

# A Study of Blockchain Consensus Mechanisms with Emphasis on Proof-of-Reputation

Yidi XING<sup>1\*</sup>

, Omar HASAN<sup>1</sup>

, Sonia BEN MOKHTAR<sup>1</sup>

, Tarek AWWAD<sup>1</sup>

, Lionel BRUNIE<sup>1</sup>

and Harald KOSCH<sup>2</sup>

Correspondence:

idi.xing@insa-lyon.fr

LIRIS Laboratory, National

Institute of Applied Sciences of

lyon, 20 avenue Albert Einstein,

69100 Villeurbanne, FR

Full list of author information is

available at the end of the article

## Abstract

- The emergence of blockchain technology enables people to build a distributed, decentralized and tamper-proof account book through a trust free P2P network. This technology has broad application prospects in the fields of digital assets, remittances, online payment and other financial services. Systems based on blockchain technologies combined the application of P2P network, public key cryptography, hash pointer and cryptographic hash function to ensure the decentralization, persistence, tamper resistance, forgery resistance and auditability of the system.
- Users, as distrustful parties, can agree on the existence, value and transaction history of each other's accounts by maintaining consistency on the global blockchain network. This feature of blockchain network makes it possible to greatly save transaction costs, especially financial transaction costs, and improve transaction processing efficiency. It also allows financial services without the support of any banks or intermediaries.
- In the area of blockchains, consensus algorithms are the key elements in each blockchain P2P network, because they are responsible for maintaining the integrity and security of these distributed systems and ensuring that the system can operate on a trust-free basis. Consensus algorithms can be defined as a mechanism to achieve agreement in blockchain networks. Blockchain systems have decentralized attributes and are constructed as distributed systems. Since they do not rely on a central authority, decentralized nodes need to agree on the validity of transactions, which is the function of consensus algorithms. Consensus algorithm ensures that all nodes comply with the rules defined by the system designer and that all transactions are conducted in a reliable manner. For example, in the field of cryptocurrency, each token coin used for trading can only be spent once.

1

2

## Abstract

• While trying to balance security with functionality and scalability, each consensus protocol shows its own advantages and disadvantages. In this paper, we will focus on the analysis and comparison of different types of consensus protocols. In the second section, we first present the general design model of the hierarchical block chain system we envisage. We will further reveal the importance of the consensus layer by showing its importance, utility and potential interaction with other layers. Then in sections III and IV, we analyze and compare fourteen different consensus protocols. In the fifth, sixth and seventh sections, we will focus on an innovative concept of consensus protocols: proof-of-reputation protocols (PoR). PoR introduces the concept of reputation into the consensus process. We first introduce the general design model of PoR. Then we enumerate five existing por projects, compare and analyze their ideas, advantages and disadvantages, and try to provide possible trends for the future development of proof-of-reputation protocols.

**Keywords:** blockchain; consensus protocol; proof-of-reputation; decentralization

## Declaration

### Availability of data and materials

The blockchain systems data that support the findings of this study are available from “bitcointalk.org”, “www.coingecko.com/fr/pièces/”, “www.feixiaohao.com”, “coincheckup.com”, “blocktivity.info”, “bitinfocharts.com”, “www.reedit.com/r/Vechain/comments/97zmoy”.

Also, the next reported blockchain systems data were used to support this study and are available at “Practical Byzantine fault tolerance”, “Bitcoin: A peer-to-peer electronic cash system”, “https://blackcoin. co/blackcoin-pos-protocol-v2-whitepaper. pdf”, “DBFT: Efficient byzantine consensus with a weak coordinator and its application to consortium blockchains”, “The ripple protocol consensus algorithm”, “On security analysis of proof-of-elapsed-time (poet)”, “Slimcoin: A peer-to-peer crypto-currency with proof-of-burn”, “Proofs of space”, “Delegated proof-of-stake (dpos)”, “Komodo: An Advanced Blockchain Technology, Focused on Freedom”, “Komodo: An Advanced Blockchain Technology, Focused

on Freedom”, “Solana: A new architecture for a high performance blockchain v0.8.13”, “Pbft vs proof-of-authority: applying the cap theorem to permissioned blockchain”, “Algorand: Scaling byzantine agreements for cryptocurrencies”, “gochain.io/assets/gochain-whitepaper-v2.1.2.pdf”, “Blockchain: The State of the Art and Future Trends”. These prior studies (and datasets) are cited at relevant places within the text as references [8-11, 13-23].

#### Competing interests statement

The authors declare that they have no competing financial interests.

#### Fundings

//TO DO

#### Authors' contributions

Y has drafted the work. Y was the major contributor in writing the manuscript and also substantively revised it. O and SB have made substantial contributions to the conception and the design of the work. O and SB have also substantively revised the manuscript. L and H have drafted the work, and have made important contributions to the conception of the work. T have made important contributions on the substantive amendments. All authors read and approved the final manuscript thus the submitted version.

## 1 Introduction

Blockchain technology was first implemented by Nakamoto with Bitcoin applications in 2009[9]. It combines the application of encrypted hash functions, digital signature, Merkle tree, consensus protocol and peer-to-peer (P2P) network, so as to build a distributed and decentralized system based on trust-free P2P network. It could be used not only for financial trading systems[1],[2], but also Scientific research, resource management[3],[4], political domain[6],[7], etc. Using blockchain technologies, we can build a distributed database system based on distributed P2P network. The system could record a public account book, or called a “public ledger” – this ledger sorts groups of transactions in chronological order and uses encrypted hash function such as SHA256 to encryptedly link each group of transactions. Those sets of transactions in the record are stored in a specific data structure, which we

50 call a data block. As new transactions continue to be completed(), they are pack-  
51 aged into data blocks, which are submitted to the end of the list of data blocks on  
52 the public ledger. That is also why we call this technology blockchain.

53 The information contained in the ledger shows transaction history up to the cur-  
54 rent time through block chains. These transaction records prove the existence and  
55 value of each account. Therefore, in a decentralized block chain system, every up-  
56 date of the ledger must be authenticated by each account holder in the network. Of  
57 course, this means that there is a need for consensus among participants. In the real  
58 world, we may not be able to find application examples with the same limitations.  
59 For example, when an entity (bank or country) decides to issue legitimate digital  
60 currency, it does not need to establish a public ledger that must be confirmed in  
61 real time by each currency holder, because the entity, as the central agency, is re-  
62 sponsible for the verification needed to use such digital currency for transactions  
63 and ensures the security of transactions. In blockchain networks, this is not the  
64 case: nodes operate independently. In order to reach consensus, it is essential and  
65 necessary for nodes to communicate with each other through the network.

66 It can be imagined that in such a distributed system, there will be many kinds of  
67 errors in the process of sending messages between nodes. We can generally divide  
68 them into two types: the first is the error including node crash, data packet loss and  
69 network failure. The characteristics of these errors are that the nodes themselves  
70 are not malicious to the system. We call them “non-Byzantine errors”[?????]. The  
71 second type of errors refers to the arbitrary actions of the nodes and deliberate  
72 violations of the rules of action formulated by the system designers. At this point,  
73 the wrong node may itself be malicious. The behaviors include sending messages  
74 with different contents at the same time to different nodes, delaying or rejecting  
75 messages in networks, deliberate attempts to submit illegal transaction records,  
76 and so on. Such errors are called “Byzantine errors”[?????]. In serious cases, there  
77 may be collaboration between malicious nodes, making Byzantine errors a serious  
78 problem.

79 The consensus protocol is designed to build a distributed blockchain system into  
80 a Byzantine fault-tolerant system. In the face of two mentioned types of errors,  
81 the design of a qualified consensus protocol can keep the consistency and the live-  
82 ness of system. Consistency means that honest and harmless system participants

83 agree on records in the public ledger. The liveness represents that the ledger can  
84 be updated continuously, efficiently and effectively. There are a lot of practices of  
85 consensus protocols: Bitcoin which made successes on marketing, uses the Proof-of-  
86 Work protocol where users profit from computing proofs. They randomly find the  
87 node determining the next block[9]; or PoS protocol[10], which is used by Peercoin,  
88 where users profit there locked stake within the blockchain system prove that they  
89 are trustworthy, and to compete to win the right of generating subsequent blocks;  
90 or as PBFT protocols, all nodes identity should be known under this configuration.  
91 All nodes have equivalent voting rights, and they consumes numerous rounds of  
92 communications to reach consensus[8]. In this paper, we will focus on consensus  
93 protocols. First, we will give a general blockchain model which is widely used in  
94 practice. Next, we will introduce fourteen different consensus protocols that have  
95 been applied in practical projects, and analyze and compare them. Finally, we will  
96 mention a new and noteworthy consensus protocol concept, proof-of-reputation. We  
97 will focus on its introduction and analysis, and explain its unique advantages.

98 The rest of this paper is organized as follows. Section II introduces the general  
99 design model for blockchain system. Section III shows the state-of-art of fourteen  
100 different consensus protocols. Section IV summarizes the precedent ones by giv-  
101 ing tables and explanations showing the analysis results of those protocols, with  
102 a detailed explanation for these table and figures. Section V introduces the idea  
103 of proof-of-reputation, explains its idea, its operation principles, its general model,  
104 advantages and disadvantages. Section VI is an another state-of-art section where  
105 we list and present five different existing por blockchain projects. Section VII con-  
106 cludes.

## 107 II Background

108 In this section, we will introduce a general, layered and modular blockchain sys-  
109 tem model. It can be regarded as a template for blockchain projects that are now  
110 in operation. We will explain its composition, analyze which functional units the  
111 system consists of, which functions and operations the system supports, and which  
112 technologies the system uses to achieve them. The model in this section is inspired  
113 by the work of Yuan et al.[23]. Some changes have been made in the specific content,  
114 then in the layers and modules division. This basic model will consist of five layers:

the data layer, the network layer, the consensus layer, the incentive schemes and the application layer.

The data layer defines the representation of data in the blockchain system. It encapsulates the underlying data blocks, basic data, data encryption, timestamp and the related algorithms. The network layer determines the mode of data transmission. It includes distributed networking mechanism, data propagation mechanism and data validation mechanism. Consensus layer focuses on reaching consensus of data verification at the system level. It mainly encapsulates network nodes and specific consensus algorithms. Under our assumption, the three levels of network, consensus and incentive are particularly related to the implementation of consensus protocol.

The incentive schemes exist to ensure the honesty and legitimacy of users (network nodes), because data generation, data propagation and data validation depend on users' behavior and operations. It integrates economic factors into blockchain technology system. Incentive mechanism mainly includes the issuing mechanism and distribution mechanism of economic incentives. Finally, the application layer encapsulates various application scenarios and cases of blockchain.

