of an unbiased source of randomness enables us to build a filter to make building a set of transactions more predictable, and more importantly, verifiable.

Such a change, while desirable, is left as a future improvement to the protocol.

Quorum balancing event

Quorum assignment is done to reduce the overhead of communication between nodes. Provided quorum assignment is essentially splitting the network into smaller, yet overlapping network, the main challenge is to provide a configuration which minimizes communication without compromising safety.

The quorum balancing event is currently being designed by our team and requires experimentation, and as such will be subject to changes in a later revision.

Reward allocation

Reward allocation follows the structure outlined previously in this whitepaper. A total of 27 coins are initially issued per 5 seconds, and distributed evenly to the validators when a new block is generated. A decreasing rate is applied over a fixed period.

Conclusion

The BOSAGORA team aims to overcome the technical and operational issues inherent in many cryptocurrencies. The incentive scheme and issuance plan is aimed towards creating value for the coin while deterring the centralization of power. The Modified Federated Byzantine Agreement algorithm will allow for low latency transactions while being more energy efficient. The Congressional System is aimed towards creating a more democratic and productive decision making process. Trust contracts will provide a decidable and approachable framework for creating and executing contracts on the blockchain. The BOSAGORA team will aim to achieve these goals while leveraging the security and integrity that can be gained through blockchain technology.

Appendix 1 : What is Federated Byzantine Agreement

In 2015, Professor David Mazieres, head of Stanford's Secure Computer Systems Group, introduced an alternative to pBFT called the Stellar Consensus Protocol, or Federated Byzantine Agreement(hereinafter FBA), a decentralized alternative to existing consensus protocols such as PoW or pBFT.

The consensus protocol is likely to require some extensions in order to make it fully decentralized and open. However, FBA has a proven track record of technical excellence and is unlikely to change. FBA, short for Federated Byzantine Agreement, powers Stellar, the 13th biggest cryptocurrency, with a market capitalization of over 900 million dollars.

Federated Byzantine Agreement can be described by the following:
A network consisting of quorums, and each quorum is a set of nodes sufficient to reach an agreement. FBA also introduces the concept of a quorum slice, the subset of a quorum that can convince one particular node of agreement. The consensus process is achieved via the quorums, and the collective agreement of the quorums is used as the final decision of the entire network despite byzantine failure.

Pros about FBA

There are two main features that FBA is suitable for BOSAGORA consensus protocol.

First, the confirmation of the transaction by the consensus protocol gets finalized in a few seconds. Unlike PoW, there is no mining process which means there should not be much computing power involved to reach an agreement. The agreement happens during the data passing within the voting process. Also, there is no need to validate every single node's data but validate the result of voting of the quorums. As a utility coin, the confirmation speed and low latency are critical to be utilized in a real-life environment.

Second, the membership mechanism of the network is open to the public. In FBA, there is no validator list chosen by someone or an organization. Rather, each validator decides which other validators they trust, and their list of trusted validators is called their quorum slice. The quorum slices of each validator overlap to form a quorum or network-wide consensus on a transaction. Because of the character of the FBA network, anyone can spin up a validator and participate in consensus if any other participating validator adds you to their quorum slice.

Like Bitcoin, we can expect validators joining and leaving the network without much impact on consensus. Currently, the Stellar network is the biggest network utilizing FBA. There is an argument that Stellar network is not yet as decentralized as, say, Bitcoin. But it's important to note that it's construction inherently allows for growing decentralization (unlike PBFT) as

more and more nodes are added to the network and new quorum slices form. Therefore, this will lead the entire network to a more decentralized network as we wanted.

Openness

Although FBA is pursuing an open network to the public, it still has its shortcomings. For example, to be a validator node, it should have its own quorum set, so the validating node per account can participate in the consensus process. However, if a new node attempting to join the network and declares itself as a validator then composes their own quorum set, but if the existing validators do not accept the new node into their quorum set, the decision of the new node will not be received by the other validators. This will lead the new nodes can not participate in the decision process.