## II.a Data layer

The data layer represents distributed accounts. Its content is shared by all nodes in the distributed block system. It encapsulates the underlying data blocks, related data structures, data encryption and timestamp algorithms.

Through the presence of data layer, every distributed node can use a specific hash algorithm (determined within this layer) and the Merkle tree data structure, to encapsulate the transactional data received in a certain time period into a data block and with time stamping on it. Nodes can add it to the end of local main blockchain and broadcast their local main chain to try to get agreement with nodes in the network.

In order to achieve the functions described above, the data layer mainly relies on six technologies: the data block, the hash pointers, the cryptographic hash function, the Merkle tree, the timestamps and the asymmetric cryptography.

**Data block** Also called as "transaction block" because it stores mostly transactions' information. Each data block contains a Header part and a Body part.

147 The block header encapsulates current block index, the address of the previous  
148 block, the hash value of current block, the Merkle-root of current block and  
149 its timestamp.

150 The block body contains the amount of transactions stored in current block,  
151 then the records of all validated transactions encapsulated during the genera-  
152 tion of this block. Those transaction records together generate the Merkleroor  
153 (through the hashing process of a Merkle tree) saved in the block header.

154 **Hash pointers** The data structure which allows the node to link the latest block  
155 to the previous one, thus constructing the chain of data blocks.

156 Through this technology, all history of data appeared in the blockchain system  
157 is locatable and auditable.

158 Sometimes, a node may have two or even several valid latest blocks that it  
159 must make choice among them to adding one of them on their local main  
160 blockchain. This is called as “fork selection problem”. This problem needs to  
161 be solved by the consensus layer.

162 **Timestamps** The timestamp is encapsulated in the header part of a data block,  
163 during the creation time of the block. It signifies the write-in time of the  
164 corresponding block. The purpose is to enable the confirmation that blocks  
165 are arranged in chronological order within the blockchain.

166 The hash pointers and the timestamps, together they construct the proof of  
167 existence of every data block, thus make the blockchain becoming a tamper-  
168 resistant ledger.

169 **Cryptographic Hash function** The raw transactions data are not recorded in  
170 the blockchain, but their hash value. The choice of using cryptographic hash  
171 function gives six properties to the records data:

- 172 1) As input, the raw data can be any string of any size.
  - 173 2) The output is a fixed size.
  - 174 3) The process to transform raw data to hash value is efficiently computable.
- 175 Intuitively it means that for a given input string, we can figure out the  
176 output of the hash function in a reasonable amount of time. More tech-  
177 nically, computing the hash of an  $n$ -bit string should have a running time  
178 that is  $O(n)$ .



4) Collision-resistant: even if the input differs only by one byte, it will produce significantly different output values. So it is infeasible to find same output value with different input.

5) Hiding: there's no feasible way to reverse the input value through the hash output.

6) Puzzle friendliness: if someone wants to target the hash function to come out to some particular output value  $y$ , but part of the input is decided in a suitably randomized way, it's very difficult to find an input value that hits exactly the output target.

The use of cryptographic hash functions guarantee the “tamper-resistant”, “efficiently computable during the creation” and “auditable” properties of blockchain records. The function that is most generally used is SHA256.

**Merkle Tree** The Merkle tree's function is to allow to the efficient summarization and validation for the existence and integrity of block data.

**Asymmetric Cryptography** Asymmetric encryption usually uses two asymmetric ciphers in the encryption and decryption process, called public and private keys. This key pair has two characteristics: The first is when one of the keys is used to encrypt the information, only the other key can decrypt the data. Secondly, the public key can be disclosed to others, and the private key is kept secret, and other people cannot calculate the corresponding private key through the public key.

The asymmetric encryption technology is applied in the scenarios of the blockchain's information encryption, digital signature, and login authentication. The information encryption scenario that the sender of the information (denoted as A) uses the public key of receiver (denoted as B) to encrypt the information then send encrypted data to B. B decrypts the information by using its own private key.

The digital signature scenario is that sender A sent messages with his/her own private key to B, B uses the public key of A to decrypt. In this way, B can be ensured that the messages are made by A.

As for the login authentication scenario, the client encrypts the login information with the private key and sends it to the server. The latter takes client's public key to decrypt and authenticate the login information.

## 212 II.b Network layer

213 The network layer encapsulates the network building mode, the messaging proto-  
214 col, the data verification mechanism, etc.

215 Those mentioned modules of network layer should be defined corresponding to  
216 the need of real applications based on. Through this layer, every node within the  
217 blockchain system can participate to the maintenance (verification of data) and the  
218 updating of data blocks.

219 The function of network layer is basic for a blockchain system since the system is  
220 distributed. We also need that all the nodes could synchronize with each other on  
221 the updating of distributed ledger. This challenge can be resolved by the cooperation  
222 between consensus layer and network layer.

### 223 • Network Building Mode

224 Existing blockchain systems generally take Peer-to-Peer Network(p2p net-  
225 work) as their networking mode. Nodes within the network are the users who  
226 have the right to participate to do the data verification and ledger's updating.  
227 Within a p2p network, all nodes possess the same standing. They connect  
228 and communicate with each other based on a flat topology. There are no  
229 special centralized nodes, neither hierarchical structures. Each node will in-  
230 dependantly take on the network routing, block data verification, block data  
231 propagation and new nodes' discovering tasks.

232 For a blockchain network, nodes are often divided into "full nodes" and  
233 "lightweight nodes". The former stores the total records from the genesis  
234 block(first instantiated block at the creation of the blockchain system) until  
235 the latest one, participates on real-time to the data verification and ledger  
236 updating. As for the "lightweight nodes", they record only partially the  
237 blockchain, and generally request their required data from connected nodes  
238 to accomplish their operation such as data verification. A general reason that  
239 not every user could support a full node is the high space cost of it, as for  
240 Bitcoin, after 2016, a full node needs to store in local a data set more than  
241 60GB[23]; Different existing blockchain projects offer their own strategy for  
242 their "lightweight nodes", again as for Bitcoin, they also have designed a  
243 Simplified Payment Verification method to support[????].

For a blockchain network, the entire network data is stored on all nodes of the decentralized system. Even if some nodes fail, as long as there is still a functioning node, the blockchain main chain data can be completely recovered without affecting the recording and update for subsequent block data. This decentralization-based concept brings a better data security compare to other centralized or multi-centralized data storage mode such as Cloud.

#### ● Messaging Protocol

Since the network is distributed, once upon the generation of a data block, the generator node needs to broadcast its result to other nodes on the global network in order to get their verification for this block. For a blockchain system, the messaging protocol generally include five steps as shown below:

- 1) Nodes involved by transactions broadcast their transaction data to the nodes on the global network.
- 2) Every full node collect their received transactions then package them into a data block.
- 3) Through the consensus protocol adopted by current system, some of the full nodes will get the right to sign and publish their block packaged - they broadcast the block to the nodes on the global network.
- 4) Data verification: other nodes only validate the block when all transactions within are legitimate and not stored in the ledger yet.
- 5) Block acceptance: once the data verification has done, nodes could accept this received block and add it in the ledger(on the end of their local blockchain).

#### ● Data verification mechanism

This mechanism mainly handles two operations: verification for transaction data, and verification for data blocks.

For the transactions' data received from connected nodes, their validity are firstly be verified. If they are valid data, they will be put into a local transaction pool by chronological order, and be broadcasted at the same time to the subsequent connected nodes; if they are illegitimate transactions, these data will be rejected thus banned from the blockchain network.

The validity of transaction data concerns mostly their data structure, their grammatical normative, their data signature, etc.

277 As for the data blocks, their validity is also firstly verified. If they are vali-  
278 date, they will be locally accepted into a main chain by current node, and be  
279 broadcasted to the subsequent connected nodes; if not, they will be rejected  
280 and thus banned from the network.  
281 The validity of data blocks concerns their hash value, their timestamp, their  
282 content transactions' validity, etc.

## 283 II.c Consensus protocol

284 How to achieve consensus efficiently in distributed systems is an important re-  
285 search issue in distributed computing field, the utility and the importance of consen-  
286 sus layer is to - in a decentralized system with highly decentralized decision-making  
287 power - make each node highly efficiently achieve agreement on block data validity.

288 Existing consensus protocols are various, some of the representative ones are  
289 PoW(Proof-of-Work) and its variants such like PoS(Proof-of-Stakes), dPoS(delegated-  
290 Proof-of-Stakes); PBFT(practical-byzantine-fault-tolerance) and its variants such as  
291 FBA(federate-byzantine-agreement), dBFT(delegated-byzantine-fault-tolerance).

292 The general idea of existing consensus protocol is to - for each round of the system  
293 - as much as possible randomly elect a leader(or multiple leaders), so that all nodes  
294 could have consensus on the updated content of the ledger after locally completing  
295 data verification, and every node has equivalent opportunities to become a leader  
296 node. For that purpose, the general design of existing consensus protocols is that  
297 nodes must show a proof supported by a certain scarce resource(such as hash com-  
298 puting power with PoW, cryptocurrencie tokens with PoS and dPoS, nodes' votes  
299 with dBFT[11], dPOS[17] and FBA, etc) in order to win the right of ledger updat-  
300 ing. The scarcity of such resource guarantees the fairness of this "leader election"  
301 process, and could be considered as a "security deposit" that winner nodes will  
302 honestly and legitimately operate - if they act maliciously then they will lose their  
303 invested resource.

304 The existing consensus mechanisms have their own advantages and disadvan-  
305 tages. The PoW-like consensus mechanism has formed a mature cryptocurrency-  
306 mining industry based on its first-mover advantage, for example, Bitcoin and Lite-  
307 coin projects; while emerging mechanisms such as dPoS, FBA have their relative  
308 advantages on safety, environment friendly and/or efficiency[?????]. The choice of

consensus protocols has become the most difficult problem to reach a consensus for  
blockchain system researchers[?????].

### *II.c.1 Main challenges faced by the consensus protocols nowadays*

#### ● Pormance bottle neck:

Taking Bitcoin and Ethereum – the most successful blockchain projects – as  
examples: in Bitcoin, the system could process 7 transactions per second in  
average, and with Ethereum, this number is currently 20, which is much lower  
than centralized online payment system such like Paypal and Visa, which –  
in practice - process separately 115 and 2000 transactions per second[9],[23].  
Most of the recent consensus protocols aim on the improvement on perfor-  
mance with, however, a trade off between the performance and the scalability,  
the security and/or the decentralization.

#### ● Energy overhead issue:

As of today, 3.5 million US households could be powered with the energy  
used to run the Bitcoin network, while Ethereum uses the equivalent power  
of 1 million households. This is an unsustainable overhead. To resolve this  
problem, there exists 3 convenient ways which are “decreasing the exigency  
on local computing ability for the individual node”, “reducing the complexity  
of data/messages transmitted on the network”, “reducing the complexity of  
number of rounds needed to reach the consensus” - numerous recent protocols  
proposed different solution concepts.