The SCP (Stellar Consensus Protocol) states that anybody can operate as a node and can join the Stellar network, which characterizes the network to be open. But this is only half correct. Although joining the network is open to anyone, to join the network and participate in the consensus process as a validator is limited. Currently, in the SCP network, to join as a validator, the existing validators must approve and accept the new node. In other words, everyone who wants to join the network needs permission from someone.

Appendix 2 : Trust Contracts

The original white paper explains trust contract as following :
“Trust Contracts are securely executable contracts based on a protocol layer called Owlchain, which consists of the Web Ontology Language and the Timed Automata Language. Trust Contracts aim to overcome the issues regarding non-decidable smart contracts by using a more contained and comprehensible programming framework, which provides secure and decidable transactions of contracts.”

The ultimate goal of this architecture is to be able to build a decidable contract, which ensures safe and accurate execution while maximizing its scalability.
To achieve the goal, the original white paper mentions two methodologies. One is through using a flexible programming language on a virtual machine, the other is to use a slightly less flexible but decidable domain-specific language. The original plan was going with the second option.

The initial development team(BlockchainOS) researched the inference engine based on semantic web technology. However, there was no result from the research nor discovery of method or technology to overcome the issue.

“An ontology is an explicit specification of a conceptualization. The term is borrowed from philosophy, where an ontology is a systematic account of Existence. In knowledge-based systems, what “exists” is exactly that the contents that can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. The set of objects and the describable relationships among them are reflected in the representational vocabulary with which a knowledge-based program represents knowledge. Thus, we can describe the ontology of a program by defining a set of representational terms.”[9]

“An ontology is a formal, explicit specification of a shared conceptualization of a domain of interest.”

The ontology has been researched and developed in artificial intelligence and the natural language processing field for a long time. It enables the computers to understand the information that is given from the relationships and definitions. On this basis, the computers will eventually infer the requested information.

However, building an ontology in the real-world takes a lot of effort and time. It is not only difficult for the decentralized general public to work on the meaning of the inference engine until the function of the inference engine is completed, but it is also limited in the use of the

[9] A Translation Approach to Portable Ontology Specifications :
https://pdfs.semanticscholar.org/5120/f65919f77859a974fcc1ad08f72b2918b8ec.pdf

engine and verify the results according to the input value. And there is no commercialized technology that can make inference engines easy for different situations.

It's even more difficult to automate Time Automata Language-based verification by analyzing semantic source codes that are implemented in OWL. As the complexity of the source code increases, the number of states increases exponentially, making it almost impossible to verify. When OWL is used to write a contract, it will require the complex and detailed specification to produce a realistic contract and if the process of creating a predictable and realistic contract is too difficult to verify, it will be challenging to be used by the users.

BOSAGORA team aims to implement Trust Contracts, which enable a safe, accurate, programmable and executable contract as in the original intention.

Rather than continuing what is not feasible, we will redefine 'Trust Contracts' and will actively pursue the selecting the optimal direction and applying the suitable technology to improve the core protocol. This approach also considers adopting a methodology that uses flexible programming language on top of virtual machines and we are currently exploring WebAssembly as other industry players do. In the end, we intend to provide an efficient, safely designed smart contract engine and provide an easy-to-develop language with many tools and popularity for easy adoption by developers.

WebAssembly is a new type of code that can be run in a modern web browser. It provides new capabilities and offers significant performance benefits. “WebAssembly is a binary instruction format for a stack-based virtual machine. WebAssembly is designed as a portable target for compilation of high-level languages like C/C++/Rust, enabling deployment on the web for client and server applications.”[10]

Running programs that are written in multiple languages on the web at near-native speeds using client applications was previously impossible. Running the codes on WebAssembly is similar to the actual hardware. With WebAssembly, developers can code in a variety of programming languages such as C++ and Rust, and they can expect to run the program in near-native performance. EOS also uses WebAssembly, and many blockchain platforms such as Ethereum, Tron and Cardano have already introduced or plan to introduce the virtual machines using WebAssembly.

Once the feasibility of the implementation plan has been studied and the solution has been discovered, technical and practical measures will be taken to complete the objectives and directions for the "Trust Contracts" presented in the white paper.