#### ● Scalability problem:

As for a blockchain system, the scalability represents principally the openness,  
and the admissible network size of the system. It’s considerable that a lot of  
recent protocols – in order to improve the system performance – sacrificed the  
scalability, making their system became closed, or the acceptable number of  
nodes being limited.

#### ● Security problem:

The security notion signifies principally the reliability of results of the proto-  
col, the security of transaction operation lanced by every individual node, and  
the confidentiality of data for every individual node. The classical consensus  
algorithm of Bitcoin provides – well proved in practice – a very nice security.

341 Although for some new protocols which direct the performance and the en-  
342 ergy efficiency improvement, a strict proof on their security is lacking. Some  
343 of them even have a hard-to-solve security hole, thus can not be operated  
344 independently.

345 In fact, even for the Bitcoin algorithm, the recent research on “selfish mining  
346 strategy/attack” also pointed that, the Bitcoin’s security mechanism could  
347 only tolerate half of the malicious nodes compare to its intended design.

348 ● Centralization issue:

349 As for 2017, 80% of all blocks generated in Bitcoin network are mined by large  
350 mining companies in Iceland and in China[23], the system’s decentralization  
351 has been gradually lost. The degree of decentralization of system rights is one  
352 of the most significant difference among the various protocols. In addition,  
353 some of recent protocols made concessions on the decentralization degree for  
354 the system’s performance and reliability[????].

355 **II.d Incentive schemes**

356 The nature of the consensus layer is to outsource the ledger updating and mainte-  
357 nance tasks to all nodes. Since every rational node is self-interested, the purpose of  
358 having incentive schemes is make the individual rational behavior that maximizes  
359 the benefits of each node being consistent with the overall goal of the security and  
360 effectiveness during the consensus process of the decentralized system.

361 ● Issuing mechanism

362 Currently, the issuing of incentive tokens is mostly based on the augmentation  
363 of new data blocks and new transactions, the reason of this situation is that the  
364 practical effect of incentive mechanism is to make the action of using system  
365 services by nodes being always profitable for the users. Taking the Bitcoin  
366 as example, each block since the genesis block will issue 50 bitcoins to the  
367 bookholders of the block, after which the number of bitcoins issued per block  
368 will be reduced by half every 4 years (namely 210,000 blocks in average). The  
369 number of Bitcoins will stabilize at the upper limit of 21 million. The bitcoin  
370 transaction process will also incur a fee, the current default fee is one ten  
371 thousandth of a bitcoin.

372 ● Distribution mechanism

373 The general distribution approach of incentive tokens could be divided into  
374 two parts: one part is for the ledger updaters nodes, they have contribution  
375 for the maintenance and updating of the distributed ledger, so they should be  
376 rewarded because of their contribution; the another part is for the transaction  
377 proposer nodes within the system, their action brings liveness to the system,  
378 increases system network traffic and creates needs of system service.

## 379 II.e Application layer

380 The blockchain system has the characteristics of distributed high-redundancy stor-  
381 age, time-series data, tamper-resistant and forge-resistant, decentralized credit, in-  
382 telligent execution of smart contracts, security and privacy protection, which makes  
383 blockchain technology not only could be successful in the field of digital cryptocur-  
384 rency, there are also a wide range of applications in economic, financial and social  
385 systems.

## 386 III Related Work – Consensus algorithms

387 In order to let the reader get a better understanding about the evolution and the  
388 state of the art of the blockchain consensus protocols, we list and explain sixteen  
389 different protocols below. The content of the explanation includes a general intro-  
390 duction, their mechanism, and an analysis about their strengths and weaknesses.

### 391 III.a Proof-of-Work(PoW)

392 **Overview** PoW is the first consensus protocol applied to the blockchain system.  
393 As a landmark model of consensus protocols, it mainly answered to four questions  
394 below:

- 395 1. Who packages transaction blocks and then updates the ledger(maintain the sys-  
396 tem operation)?
- 397 2. Why users would have the motivation to take care of the update of the ledger?
- 398 3. How the rewards of maintaining the system operation are published and dis-  
399 tributed?
- 400 4. How do we locally determine our main chain while fork selection problem occurs?

401 **Consensus process** The detailed mechanism of PoW contains four phases:

402 1. In order to commit the transactions(such as, online payment, data/file trans-  
403 mission, etc) to the ledger, the nodes need to broadcast their own transactions in  
404 the p2p network.

405 2. The nodes that are willing to participate in the update of the ledger are called  
406 as “miners”, they firstly verify the received transactions. They store the valid ones  
407 locally, thus forming a pre-committed transactions pool.

408 3. For each round(in Bitcoin, one round is 10 minutes, and in Ethereum, it is  
409 15 seconds), miners need to compete, trying to – in the fastest way – resolve a  
410 mathematical problem called “hash puzzle”. Only the miners who have found a  
411 solution are able to package their transactions in the pool into a block, and then  
412 sign, publish, and broadcast this block to the entire p2p network.

413 When a block is accepted into the main chain, then the signer can get rewards  
414 for it - which can be an amount of cryptocurrencies, or in form of other tokens.

415 4. The solution found by the block signer is put into the block’s header. All nodes  
416 can verify the validity of this received block by cheking this “hash puzzle” solution.  
417 This verification is mathematically simple and efficient, so the common nodes can  
418 easily check if this signer published a valid block.

419 The earlier a miner publishes its block, the higher the probability it will gain for  
420 winning the current round’s competition. Because of this fact, whenever a miner  
421 receives blocks signed by the other miners, it will have the motivation to verify it,  
422 accept it then turn to find solutions for next round - in order to have more chance  
423 to be the winner for the next round.

424 At the same time, nodes have also the tendency to accept a new block preceded  
425 by a longer chain, because that means more computing power are invested on this  
426 fork, miners will have a higher probability to gain benefits from mining on this fork,  
427 and normal users can have more security by accepting this blockchain(this ledger).  
428 Through the incentive mechanism which allows the mining being a profitable thing,  
429 the PoW protocol guaranteed that the selection of forks by the miners is converge.  
430 As for the common users, they take the same decision as the honest and rational  
431 miners in order to more securely user the services provided by the system. In this  
432 way, a global consensus of the network on the main chain can be achieved.

433 **Strengths of PoW:**



434     • **Reliable:** It has been widely tested Since 2009 and still generally used nowadays.  
435 Its reliability and safety have been proved in practical operation.

#### 436     **Weaknesses of PoW:**

437     • **Not friendly to the environment:** In the consensus process, “solving hash puzzle”  
438 step is very demanding in terms of computing resources and electricity power. This  
439 overhead is considered as a waste of resources.

440     • **Plutocracy:** The amount of actual investment directly affects the computing  
441 capacity of nodes. It means that the ”economies of scale” can easily disrupt the  
442 decentralization feature and security mechanism of the system.

### 443     **III.b Proof-of-Stake(PoS)**

444     **Overview** Proof-of-Stake is a variant of PoW[10]. Its conception is to replace the  
445 notion of “work(or, computing power as in PoW)” by the notion of “interests(or  
446 assets, stakes)”. Stake refers to the capital held by system participants in the system,  
447 and stakes are themselves a proof of scarce resources. When participants use their  
448 own capital as a deposit, they can already be trusted to act honestly.

449     On the other hand, this design allows us to skip the “hash puzzle resolving” step  
450 as in PoW, that means a significant drop in energy overhead.

451     **Consensus process** PoS retains the four-step consensus processes of PoW:  
452 “Propagation of transactions”, “Collection and verification of transactions”, “Min-  
453 ing Competition and publishing new blocks”, “Verification of new blocks”. PoS  
454 adjusts this process in the second, third and fourth step:

455     *Changes in step 2 and 3:* Nodes do not need to invest computing power to re-  
456 solve the hash puzzle problem. Instead of that, they need to show a portion of or  
457 entirely their assets held in their system account. These locked assets are called as  
458 “stakes”, and users who hold stakes become “validators”. At each round, validators  
459 can randomly be chose to become new block generators based on the percentage of  
460 the total stakes and stakes they own. This percentage could not only influenced by  
461 the amount of capital of the stakes, but also by more factors, such as the “coin age”  
462 - the length of time the user holds these stakes - of the tokens of stakes. When a  
463 validator is selected as the block producer for the current round, it begins to collect  
464 transactions and package them into blocks.

465 We can note that PoS changes the rules for deciding who becomes the block  
466 generator, and changes the order of the second and third steps: in PoW, each miner  
467 has the right to package his own blocks; in PoS, it is not necessary for miners to  
468 package blocks ahead of time before they are selected.

469 *Changes in step 4:* In PoW, each user should independently verify the blocks it  
470 receives and broadcast to the network the blockchain it chooses. In PoS, only the  
471 validators need to verify the block independently, sign the block and broadcast its  
472 own choice. The blocks with enough signatures are regarded as final blocks, and  
473 common users simply need to accept the final blocks received first.

#### 474 **Strengths of PoS:**

475 • People with a vested interest: Attacking a PoS system during the consensus  
476 process is very harmful for the attackers, because that would directly damage the  
477 assets in its accounts - in order to participate in the consensus, these assets are  
478 equivalent to deposits.

479 • Finality: Compared with PoW, the consensus process of PoS is more decisive  
480 for the new blocks in each round. For the common users of the system, this means  
481 that they can experience less waiting time to determine their trading results and  
482 be more confident about the results.

483 • Resistant to the “scale economy”: in PoW, for ten thousands miners that each  
484 pays one euro electricity fee per minute, they hold actually a pretty low computing  
485 power, although for one miner who pays ten thousands euros electricity fee per  
486 minute, it gets a very high computing power[????]. While in PoS, we can guarantee  
487 that the interest brought by one euro is constant.

**Weaknesses of PoS:** • “Nothing-at-the-stake problem: seeing the fact that mining is almost free for every participant  
488 in a PoS system and the more bifurcations, the more number of block generators  
489 in each round, the rational users will have the tendency to generate as many as  
490 possible forks, and generate blocks on all these forks, in order to gain a maximal  
491 benefit. This behavior can lead to a system inflation, then a serious depreciation of  
492 system assets. Many PoS systems have designed rules to limit the impact of this  
493 problem with specific algorithms.

495 • First-mover advantage: The sooner users enter the system and the more re-  
496 sources they invest into the system, the more benefits they can naturally expect

497 from the system and the higher weight they gain to influence the consensus pro-  
498 cess.

- 499 • Stochastic algorithm: In PoS, the producer of block is determined randomly by  
500 the system in each round. The reliability of stochastic algorithm is very important.