[10] WebAssembly, https://webassembly.org

Appendix 3 : Token Issuance Schedule Chart

	Confirmation Reward	Circulating Supply	Commons Budget	Total supply
Initial	0	450,000,000	0	450,000,000
Year 1	170,294,400	620,294,400	315,360,000	935,654,400
Year 2	159,548,823	779,843,223	315,360,000	1,410,563,223
Year 3	149,481,293	929,324,516	315,360,000	1,875,404,516
Year 4	140,049,023	1,069,373,539	315,360,000	2,330,813,539
Year 5	131,211,930	1,200,585,469	315,360,000	2,777,385,469
Year 6	122,932,457	1,323,517,926	223,200,000	3,123,517,926
Year 7	115,175,419	1,438,693,345		3,238,693,345
Year 8	107,907,850	1,546,601,195		3,346,601,195
Year 9	101,098,865	1,647,700,060		3,447,700,060
Year 10	94,719,526	1,742,419,586		3,542,419,586
Year 11	88,742,724	1,831,162,310		3,631,162,310
Year 12	83,143,058	1,914,305,368		3,714,305,368
Year 13	77,896,731	1,992,202,099		3,792,202,099
Year 14	72,981,448	2,065,183,547		3,865,183,547
Year 15	68,376,318	2,133,559,865		3,933,559,865
Year 16	64,061,773	2,197,621,638		3,997,621,638
Year 17	60,019,475	2,257,641,113		4,057,641,113
Year 18	56,232,246	2,313,873,359		4,113,873,359
Year 19	52,683,991	2,366,557,350		4,166,557,350
Year 20	49,359,631	2,415,916,981		4,215,916,981
Year 21	46,245,039	2,462,162,020		4,262,162,020
Year 22	43,326,977	2,505,488,997		4,305,488,997
Year 23	40,593,044	2,546,082,041		4,346,082,041
Year 24	38,031,623	2,584,113,664		4,384,113,664
Year 25	35,631,828	2,619,745,492		4,419,745,492
Year 26	33,383,460	2,653,128,952		4,453,128,952
Year 27	31,276,963	2,684,405,915		4,484,405,915
Year 28	29,303,387	2,713,709,302		4,513,709,302
Year 29	27,454,343	2,741,163,645		4,541,163,645
Year 30	25,721,974	2,766,885,619		4,566,885,619
Year 31	24,098,918	2,790,984,537		4,590,984,537
Year 32	22,578,276	2,813,562,813		4,613,562,813

Year 33	21,153,587	2,834,716,400		4,634,716,400
Year 34	19,818,795	2,854,535,195		4,654,535,195
Year 35	18,568,229	2,873,103,424		4,673,103,424
Year 36	17,396,574	2,890,499,998		4,690,499,998
Year 37	16,298,850	2,906,798,848		4,706,798,848
Year 38	15,270,393	2,922,069,241		4,722,069,241
Year 39	14,306,831	2,936,376,072		4,736,376,072
Year 40	13,404,070	2,949,780,142		4,749,780,142
Year 41	12,558,273	2,962,338,415		4,762,338,415
Year 42	11,765,846	2,974,104,261		4,774,104,261
Year 43	11,023,421	2,985,127,682		4,785,127,682
Year 44	10,327,843	2,995,455,525		4,795,455,525
Year 45	9,676,156	3,005,131,681		4,805,131,681
Year 46	9,065,591	3,014,197,272		4,814,197,272
Year 47	8,493,552	3,022,690,824		4,822,690,824
Year 48	7,957,609	3,030,648,433		4,830,648,433
Year 49	7,455,484	3,038,103,917		4,838,103,917
Year 50	6,985,043	3,045,088,960		4,845,088,960
Year 51	6,544,287	3,051,633,247		4,851,633,247
Year 52	6,131,342	3,057,764,589		4,857,764,589
Year 53	5,744,454	3,063,509,043		4,863,509,043
Year 54	5,381,979	3,068,891,022		4,868,891,022
Year 55	5,042,376	3,073,933,398		4,873,933,398
Year 56	4,724,203	3,078,657,601		4,878,657,601
Year 57	4,426,105	3,083,083,706		4,883,083,706
Year 58	4,146,818	3,087,230,524		4,887,230,524
Year 59	3,885,154	3,091,115,678		4,891,115,678
Year 60	3,640,001	3,094,755,679		4,894,755,679
Year 61	3,410,317	3,098,165,996		4,898,165,996
Year 62	3,195,126	3,101,361,122		4,901,361,122
Year 63	2,993,513	3,104,354,635		4,904,354,635
Year 64	2,804,623	3,107,159,258		4,907,159,258
Year 65	2,627,651	3,109,786,909		4,909,786,909
Year 66	2,461,846	3,112,248,755		4,912,248,755
Year 67	2,306,504	3,114,555,259		4,914,555,259
Year 68	2,160,963	3,116,716,222		4,916,716,222
Year 69	2,024,606	3,118,740,828		4,918,740,828

Year 70	1,896,854	3,120,637,682		4,920,637,682
Year 71	1,777,162	3,122,414,844		4,922,414,844
Year 72	1,665,023	3,124,079,867		4,924,079,867
Year 73	1,559,960	3,125,639,827		4,925,639,827
Year 74	1,461,527	3,127,101,354		4,927,101,354
Year 75	1,369,305	3,128,470,659		4,928,470,659
Year 76	1,282,901	3,129,753,560		4,929,753,560
Year 77	1,201,950	3,130,955,510		4,930,955,510
Year 78	1,126,107	3,132,081,617		4,932,081,617
Year 79	1,055,050	3,133,136,667		4,933,136,667
Year 80	988,476	3,134,125,143		4,934,125,143
Year 81	926,103	3,135,051,246		4,935,051,246
Year 82	867,666	3,135,918,912		4,935,918,912
Year 83	812,917	3,136,731,829		4,936,731,829
Year 84	761,622	3,137,493,451		4,937,493,451
Year 85	713,563	3,138,207,014		4,938,207,014
Year 86	668,537	3,138,875,551		4,938,875,551
Year 87	626,353	3,139,501,904		4,939,501,904
Year 88	586,830	3,140,088,734		4,940,088,734
Year 89	549,801	3,140,638,535		4,940,638,535
Year 90	515,108	3,141,153,643		4,941,153,643
Year 91	482,605	3,141,636,248		4,941,636,248
Year 92	452,153	3,142,088,401		4,942,088,401
Year 93	423,622	3,142,512,023		4,942,512,023
Year 94	396,891	3,142,908,914		4,942,908,914
Year 95	371,847	3,143,280,761		4,943,280,761
Year 96	348,384	3,143,629,145		4,943,629,145
Year 97	326,401	3,143,955,546		4,943,955,546
Year 98	305,805	3,144,261,351		4,944,261,351
Year 99	286,509	3,144,547,860		4,944,547,860
Year 100	268,430	3,144,816,290		4,944,816,290
Year 101	251,492	3,145,067,782		4,945,067,782
Year 102	235,623	3,145,303,405		4,945,303,405
Year 103	220,755	3,145,524,160		4,945,524,160
Year 104	206,825	3,145,730,985		4,945,730,985
Year 105	193,775	3,145,924,760		4,945,924,760
Year 106	181,548	3,146,106,308		4,946,106,308

Year 107	170,092	3,146,276,400		4,946,276,400
Year 108	159,359	3,146,435,759		4,946,435,759
Year 109	149,304	3,146,585,063		4,946,585,063
Year 110	139,883	3,146,724,946		4,946,724,946
Year 111	131,056	3,146,856,002		4,946,856,002
Year 112	122,786	3,146,978,788		4,946,978,788
Year 113	115,038	3,147,093,826		4,947,093,826
Year 114	107,780	3,147,201,606		4,947,201,606
Year 115	100,979	3,147,302,585		4,947,302,585
Year 116	94,607	3,147,397,192		4,947,397,192
Year 117	88,637	3,147,485,829		4,947,485,829
Year 118	83,044	3,147,568,873		4,947,568,873
Year 119	77,804	3,147,646,677		4,947,646,677
Year 120	72,895	3,147,719,572		4,947,719,572
Year 121	68,295	3,147,787,867		4,947,787,867
Year 122	63,986	3,147,851,853		4,947,851,853
Year 123	59,948	3,147,911,801		4,947,911,801
Year 124	56,165	3,147,967,966		4,947,967,966
Year 125	52,621	3,148,020,587		4,948,020,587
Year 126	49,301	3,148,069,888		4,948,069,888
Year 127	46,190	3,148,116,078		4,948,116,078
Year 128	43,275	3,148,159,353		4,948,159,353