### 501 **III.c delayed-Proof-of-Work(dPoW)**

#### 502 **Overview**

503 The dPoW is first run by Komodo platform[18]. The idea of this platform is to  
504 build a lightweight but equally secure blockchain system using the high computing  
505 power of some existing PoW-based platforms or the security (a large amount of cap-  
506 ital) of some PoS-based platforms: by backing up the ledger of the dPoW blockchain  
507 into the ledger of high-computing and high-security platforms. At a lower cost, the  
508 dPoW provides a safe implementation plan for small and medium-sized blockchain  
509 projects.

510 **Consensus process** Here we take the Komodo as example - the first cryptocur-  
511 rency where the dPoW is implemented:

512 By select a group of nodes called “notaries” in the network of the original system,  
513 the new one transmits firstly all its pre-committed transactions to these notaries; the  
514 selected nodes submit those transactions to the safe and existing PoW blockchain,  
515 then return the results of transactions processing back to the new system - here  
516 comes the notion “delay” in the title of this protocol.

#### 517 *Strengths of dPoW:*

- 518 • The dPoW system does not have any necessity on hash computing power, thus  
519 is it environment friendly.
- 520 • Even without the “hash puzzle resolving” step, the system can also have a good  
521 security guaranteed.
- 522 • dPoW can give additional value to other system, without need of directly offering  
523 cryptocurrencies, neither making any tradings among them

#### 524 *Weaknesses of dPoW:*

- 525 • The system must rely on a PoW/PoS system.

- With the existing of notaries, the original system must arrange different hash rates for common nodes and notaries nodes, otherwise, the relied system can not actually operate, or the original system's security will be weakened.

### III.d PoET(Proof-of-Elapsed-Time)

#### *Definition*

The PoET protocol was introduced by Intel research team[14], it's also a variant of PoW. Its idea is to replace the notion of "work(or computing power)" by the notion of "time cost".

#### *Consensus process*

The process of PoET is also basically the same to PoW, only differs at the block generation method: in PoET, in order to generate new blocks and get rewards, nodes need to firstly sleep for a randomly generate length of time. Once it's awoken, it can send the awoken time to a pre-committed block for current round. Among all the nodes competing for a same block, the first of them to wake up wins.

#### *Strengths of PoET:*

- The PoET system gives an equal chance of winning to a large number of network participants, low resource users are also worthy to join the competition.
- For all the participants, it's very easy to verify that the block generator was delegated in a legal way.
- The cost that every node needs to pay for being delegated, is proportional to the benefit obtained from it.

#### *Weaknesses of PoET:*

- Hardware dependencies & Single point of failure: The PoET mechanism has 2 critical exigencies: the waiting(sleeping) time of each node is randomly choosed, and the winner participant has really accomplished the wating. This internal mechanism demands that this part of trusted codes need to be operated in a trusted environment, as for PoET, it relies on some specific Intel hardwares. It also can cause a single point of failure issue, whenever someone hack the Intel hardware, the corresponding node can generate as much blocks as it wants.

### 555 III.e dPoS(delegated-Proof-of-Stake)

#### 556 *Definition*

557 dPoS is a variant of the PoS protocol. With dPoS, it's still important for the nodes  
 558 to hold an amount of equity within the system, but they no more need to partially  
 559 block their assets as tokens, and they do not compete to gain a "stake holder"  
 560 identity[17]: different from PoS, the nodes do not compete to win the right of block  
 561 generation, their right is to elect leaders(called as "witness"). The witnesses form  
 562 a committee, then take charge of the generation of blocks in a cooperative way. In  
 563 dPoS, the system actually centralized the block generation step.

#### 564 *Consensus process*

565 Here's a concrete process of dPoS protocol:

- 566 1. During each period of "ledger maintaining", nodes can vote for other nodes as  
 567 "witnesses of current period". Most of the dPoS systems use "affirmative votes"  
 568 mechanism, which means they can only vote in favor, thus the nodes who get the  
 569 highest accumulated weight can be elected: the weight of votes of every node de-  
 570 pends directly on their holding stakes, more specifically, it depends on the propor-  
 571 tion of their holding stakes to the total stake of the system.
- 572 2. Once the election completed - some of the dPoS systems will also elect a list of  
 573 alternative witnesses, who will replace some of the actual witnesses if they acted  
 574 maliciously or if they can't work normally - a committee of witnesses is actually  
 575 established, the witnesses collect the pre-submitted transactions, then package them  
 576 into transaction blocks by a polling manner.

577 Without changing the solutions proposed in PoW of "why the nodes have the moti-  
 578 vation to maintain the ledger" and "the distribution of incentive tokens", the dPoS  
 579 made innovations on the solutions of "the generation of new blocks" and "the se-  
 580 lection of blockchain forks": the former is taken over by a delegated committee, the  
 581 latter's answer is that every on duty witness signs and publishes deterministically  
 582 their block.

#### 583 *Strengths of dPoS:*

- 584 • High energy efficiency compare to PoW and PoS. The existing of the elected  
 585 committee reduces the complexity of messages and rounds needed to reach the con-  
 586 sensus, the skip of "hash puzzle" step saves also a lot of computing power.

- High performance. The reduced messages and rounds complexity also improve the protocol performance.

#### *Weaknesses of dPoS:*

- The centralization in “blocks generation” step make the system being possibly controlled by a group of high equity nodes.
- As a supplement to the above point: in order to get the incentive tokens, high stake holder nodes will always have a tendency to vote for themselves - and they have high voting weight by themselves - which make the elect process also becoming centralized.

### III.f Algorand

#### *Definition*

The Algorand protocol was proposed by MIT’s research team in 2017[21]. It’s a protocol based on PoS, PBFT[8] and elect mechanism, the research team focused on the “random leader election problem”, or in other words, “the distribution of the right of blocks generation”. For that purpose, the Algorand protocol mainly answered to 3 questions: “how to build a randomness generator”, “how to guarantee that elected leaders can prove themselves without revealing their identity(avoiding leader-targeted attack)”, and finally, “how to deal with off-line nodes(appeared in the election process)”.

#### *Consensus process*

The concrete process of Algorand consists of 2 basic phases:

1. Proposer election. The proposers have the right to generate blocks in the current period. The election process is an imitation to PoS, the weight of being selected of a node depends on its holding equity.

2. Using BA\*(Byzantine Agreement\*) algorithm to reach the consensus.

The Algorand protocol uses a cryptographic sortition algorithm, such that every proposer learns in a secret situation that it was selected.

Each proposer firstly broadcasts the highest priority block that it considers, afterward broadcasts its known highest priority block, these 2 steps are achieving by using PBFT process.

617 The consensus is firstly made among the proposers, thus would be inserted in local  
 618 for all other normal nodes.

#### 619 *Strengths of Algorand:*

- 620 • It combines the using of PBFT algorithm and the idea of public blockchain:  
 621 the Algorand system is freely for nodes to join or leave, and benefits from the fault  
 622 tolerance feature of PBFT consensus protocol.

#### 623 *Weaknesses of Algorand:*

- 624 • Despite its complex process, there is no direct results showing that Algorand  
 625 has a better performance than other election mechanism based protocol such as  
 626 dPoS.

### 627 III.g PoC(Proof-of-Space)

628 PoSpace, also called as PoC(Proof-of-capacity), is a variant of PoW protocol,  
 629 instead of hash computing power, the tokens that nodes need to invest into the  
 630 competition is a certain amount of memory or disk space[16].

631 The concrete process of PoC is very similar to the PoW, only using a different and  
 632 special hash function called MHF(Memory Hard Function): the function feature is,  
 633 its computing cost depends on the memory size that this function can call.

634 The “hash puzzle” step in PoC can prove that the node - which have found  
 635 a solution - saved or say “invested” enough memory space for the competition.  
 636 The verification step should stay efficient, one possible solution is by asking the  
 637 competitors to generate Pebbling figures, and verifiers just simply needs to check  
 638 several random spaces in the figure.

#### 639 Advantages of PoC:

- 640 • It is more environment friendly compare to PoW, because the storage space is  
 641 a more generic resource than the hash computing power, and occupy also lesser  
 642 energy.

#### 643 Defects of PoC:

- 644 • The capacity based competition can lead to an another centralization situation.
- 645 • The fact that hard disk space become valuable can encourage hackers to develop  
 646 malicious software, and attack people’s hard disk.

### 647 III.h PoBurn

648 The PoBurn protocol is a variant of PoW[15], instead of investing on hash comput-  
649 ing power, the miners need to send their cryptocurrencies(tokens) to a unretrievable  
650 address and thus “burn” their tokens, in order to win the right of mining new blocks.

651 Basically the same as PoW, the only change that PoBurn has made in its consensus  
652 process is that the protocol will randomly generate some addresses which do not  
653 have a private key, thus the coins stored in there can not be spent, and the protocol  
654 also creates a book to track these coins.

655 Advantages of PoBurn:

- 656 • Users who tend to hold cryptocurrencies for long-term gains would have more  
657 chance to be benefited from a such system.

658 Defects of PoBurn:

- 659 • Still wasting resources insignificantly.
- 660 • Nodes that don’t care the waste of their coins would have more possibility to  
661 generate blocks, which means, the high resource nodes can still control the system  
662 service, just like in PoW now.
- 663 • The fact that “coins have been burnt” is not easy to be verified, this can either  
664 cause security issue, either lead to delay in transaction processing.

### 665 III.i PoA(Proof-of-Authority)

666 PoA protocol runs based on a pre-determined committee of nodes called sign-  
667 ers[20]; the signers take charge of blocks generation; signers can vote for invite new  
668 members; signers work in a polling manner, and each signer must wait for a fixed  
669 period to have the chance to generate a block again.

670 Here’s the concrete process of PoA Protocol:

- 671 1. A list of initiate signers are determined in the genesis block.
- 672 2. The signers take charge of the blocks generation in a polling manner, which  
673 means, the “IN-TURN” signer can publish its block with a higher priority, and  
674 the other “OFF-TURN” can also propose their own block - but with an inferior  
675 priority - in order to deal with the situation that the “IN-TURN” one was offline.
- 676 3. The signers can potentially make a proposal of “invite new signer join in the list”  
677 or “exile an original signer” by broadcast it as a transaction.



678 Advantages of PoA:

- 679 • The consensus has high energy efficiency compare to PoW.
- 680 • The consensus has high performance.

681 Defects of PoA:

- 682 • The system is actually centralized, or more specifically, “multi-center”, thus more  
683 adoptable for a system where all the nodes identity are verified before joining.

### 684 III.j PoHistory

685 PoH protocol aims on making transactions processing independent from the con-  
686 sensus process. This protocol is a variant based PoS algorithm[19].