Reference

The BOSAGORA White Paper, https://bosagora.io/

WebAssembly, https://webassembly.org/

A Translation Approach to Portable Ontology Specifications :
https://pdfs.semanticscholar.org/5120/f65919f77859a974fcc1ad08f72b2918b8ec.pdf

The DAO, https://slock.it/dao.html

David Mazieres, Stellar Consensus Protocol,
https://www.stellar.org/papers/stellar-consensus-protocol.pdf

Andrychowicz, Dziembowski, Malinowski and Mazurek, Modeling Bitcoin Contracts by Timed Automata, Lecture Notes in Computer Science Formal Modeling and Analysis of Timed Systems, 7-22, 2014, https://arxiv.org/pdf/1405.1861v2.pdf

David Mazieres, Stellar Consensus Protocol,
https://www.stellar.org/papers/stellar-consensus-protocol.pdf

Decentralized Prediction Market, https://www.augur.net/

Evan Duffield, Daniel Diaz, Dash: A PrivacyCentric CryptoCurrency,
https://www.dash.org/wp-content/uploads/2015/04/Dash-WhitepaperV1.pdf

Golem, https://golem.network

Hodges, Andrew, Alan Turing: the enigma, London: Burnett Books

Ian Grigg, The Ricardian Contract, First IEEE International Workshop on Electronic Contracting (WEC) 6th July 2004, http://iang.org/papers/ricardian_contract.html

Leading the Pack in Blockchain Banking: Trailblazers Set the Pace,
https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBP03467USEN&

N. Atzei, M. Bartoletti, T. Cimoli, A survey of attacks on Ethereum smart contracts,
https://eprint.iacr.org/2016/1007.pdf

Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System,
https://bitcoin.org/bitcoin.pdf

Simple Declarative Language, https://sdlang.org/

The DAO, https://slock.it/dao.html

Using Decentralized Governance: Proposals, Voting, and Budgets, https://dashpay.atlassian.net/wiki/display/DOC/Using+Decentralized+Governance%3A+Proposals% 2C+Voting%2C+and+Budgets

OWL Web Ontology Language, https://www.w3.org/TR/owl-features/

OWL Web Ontology Language Reference, https://www.w3.org/TR/owl-ref

Vitalik Buterin, Ethereum Whitepaper, https://github.com/ethereum/wiki/wiki/White-Paper

De Filippi, P. & Loveluck, B. (2016) The invisible politics of Bitcoin: governance crisis of a decentralised infrastructure. Internet Policy Review, 5(3). Retrieved March 18, 2018 from https://policyreview.info/articles/analysis/invisible-politics-bitcoin-governance-crisis-decentralised- infrastructure

Ehrsam F. (2017) Blockchain Governance: Programming our future. Retrieved March 18, 2018 from
https://medium.com/@FEhrsam/blockchain-governance-programming-our-future-c3bfe30f2d74

Albert O. Hirschman. 1970. Exit, Voice, and Loyalty: Responses to Decline in Firms, Organizations, and States.Cambridge, MA: Harvard University Press. Retrieved March 18, 2018

Duncan L. (2017) Thoughts on Governance and Network Effects. Medium. Retrieved March 18, 2018 from
https://blog.aragon.one/thoughts-on-governance-and-network-effects-f40fda3e3f98

Surowiecki J. (2005) The Wisdom of Crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations. Anchor. Retrieved March 18, 2018
Homomorphic Encryption Standardization homepage, Retrieved March 18, 2018 from http://homomorphicencryption.org/introduction/

Bernhard D., Warinschi B. (2014) Cryptographic Voting — A Gentle Introduction. In: Aldini A., Lopez J., Martinelli F. (eds) Foundations of Security Analysis and Design VII. Lecture Notes in Computer Science, vol 8604. Springer, Cham, Retrieved March 18, 2018 from https://link.springer.com/chapter/10.1007/978-3-319-10082-1_7

B. Thiyaneswaran, S. padma. (2012) Iris Recognition Using left and right Iris feature of the Human Eye for Bio-metric Security system.