687 With PoH, we form a “hash chain” by continuously running the hash function.  
688 This chain includes the number of times the function runs, the function state, the  
689 output value, and the block index. Each record on this hash chain is stored in-  
690 side a transaction block, which is equivalent to, coding a trusted clock into the  
691 blockchain—the research team’s assumption here is that the timestamps of trans-  
692 actions received by the system are not necessarily trusted.

693 The significance of PoH is that the nodes do not need to witness, neither to  
694 communicate with each other, every node can verify locally the time and sequence  
695 of event occurrences. Thus the PoH system does not demand to all the nodes to  
696 achieve a consensus, but only asks everyone to agree that event A occurred before  
697 event B.

698 The hash chain generated by PoH is a part of blockchain, as for the generation of  
699 blocks, the PoH protocol relies on PoS algorithm.

700 Advantages of PoH:

- 701 • High Performance, especially high throughput, because of reduction on message  
702 exchanging complexity.
- 703 • The consensus has high performance.

704 Defects of PoH:

- 705 • The PoH project in the real world is still in early days, lack of information.
- 706 • Experiments about the system’s reliability are not begun yet.

### 707 III.k BFT(Byzantine Fault Tolerance)

708 The BFT is the description of the reliability of a fault-tolerant computer system  
709 facing Byzantine failures: the Byzantine failure is a crash(or fail-stop) where the

failure nodes can have any arbitrary behaviors. While happening Byzantine failures, if the node behaviors include malicious responses and information forged, we call this situation as “Byzantine faults”, and these nodes as “Byzantine nodes”.

### III.l PBFT (Practical Byzantine Fault Tolerance)

PBFT is a state machine replication algorithm[8]. The service is modeled as the state machines, the state is replicated in different nodes of the distributed system. PBFT is adopted for closed system and demands communications among every pair of 2 nodes.

The concrete consensus process of PBFT is:

1. The client send requests to primary nodes.
2. The primary nodes broadcast the received requests to backup nodes.
3. The backup nodes verify the primary identity.
4. The backup nodes commit the received transaction/request.
5. The backup nodes reply to the primary one.

Advantages of PBFT:

- High Performance: high throughput and high bandwidth.
- High Security: It has a relative security since all members joining the network are being validated. However, this situation can be considered as “insecure” for small users who don’t belong to any of those center organizations.

Defects of PBFT:

- Only adopted for closed and non-large scale system.
- The system is centralized, or at least “multi-center”.

### III.m dBFT(delegated Byzantine Fault Tolerance)

With dBFT protocol, the global nodes select some agents nodes by voting; then those agents run the PBFT algorithm[8] between them to decisively complete the block generation mission. Voting in the network is real-time and asynchronous[11].

Advantages of dBFT:

- High Performance.
- High scalability for large scale system.

Defects of dBFT:

- The system is centralized, or at least “multi-center”.

### 741 III.n FBA(Federated Byzantine Agreement)

742 The main difference between FBA and PBFT is that, the nodes no more need to  
743 get consensus with other nodes on the entire network, but with “a certain quorum  
744 of nodes”, or with a “subnet representing a sufficient number of nodes”.

745 As for the concrete process, FBA works basically the same as PBFT, the only  
746 difference is that the system can have - at the same moment - a list of primary  
747 nodes, each primary node takes care of its own main chain, then in chronological order  
748 make consensus among them to get an agreement of the global view.

749 Advantages of FBA:

- 750 • Tremendeous throughput.
- 751 • Low transaction processing delay.
- 752 • Good system scalability.

753 Defects of FBA:

- 754 • It relies on the trustworthiness of the subnetwork chosen by each node.

### 755 III.o Ripple consensus

756 Ripple protocol is a variant of FBA protocol. It's nowadays an opensource online  
757 payment protocol[13].

758 In Ripple's network, the transactions are initiated by the clients (applications).  
759 Then the transactions are broadcasted to the entire network via the tracking nodes  
760 or the validating node.

761 Ripple's consensus is achieved between the validating nodes. Each validating node  
762 is pre-configured with a list of trusted nodes called UNL (Unique Node List). The  
763 nodes on the list should vote on the transaction deal. Once the approved votes reach  
764 a threshold, the current validating node will send these deals to other validating  
765 nodes: this transmission will continue, until the transaction reaches the fourth time  
766 the threshold - which is, 80% of approved vote. Afterward this deal/transaction can  
767 be recorded in the ledger.

768 Advantages of Ripple:

- 769 • High performance, low transaction processing delay.
- 770 • High Security: It has a relative security since all members joining the network are  
771 being validated. However, this situation can be considered as “insecure” for small  
772 users who don't belong to any of those center organizations.

773 Defects of Ripple:

- 774 • The fault tolerance percentage is only 20% for Ripple system.

### 775 III.p Stellar

776 **Overview** Also called as “Stellar Consensus Protocol” (SCP). This protocol is ac-  
777 cepted by Stellar Lumens, the intended object of its bussiness project is to construct  
778 a opensource and distributed payment infrasctructure.

779 The Stellar is also a variant of FBA protocol[12]. In fact, the design of Stellar  
780 originated from Ripple’s scheme. It can be considered as a branch of Ripple.

781 **Design Features** Unlike in Ripple, the Stellar system does not pre-set trusted  
782 nodes, or in other words, there is no UNL for the validating nodes[13]. In Stellar,  
783 the nodes themselves decide the subnet they trust.

784 Advantages of Stellar:

- 785 • High performance and good scalability.

786 Defects of Stellar:

- 787 • Configure a list of trustble nodes is costly for every user; and a bad configuration  
788 can cause forks or other Byzantine faults.

## IV Analysis of Consensus protocols

*Consensus algorithms comparison* Various consensus algorithms have different strengths and drawbacks. Table I to Table IV bring an assessment around various consensus algorithms, and we use the properties considering following [24], [26], [27], [28], [29], [30].

Protocols/E-sample	Blockchain Type /Node Identity	Perfomance	Energy Efficiency
PoW/Ethereum	public (public blockchain protocols are also suitable for consortium and private blockchain systems)/public	15tps(transactions per second)	no
PoS/Peercoin	public/public	97tps	partial - Hash computing(mining process) still exists
dPoW/Komodopublic	public/public	100tps, potential 45.000 tps	partial - Hash computing(mining process) still exists
dPoS/Bitshares	public/public	100.000tps claimed, daily proven 3400tps	partial - Hash computing(mining process) still exists
Algorand / Algorand	public/public	>1000tps claimed	partial
PoC/Burstcoin	public/public	80tps	partial-using hardware memory instead of hash computing power, however the energy-consuming mining process still exists

795 *Table I-1. Comparison of consensus protocols for blockchain type, performance and*  
796 *energy saving level.* 1) Blockchain type and Node identity: it's useful to understand  
797 if a protocol can serve for a public system, or only for a closed system. Nowadays,  
798 the blockchain systems generally include 3 concepts in terms of type division—  
799 a) the public chain, in which all member nodes can freely join and leave; in  
800 Ethereum, Bitcoin, Peercoin, Bitshares, their purpose for a decentralized network  
801 made them choosing public chain.  
802 b) the private chain, completely private, with strong third party providing node  
803 identity assurance and controlling node permissions distribution; these systems are  
804 often controlled by a single organization or company.  
805 c) the consortium chain, “partially guaranteed decentralization” – also called as  
806 “semi-private chain”. It is generally operated by specific organization groups that  
807 opens the inscription access to qualified users and ensures that the identity of the  
808 nodes is audited and documented. In practice, many financial and commercial in-  
809 stitutions are building their own ”circle of friends” based on block chain technology  
810 with consortium chain, especially like Lawtooth Lake Hyperledger, Hyperledger  
811 Fabric, etc.

Protocols/E- xample	Blockchain Type /Node Identity	Performance	Energy Efficiency
PoA/Vechain	consortium (consortium blockchain pro- tocols are also suitable for private blockchain)/permi- ssioned	10,000tps claimed, 500tps proven in history[25]	yes
PoET / Saw- tooth Lake	consortium/public	1300tps claimed	yes - timer certifi- cate instead of con- sumption of elec- tricity
PoHistory/ Solana	public/public	50.000tps claimed	yes
PoBurn/ Slimcoin	public/public	up to 1000tps claimed	partial - Hash com- puting(mining pro- cess) still exists
PBFT/Hyper- ledger	consortium/permi- ssioned	1000tps	yes - pbft process excluded hashing procedure. So do the following four pbft-like algorithms
dBFT/Neo	public/public	1000tps, potential 100.000 tps	yes
FBA/Bravo (BVO)	public/public	1500tps claimed	yes
Ripple/Ripple	consortium/public	1500tps	yes
Stellar/Stellar	public/public	1000tps	yes

Table I-2. Comparison of consensus protocols for blockchain type, performance and energy saving level. 2) Performance: Blockchain performance is generally measured

816 by transactino processing delay and network throughput. These two factors can be  
817 indicated by “transactions (processed) by second”.

818 We can see that dpos and Ripple have most extraordinary performance. We can  
819 also notice that it’s hard to prove the maximum performance claimed by a lot of  
820 protocols.

821 3) Energy Saving: As for PoW and some of its variants such like PoBurn[15],  
822 PoHistory, the demand on hash computing power make the system environment  
823 unfriendly; as for PoS and its variants such like dPoS, dPoW, the competition of  
824 hash computing power is removed, but the mining process is stille kept[10],[17],[18];  
825 Regarding PBFT, FBA series protocols, there is no more concept of mining, the  
826 block generation phase is somehow centralized and thus saved power tremendously.



827

Protocols/E-sample	Adversary ance Ability	Toler- ance	Scalability(Openess and Expandability)	Decentralization
PoW/Ethereum	<25% power	computing	Open  Lack of expandability due to low performance	Relative centralization: decentralization gradually lost with pow
PoS/Peercoin	<51% stake		Open and Expandable	Relative centralization: first mover advantage with pos
dPoW/Komodo	<25% power	computing	Open  Lack of expandability due to dependence on pow protocols	Relative centralization: dependency on pow and pos protocols
dPoS/Bitshares	<51% validators		Open and Expandable	Relative centralization: voting results can be highly involved by top users
Algorand / Algorand	<33.3% voting power	byzantine	Open and Expandable	Decentralization guaranteed
PoC/Burstcoin	<25% power	computing	Open and Expandable	Decentralization guaranteed
PoA/Vechain	<51% validators		Open and Expandable	Relative centralization: authority validators mechanism is too centralized

828 *Table II-1. Comparison of consensus protocols for attacker tolerance, scalability*  
829 *and decentralization level.* 4) Adversary tolerance ability: Considering the recent  
830 research on “selfish mining strategy”, once the controlled hash computing power of  
831 one miner party exceed 25%, the PoW security guarantee ,thus influence dPoW[18];  
832 the PoS security threshold is commonly known as 50%, same limitation for the  
833 variants of PoS; PBFT and FBA series algorithms are manufactured to manage up

834 to 33.34 defective nodes; as for Ripple, it has a more restrict reliability setting[13],  
835 which makes it only maintaining correctness when the proportion of faulty nodes  
836 in a unique node list are lower than 20%.

Protocols/Example	Adversary Ability	Tolerance	Scalability(Openness and Expandability)	Decentralization
PoET / Sawtooth Lake	potential point failure risk - highly dependent on Intel hardware enclave technologies	single risk - dependent	Restricted open(dependency on Intel hardware with SGX) and Expandable	Decentralization guaranteed
PoHistory/Solana	Unknown		Open and Unknown expandability	Unknown
PoBurn/Slimcoin	<25% power	computing	Open and Lack of expandability due to mining process and “coins burning process”	Relative centralization
PBFT/Hyperledger Fabric	<33.3% faulty replicas	byzantine	Closed	Relative centralization
dBFT/Neo	<51% validators		Open and Expandable	Decentralization guaranteed
FBA/Bravo (BVO)	Unknown		Open and Expandable	Unknown
Ripple/Ripple	<20% UNL nodes	faulty	Closed but expandable	Relative centralization: The company holds a large amount of money and controls many validation servers.
Stellar/Stellar	Unable to conclude(because of the Quorum algorithm and “quorum intersection property”)		Open and Expandable	the top 100 accounts hold 95% of the total supply

838 *Table II-2. Comparison of consensus protocols for attacker tolerance, scalability and*  
839 *decentralization level.* 5) Scalability: This factor involves two factors: the openness,  
840 whether nodes can freely join and leave the system; and the expandability, when  
841 tens of thousands, hundreds of thousands of users are online, whether the system  
842 can support with its performance.

843 Consortium chains are generally closed system; however, PoET(Sawtooth Lake)  
844 and Ripple are expandable because of its nice performance, where Fabric and Ripple  
845 is not. PBFT is not scalable with large scale network.

846 6) Decentralization: PoW will gradually losing its decentralization because of the  
847 fact that hash computing power can easily be centralized, so do dPoW, PoB, etc. As  
848 for PoS, “The poorer the poor, the richer the rich” is predictable, because the pro-  
849 tocol supports “First Mover advantage”, so does dPoS. Consortium chains generally  
850 operate under a “multi-center mechanism”: they are also relatively centralized.

Protocols/Example	Consensus process	Block generation method	Reward token distribution method
PoW/Ethereum	probabilistic(numerous forks can exist at the same time within the network)	Competitive - a. All nodes have the right to generate blocks b. Nodes compete to win the insertion on the blockchain	Coins - Emitted in proportion to amount of network activity
PoS/Peercoin	probabilistic	Competitive	Coins - Emitted in proportion to amount of network activity
dPoW/Komodo	probabilistic	Competitive	Coins - Emitted in proportion to amount of network activity
dPoS/Bitshares	deterministic(Only one or a very few forks can exist at the same time within the network)	Cooperative - a. Only a selected nodes have blocks generation right b. Selected nodes principally take turns in blocks generation	Coins - Emitted in proportion to amount of network activity
Algorand / Algorand	deterministic	Cooperative	No new tokens created
PoC/Burstcoin	probabilistic	Open and Expandable	No new tokens created
PoA/Vechain	deterministic	Cooperative	No new tokens created

852 *Table III-1. Comparison of consensus process, block generation method and reward*  
 853 *token distribution method.* 7) Consensus process: This column describes in which  
 854 way corresponding protocol reaches the global consensus view. With deterministic  
 855 process, normal nodes almost don't need to update local chain because of fork  
 856 problem. As for probabilistic process, forking occurs quite frequently. Naturally,  
 857 deterministic process can save a lot of communication messages and communications  
 858 rounds.

859 However, to make a reliable deterministic consensus protocol, the messages for  
 860 communicating before the block generation are often heavy. So there's this trade-  
 861 off.

862 8) Block generation type: The way of block generation is one of the most funda-  
 863 mental difference about how different protocols reach consensus. As for competitive  
 864 consensus: a decentralized competition exists for the generation of block of every  
 865 round, it protects the fairness for all the system users(nodes), but also costly in  
 866 terms of time and energy; a cooperative consensus generally centralizes the block  
 867 generation phase, in order to have a better performance and energy efficiency.

Protocols/E-sample	Consensus process	Block generation method	Reward token distribution method
PoET / Sawtooth Lake	probabilistic	Competitive	No new tokens created
PoHistory / Solana	probabilistic	Competitive	Unknown
PoBurn / Slimcoin	probabilistic	Competitive	Unknown
PBFT/Hyperledger Fabric	deterministic	Cooperative	No new tokens created
dBFT/Neo	deterministic	Cooperative	No new tokens created
FBA/Bravo (BVO)	probabilistic	Cooperative	No new tokens created
Ripple/Ripple	probabilistic	Cooperative	No new tokens created
Stellar/Stellar	probabilistic	Cooperative	No new tokens created

Table III-2. Comparison of consensus process, block generation method and reward token distribution method. 9) Reward token distribution method: there are two series of protocols in general: in pow-like protocols such as pos, dpos, we distribute incentive tokens(such as cryptocurrencies) to block generator nodes[10],[17]. This method serves mostly for public systems.

In PBFT-like protocols such as Algorand[21], Ripple[13], dBFT, we do not give incentive tokens to encourage block generators, but to network managers. Which means, by cancelling block reward, these protocols keep the transactions fees as the reward of collecting and validating transactions. This method serves mostly for consortium blockchains, as for these systems, in most of the time only a selected nodes have the right to generate block. But these super nodes are still worthy being rewarded because of maintain the network.

Protocols/E- xample	Algorithm within (incentive) protocol	used consensus	Language	Github release ver- sion & last commit
PoW/Ethe- reum	Ethash		Golang, C++, So- lidity, Serpent, LLL	v1.9.3 (2019-09-03); 2019-09-03
PoS/Peercoin	SHA-256		Michaleson	v0.8.3ppc (2019-08- 27); 2019-07-30
dPoW / Ko- modo	Equihash		C++, Golang, Python	2019-8-30
dPoS/ Bitshares	DPoS		Python, C++	BitShares Core 3.3.0; 2019-09-02
Algorand / Algorand	Algorand(VRF & BA*)		Golang, Java, Python, Javascript	Unknown
PoC / Burst- coin	Shabal256		Golang, C++, So- lidity, Serpent, LLL	Burstcoin Refer- ence Software 2.4.2; 2019-09-04
PoA/Vechain	SHA-256		Golang, Java	v1.1.4; 2019-09-04

Table IV-1. Comparison of mathematical algorithms, coding language and last version&commit. 10)Algorithm used within consensus protocol: these are the encryption algorithms, or some more complicated and original algorithms, operating within the protocol on mathematical layer.

11)Language: The coding language for these fourteen representative projects. We can notice that C++, Python and Golang are the most usefule and also most used languages to developing blockchain projects.

12)Github release version & last commit: This columns records the version of the data of each project that we've listed here.



Protocols/E- xample	Algorithm within (incentive) protocol	used consensus	Language	Github release ver- sion & last commit
PoET / Saw- tooth Lake	cannot summarize		Python	v1.2.2; 2019-9-04
PoHistory / Solana	Unknown		Rust, C++	Mavericks v0.18.0; 2019-9-04
PoBurn/ Slimcoin	Dcrypt		Python, C++, Shell	Slimcoin 0.6; 2019- 5-26
PBFT/Hyp- erledger	cannot summarize		Golang, Java	v1.4.3; 2019-08-30
dBFT/Neo	SHA-256		C#	v2.10.3; 2019-9-02
FBA/Bravo (BVO)	Unknown		Javascript, C++	Bravo 0.23.0 Re- lease; 2019-5-28
Ripple/Ripple	Opencoin		Java, Go, C++	rippled Version 1.3.1; 2019-8-23
Stellar/Stellar	Opencoin		Java, Go, C++	v11.4.0; 2019-9-04

Table IV-2. Comparison of mathematical algorithms, coding language and last version&commit.

## V Proof-of-Reputation

### V.1 Design Overview

The PoR is a new concept about consensus protocol in p2p network environment for blockchain system. Its core idea is to introduce the notion of reputation of each node - which represents their individual trustworthiness within the system - into the consensus process. By considering the reputation as an overall state of node after multiple transactions, the system will assign a different weight to every node in consensus process depending on their own “reputation value”. The weight represents the capacity that nodes can influence the consensus decision making process, especially 1) the leader election process. At each round, we determine the nodes that have right to update the ledger by generating new blocks; 2) the block acceptance phase. At each round, nodes need to get synchronization about their choice on local main chain if they have multiple forks as choices.

## 907 V.2 Principles

908 A consensus protocol generally deals with 3 problems: 1) the block acceptance,  
909 namely the fork selection problem; 2) the block generation, namely a random leader  
910 election problem; 3) the problem of the issue and distribution of incentive tokens.  
911 Facing these issues, the PoR brings improvements based on existing consensus pro-  
912 tocols such as PoW, PoS, PBFT, dBFT, etc.

### 913 *Fork selection*

914 While nodes received multiple new blocks propagated from block generator nodes,  
915 they need to choose one of them to add to the end of their ledger in local, or even  
916 modify some previous blocks. This is what we call the “fork selection” problem. As  
917 the latest consensus protocol, the PoR can treat this problem with two different  
918 design models: the first, is to imitate PoW-like protocols, that nodes accept the  
919 longest chain(or the “most weighted” chain) and every block generator can propa-  
920 gate their prepared block of current round. In the global view, the convergence of  
921 fork selection of all nodes is probabilistic; the second way is, all nodes know that  
922 there is one and only one block generated and propagated for current round, so  
923 that the convergence of fork selection is deterministic: no more forking problem if  
924 all nodes act honestly. The influence of the choice among these two methods on  
925 system security and performance depends on the concrete implementation, in the  
926 existing PoR projects, both options have been selected.

### 927 *Block generation*

928 Within a blockchain system, we update the ledger through generate new data  
929 blocks, so it’s critical that all nodes should have agreement about the identity of  
930 block generator nodes for each round. The Proof-of-Reputation protocols can also  
931 treat this problem with two different design models: the first, is to imitate PoW-like  
932 and PoS-like protocols, that every node can compete for the right of generate current  
933 round’s block by investing a certain scarce token(such as hash computing power for  
934 PoW, cryptocurrency shares in PoS), the block generation is competitive seen that  
935 the generation and propagation is a competition under this mechanism; the second  
936 way is that the system builds a committee among all nodes for each round’s block  
937 generation, the member nodes of the committee takes charge of block generation  
938 in a polling manner generally. The block generation is then cooperative seen that

we centralize the block generation right to a limited group of qualified nodes, the generation and propagation of new blocks don't process in the form of a competition, but the members of the committee take turns in charge of cooperation. The influence of the choice among these two methods on system security and performance depends on the concrete implementation, in the existing PoR projects, both options have been selected.