IJCA, vol 50 No. 152. Retrieved March 18, 2018 from http://www.gjimt.ac.in/wp-content/uploads/2017/11/Vijay-Kumar-Sinha_Enhancing-Iris-Security-by-Detection-of-Fake-Iris_Paper.pdf

Zyskind, Nathan, Pentland (2016) Decentralizing Privacy: Using Blockchain to Protect Personal Data. Retrieved March 18, 2018 from https://enigma.co/ZNP15.pdf

Fujioka A., Okamoto T., Ohta K. (1993) A practical secret voting scheme for large scale elections. In: Seberry J.,

Zheng Y. (eds) Advances in Cryptology — AUSCRYPT '92. AUSCRYPT 1992. Lecture Notes in Computer Science, vol 718. Springer, Berlin, Heidelberg. Retri eved March 18, 2018 from https://link.springer.com/chapter/10.1007/3-540-57220-1_66

Çetinkaya O., Doganaksoy A. (2007) A Practical Verifiable e-Voting Protocol for Large Scale Elections over a Network, Availability Reliability and Security 2007. ARES 2007. The Second International Conference on, pp. 432-442, 10–13 April 2007. Retrieved March 18, 2018 from http://ieeexplore.ieee.org/document/4159833/

Dash : Using Decentralized Governance: Proposals, Voting, and Budgets, https://dashpay.atlassian.net/wiki/display/DOC/Using+Decentralized+Governance%3A+Proposals%2C+Voting%2C+and+Budgets

Understanding Dash Governance
https://docs.dash.org/en/stable/governance/understanding.html

Dmytro Kaidalov, Andrii Nastenko, Oleksiy Shevtsov, Mariia Rodinko, Lyudmila Kovalchuk, Roman Oliynykov (2016) A Review of the Dash governance system
https://api.zotero.org/groups/478201/items/BJUUEE9Q/file/view?key=Qcjdk4erSuUZ8jvAah59Asef

Bingsheng Zhang, Roman Oliynykov, Hamed Balogun (2017) A Treasury System for Cryptocurrencies: Enabling Better Collaborative Intelligence
https://www.lancaster.ac.uk/staff/zhangb2/treasury.pdf?utm_content=buffer7118a&utm_medium=social&utm_source=twitter.com&utm_campaign=buffer

[Nak09] Satoshi Nakamoto. (2009). Bitcoin: A Peer-to-Peer Electronic Cash System. Cryptography Mailing list at https://metzdowd.com.

[KJL18] Ben Kaiser, Mireya Jurado, Alex Ledger. (2018). The Looming Threat of China: An Analysis of Chinese Influence on Bitcoin. arXiv:1810.02466v1 [cs.CR]

[KN12] Sunny King, Scott Nadal. (2012). PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. https://peercoin.net/whitepapers/peercoin-paper.pdf

[Poe15] Andrew Poelstra. (2015). On Stake and Consensus.
https://download.wpsoftware.net/bitcoin/pos.pdf

[NXT19] NXT Contributors. Version from 2018-07-02 15:03.
https://nxtwiki.org/wiki/Whitepaper:Nxt#Nxt.E2.80.99s_Proof_of_Stake_Model

[VB14] Vitalik Buterin. (2014-11-25). Proof of Stake: How I Learned to Love Weak Subjectivity.
https://blog.ethereum.org/2014/11/25/proof-stake-learned-love-weak-subjectivity/

[DLS88] Cythia Dwork, Nancy Lynch, Larry Stockmeyer. (1988). Consensus in the Presence of Partial Synchrony. https://groups.csail.mit.edu/tds/papers/Lynch/jacm88.pdf

[SCP16] David Maziere. (2016). The Stellar Consensus Protocol: A Federated Model for Internet-level Consensus. https://www.stellar.org/papers/stellar-consensus-protocol.pdf

[GTB19] https://arxiv.org/pdf/1902.10865.pdf