#### *Incentive tokens' issue and distribution*

The incentive schemes is a strategy largely accepted by existing consensus protocols, of which the purpose is to make the nodes' self-interested behavior consistent with the maintenance of the system. All rational nodes would act honestly and legitimately while participating to the update and the maintenance of the ledger, because they can get reward for it from the system. With PoR, a common choice as reward token is nodes' reputation value. And, like in almost all other kinds of protocols, the issue and distribution of reward tokens of PoR are through new block generation("block reward") and new transaction completion("transaction fee").

### V.3 Advantages Analysis

As mentioned above, while operating a consensus protocol, it's necessary that the participant nodes can prove for themselves that they will obey the protocol rules, be reliable(no malicious acts).

A common practice for consensus protocols is that, the participant nodes need to invest in some certain scarce resources as a "security deposit": in PoW, we take the hash computing power invested as the "deposit", in PoS, the stakes held by the nodes become an alternative solution. While in PoR, we talk about the reputation of a node.

This design model can bring advantages to a blockchain system on numerous aspects: the performance, the energy efficiency, the decentralization level, the fairness and the security.

#### *Energy Efficiency*

Since the "security deposit" used in PoR is - instead of the hash computing power - the nodes reputation, PoR can save a lot of electricity power and computers comput-

ing power compare to the PoW-like protocols(PoW in Bitcoin, PoW in Ethereum,  
dPoW, etc), thus the PoR is more environment-friendly.

### *Performance*

The PoR protocol can improve the efficiency of consensus achievement in 2 ways:

Firstly, using the hash computing power as “security deposit” is not only costly in terms of energy consumption, but also in terms of time overhead. PoR brings improvements on the system performance by skipping the “hash puzzle resolving” step just like in PoS(using stakes as tokens for security deposits[10]), or in PoBurn(using “burned” cryptocurrency as tokens for deposits), etc.

Secondly, the nodes reputations are quantified and can be consulted within the system - which is not the case in Pow, the system can’t offer any information about the hash computing power held by any nodes. This advantage allows the “temporal centralization during block generation phase” being realizable, which means during the step of generation of subsequent blocks, the system can - based on the ranking of nodes reputation - to distribute at each time the participation rights to a limited number of nodes. This brings advantages in terms of the complexity of number of messages transmitted, and the complexity of number of rounds needed to achieve consensus during block generation step, just like in dPoS(using the ranking of stakes to form the temporal centralized committee) and in dBFT(using the ranking of votes from all the nodes[11]).

### *Fairness*

In the case when we define the reputation as an non-consumable and non-transportable attribute, the Proof-of-reputation can offer a better environment in terms of fairness: Node’s reputation should only be accumulated through every completed transactions of it, thus its reputation takes time to augment, it makes reputation being equivalent to the time and activity that nodes have contributed or invested into the system; time and activities are the fairest investment, because users with high or low resources(in terms of assets, etc) in the real world are all equivalent in terms of their input capacity on time and activities. There can a difference in the size of the business for high and low resource nodes, although as long as the influence of the size of the transaction is controlled about the change in reputation value by protocol design, the fairness of the reputation model for all nodes

can be guaranteed. Reputation is non-consumable, non-transportable, individual for each node, only can be accumulated through node's invested time and completed transactions, these facts make the reputation not only an attribute bound to the node itself, but also a resource that can not be obtained by or converted from any type of out-of-system resources. Rich nodes aren't able to get reputation easier than the poor ones, and node groups controlling reputation resources are difficult to formed because they cannot share their own reputation with other one, neither provide (other) resources to help allies gain reputation. It can be seen that the design of PoR not only guarantees the fairness of the reputation model, but also ensures sufficient robust decentralization of the system based on this "fairness" feature.

#### 1011 *Security*

Reputation is non-consumable, so that we don't have double-spending issue with PoR; reputation needs time to be accumulated, so that naturely PoR is resistant to Sybil attack. As for service denied attack and system taken over (by attackers) risk, it depends on the concrete implementation of PoR in considered projects.

#### 1016 **V.4 General Prototype**

A blockchain system which applies a PoR protocol would typically contain two parts:

A reputation system, which defines how the "reputation value" of each node should be quantified - depending on which factors the reputation is calculated, following which kind of formulations, and how it would change along with nodes interaction and/or system operation.

A blockchain based consensus protocol that - through all nodes' reputation value - make them having agreement about block generator nodes' identity and about the latest blockchain status, thus having agreement on records and data verification for the ledger. Based on this design, we can formalize the problem of designing a prototype of a PoR consensus protocol for public or controlled blockchain system as follows. Assume  $N_{max}$  the size of maximal possible joiners for the network,  $N$  the current number of users - registered or not, depending on whether the blockchain is controlled. An individual participant can be represented by  $n_i$ ,  $i \in N$ , where  $n$  means "node". Each node stores all other peers' public key in local, it's allows every node to complete data verification tasks (for transactions and for blocks).

1033 Transactions proposed from  $n_i$  to  $n_j$  is denoted as  $\text{Sig}(x_i^j)$ : where  $x_i^j \in \mathbb{R}$  - a real  
 1034 number representing considered transaction's index - signed by  $n_i$ 's private key.

## 1035 **VI State of the Art of the Proof-of-Reputation**

1036 As mentioned in the last section, the PoR is a new concept of consensus protocol.  
 1037 Its idea is to introduce the reputation—or the trustworthiness of a node in the  
 1038 network—as the weight that this node influences the consensus. However, how to  
 1039 calculate reputation, how to make the reputation of the node affect the consensus  
 1040 process - block generation, chain fork selection, choice on incentive mode, and so  
 1041 on, different researcher groups have proposed different designs and/or methods. In  
 1042 this section, we will highlight 4 different designs of existing PoR based projects.

### 1043 **VI.1 PoR p2p**

#### 1044 *Background*

1045 The first model is from “Proof of Reputation: A Reputation-Based Consensus  
 1046 Protocol for Peer-to-Peer Network”, published in 2018 by National University of  
 1047 Defense Technology in China.

#### 1048 *Design Overview*

1049 The consensus protocol in this paper is designed for the permissioned blockchain:  
 1050 before joining the network, the identity of the node needs to be verified and recorded  
 1051 by the system.

#### 1052 *Design for consensus layer*

1053 The block generation and the fork selection are decisive in this system: nodes can  
 1054 collect transactions broadcast on the Internet into their own pool of pre-committed  
 1055 transactions. When the number of transactions in the pool exceeds the threshold,  
 1056 they can be assembled into one transaction block. However, the node can sign and  
 1057 publish this block only if it has the highest reputation value among the nodes  
 1058 involved by the transactions within this block.

#### 1059 *Design for reputation model*

1060 In the reputation model designed by the research team, the reputation of the node  
 1061 cannot be costed and transferred, and it can accumulate as the node participates in  
 1062 the network transactions (there may be negative growth). The numerical value of

1063 reputation is mainly used as an incentive for nodes to maintain and update system  
1064 ledgers.

1065 The change in reputation is mainly due to the system rewards obtained by par-  
1066 ticipating in the ledger update, as well as the rating scores obtained from other  
1067 nodes in ordinary transactions. In order to exclude the influence of human sub-  
1068 jective evaluation, the rating score only includes two cases: positive evaluation or  
1069 negative evaluation. In this case, only 1 bit needs to be used to store the scores that  
1070 affects node's reputation value. The research team calls it the "single-bit reputation  
1071 system".

## 1072 VI.2 Aigents

### 1073 *Background*

1074 The second model is from "A Reputation System for Artificial Societies", pub-  
1075 lished in 2018 by Aigents Group in Russia and SingularityNET Foundation in  
1076 Netherlands.

### 1077 *Design Overview*

1078 The Aigents team wants to - through a reputation value model - introduce the  
1079 concept of "liquid democracy" into their blockchain network: when a node gets  
1080 good reviews from other nodes, it's equivalent to the latter giving the former the  
1081 positive impact of their own reputation. Therefore the former gains a higherweight  
1082 in the process of cosensus(and other potential operations). This is like a democratic  
1083 voting process that, in some systems, voters may not vote directly, but delegate  
1084 their voting rights to other delegates, while retaining the right to withdraw their  
1085 authorization.

### 1086 *Design for consensus layer*

1087 The PoR designed by the research team is a variant of PoW. The nodes still com-  
1088 pete with each other to win the opportunity to participate in the ledger maintenance  
1089 and accept the token rewards, the only difference is that tokens placed in the com-  
1090 petition are the reputation value of the node, the rewards are also the reputation  
1091 value.

1092 The research team tried to adopt their protocol for the general public systems, es-  
1093 pecially social networks. For this reason, the storage and confirmation of reputation

status is very important. They proposed a gossip agreement to solve this problem: during the operation of the system, set a special reputation calculation cycle. All nodes broadcast the reputation data status of themselves and their own connected nodes in the network; for the reputation value of a certain node  $i$ , if node  $j$  receives enough consecutive and consistent data states, it regards it as valid. If an inconsistency (controversy) occurs, node  $j$  needs to warn the system's monitoring service and declare the source of the dispute, and validate the most important consecutive status.

#### *Design for reputation model*

The Aigents team considered five factors and four roles to construct a node's reputation. These roles are: a. "follower". When node  $i$  follow node  $j$ , it means that ratings from  $j$  to its connected nodes directly affect rating from  $i$  to the same nodes; b. "peer". Two nodes lacking the ability to influence each other's reputation and given ratings. c. "Opinion leaders". Nodes that are followed by a large number of nodes. Their ratings affect greatly the reputation of nodes being evaluated. d. 'connector'. Nodes that can connect two peer groups that are not connected.

The mentioned roles play an important role in five factors, these factors are:

- a. The direct rating from node  $i$  to node  $j$ . This will affect the reputation value data of  $j$  in front of followers of  $i$  and  $i$ .
- b. The indirect rating from node  $i$  to node  $j$ . This rating can be viewed publicly. For example, after the node generates a block, involved transactions participants can give a rating to this block; or the node leaves work like articles on the blockchain, nodes can evaluate its work. These ratings affect the reputation value of node  $j$  in public.
- c. Implicit indirect evaluations. For example, in social networks such as forums, nodes' post can receive comments. These comments are not direct ratings, but also contain positive or negative emotions.
- d. Implicit direct evaluation. For example, in social networks, node  $i$  quotes and/or excerpts from the comments or articles of node  $j$ .
- e. The financial status of the node itself. Holding stakes, conducting transaction activities can be regarded as a positive evaluation, while canceling transactions or returning goods can cause a decline in reputation.



## 1126 VI.3 Gochain

### 1127 *Background*

1128 This model is a PoR protocol proposed by its business team in 2018. The Gochain  
1129 blockchain project is developed based on Ethereum platform, dapps and smart con-  
1130 tracts running on Ethreum can be transformed on GoChain without any obstacles.

1131 The Gochain team aims on 1300tps; as for energy saving, their goal is to save 100  
1132 times more energy than Bitcoin or Ethereum. Maintaining decentralized features  
1133 and enabling more flexible intelligent contracts are also part of their work plans.

### 1134 *Design Overview*

1135 This protocol is based on the Clique algorithm which belongs to the serie of Proof  
1136 of Authority(PoA) algorithms[20], created by the Ethereum community. Its mode  
1137 of operation is that among all nodes within the network, only a selected set called  
1138 authoritative nodes(or super nodes) can play the role of “miners”, they have the  
1139 right to sign and publish - in a polling manner - the transaction blocks.

### 1140 *Design for consensus layer*

1141 Firstly, the Gochain team noted the fact that corporate reputation and orga-  
1142 nizational resources far exceed personal credit and personal resources, thus they  
1143 decided they not to allow individual users to become authoritative nodes: only 50  
1144 listed companies with sufficient reputation and assets can enter the initial system’s  
1145 authoritative nodes committee. Besides, unlike the blockchain that uses the Clique  
1146 algorithm which is currently a side chain of Ethereum, the Gochain team has built  
1147 its own blockchain system and network.

1148 In Gochain’s PoR protocol, the authoritative nodes are responsible for the assem-  
1149 bly and signing of subsequent blocks in a polling manner, so there is a concept of  
1150 “node on duty”: block published by the “on duty node” enjoys a higher weight,  
1151 thus reducing the risk of chain fork.

1152 The concept of “rounds” is preserved. Which means, any miner nodes can only  
1153 propose one block in the same round, and then they need to wait for an enough  
1154 long interval to propose an another block in a certain subsequent round, this design  
1155 can curb the ability of the malicious miner node to use the authority to destroy the  
1156 system service.

### 1157 *Design for reputation model*

1158     The renewal of the authoritative node relies on the binary voting from the mem-  
1159     bership of the committee. When a miner receives enough negative votes, it will  
1160     be removed from the committee; when there is a vacancy in the committee seat,  
1161     and a normal node receives enough affirmative votes, it can enter the committee.  
1162     The agreement proposes the concept of “epoch” as a cycle of updating the list of  
1163     committee members.

1164     Since the concept of reputation is only once used to determine the initial authoritative nodes list, in Gochain protocol, we didn’t implement any mathematical models  
1165     for reputation values.  
1166

## 1167 VI.4 Bitconch

### 1168 *Background*

1169     This model was proposed by a business project “Bitconch”, on October 3, 2018,  
1170     the research team of Bitconch released their newest test results, showing that with  
1171     their public and distributed blockchain network configured in 5 different countries,  
1172     they have achieved a peak speed up to 120,000 TPS, which is one of the fastest  
1173     blockchain under the same operating conditions at present.

### 1174 *Design Overview*

1175     The design of this model consists of 2 parts: a Proof-of-reputation consensus  
1176     protocol and a corresponding reputation system called “Bit-R”. Their PoR protocol  
1177     is a combination of a “dPoS-like or dBFT-like leader election mechanism” and  
1178     “classical PBFT algorithm”. It’s the basic protocol of Bitconch’s blockchain system;  
1179     as for the Bit-R system, it uses the quantified results of users’ trustworthiness,  
1180     activity and contribution, to build the portraits of users’ individual behavior, thus  
1181     provide a reference to the weight of each user for the election phase of their protocol.

### 1182 *Design for consensus layer*

1183     • Here’s a concrete description about how Bitconch’s PoR protocol works:

1184         a. The nodes that have the the 5% highest reputation value form a candidates  
1185         pool, each node among them is possible to be chosen to become the leader node.  
1186         The membership of this pool updates quartly.

1187 The size of the candidates pool varies from 50 to 300, depending on the scale  
1188 of the Bitconch blockchain network.

1189 b. With a priorly determined random number generation algorithm and the  
1190 candidates pool, the system conducts the election phase by selecting 1 node to  
1191 become the leader, then (M-1) other candidates - at the same time - to become  
1192 voter nodes.

1193 M is a natural number, the election of the M nodes - the leader and the voters -  
1194 is re-executed for each round within the system.

1195 c. The leader node and the voter nodes make consensus through the PBFT  
1196 algorithm: the leader takes charge of the broadcast of the uncommitted transactions;  
1197 the voters validate these transactions(or the opposite) - in Bitconch system we  
1198 describe this step as a voting action; then the leader synchronizes the voting results  
1199 and the round number with all the nodes in the network.

1200 If more than  $2/3 \cdot m$  nodes returned their voting choice(namely, committed their  
1201 validation), this round is considered as succeed, the leader and the voters gain  
1202 benefits in terms of their contribution in Bit-R system.

1203 During a successful round, a transaction that received enough certification votes  
1204 is validated(confirmed). It will be added into the ledger while the leader synchro-  
1205 nizing all the nodes. The nodes involved by a confirmed transaction gain benefits  
1206 in terms of their activity in Bit-R system.

#### 1207 *Design for reputation model*

1208 • Here is the description of reputation model within the Bitconch system:

1209 a. Activity:  $D(E,t) = \sum_{i=1}^k E_i^{\log(D_r)}$

1210  $E_i$  represents the asset weight of a transaction i,  $D_r$  represents the reputation  
1211 weight of the other party of transaction i.

1212 Thus the “activity” parameter of an user can be quantified by the transactions  
1213 that he/she has participated, and the nodes that he/has has interacted with. The  
1214 logarithm function is used here to avoid potential Sybil attacks - nodes with low  
1215 reputation weight are hard to influence other one’s activity.

1216 b. Coin age:  $T(s,t) = \beta + \alpha \log(S_t)$

1217  $S_t$  represents the length of time that current user keeps the Bitconch system  
 1218 tokens. The Bitconch system take the users who hold system rights for long-term  
 1219 are more trustworthy.

1220 The logarithm function is used here to limit the potential Matthew effect(first-  
 1221 mover advantages).

1222 c. Contribution:  $C(N,t) = \sum N_{file} + \log N_{Rnd}$

1223 The “contribution” parameter reflects the frequency that nodes contribute to  
 1224 the normal operation of the system, especially including files sharing( $\sum N_{file}$ )  
 1225 and ledger updates( $\log N_{Rnd}$ )

1226 d. Summary: Based on 3 above parameters, the Bit-R is able to describe the  
 1227 integrity of each user, thus able to give nodes’ integrity as a proof, to allow them  
 1228 to participate to the consensus, to contribute their network resources, and to gain  
 1229 rewards token.

## 1230 VI.5 Repucoin

### 1231 *Background*

1232 Repucoin was proposed in February 2019 by a research team from the University  
 1233 of Luxembourg. The proudest design objective reached by Repucoin is the resistancy  
 1234 to 51% computing power attack. Repucoin system calculates voting rights based on  
 1235 miners’ reputation. By builing a model of reputation with stringent mathematcial  
 1236 literacy, the system requires miners to accumulate long-term, continuous and honest  
 1237 mining operations.

1238 A Repucoin blockchain can support more than ten thousands tps, even much  
 1239 higher than Visa which can support around 1700 tps.

### 1240 *Design Overview*

1241 Repucoin blockchain system is deterministic: generally, only one node has the  
 1242 right to package and sign the next block at each round.

1243 The generation of blocks is cooperative: not everyone but only a selected set of  
 1244 nodes can be randomly elected to become block generator. This group takes also  
 1245 the validation of new blocks in charge.

1246 The selected group of nodes is called as the “cosensus group”, it is constituted by  
 1247 nodes who have the highest reputation scores. A ramdonly chosen leader is elected

from the membership at each “epoch” and this leader takes charge of the production of blocks of the whole current “epoch”. Epoch is a period of time determined by a chunk of blocks on blockchain.

Blocks in Repucoin system are divided into two types: keyblocks and microblocks. Miners use PoW protocol rules to compete to become the leader(block generator) for next epoch, by resolving Repucoin’s original hash puzzle. Microblocks are signed and proposed by the current leader to record the transactions into the blockchain.

#### *Design for consensus layer*

The consensus process in Repucoin system can be divided into two parts: a periodical election based on PoW mechanism, then a regular blocks validation process based on PBFT mechanism.

During the election phase - which is also the beginning of each epoch - a consensus group having X members is firstly updated. The size of X is determined by meeting a target percentage in global decision power, and the decision power is directly and only based on nodes’ cumulative reputation scores.

#### *Design for reputation model*

The reputation scores calculation model is designed as a sigmoid function: for beginners and high scores holders, the changing on their scores is slow or even towards stagnation. As for mature participants, users who joined the system for a while and honestly acted so long, they have the opportunity to enjoy potential high-speed returns.

As the calculate method is a sigmoid function, system designers can control the slope and also inflection point of function by two parameters that can be pre-determined. Here’s the simplified equation for reputaion score R:

$$R = \min(1, H * (Ext + \frac{1}{2} * (1 + \frac{y1 * y2 * L - a}{\lambda + |y1 * y2 * L - a|}))) \quad (1)$$

where  $\lambda$  and  $a$  are two parameters pre-defined by the designers to adjust the slope and the inflection point.

H is a boolean value, which is set to 1 for every newly joined user, and can be reset to 0 once if a node has misbehaved(especially as a miner).

Ext is a reputation judgement from external resource.

1278 The meaning of  $y_1$  and  $y_2$  are slightly more complicated:  $y_1$  is calculated by the  
 1279 percentage

## 1280 VII Conclusion

1281 Blockchains, with their core characteristics of decentralization, anonymity,  
 1282 tamper-resistancy, forge-resistancy and auditability, have shown their potential to  
 1283 transform the traditional business.

1284 In this article, we provide a complete overview of blockchain models and  
 1285 blockchain basic rules (consensus protocols). We first outline blockchain technology,  
 1286 giving a general model of the system itself. Then we discuss the standard consen-  
 1287 sus protocols used in blockchains. We analyzed and compared these protocols from  
 1288 different perspectives.

1289 In addition, we highlight the concept of proof-of-reputation, explaining its po-  
 1290 tential advantages to the existing ones by listing the potential solution to some  
 1291 challenges and problems by implementing PoR, and summarize some of the exist-  
 1292 ing PoR blockchain projects for indicate their features and for show how the real  
 1293 PoR protocols look like. At present, the applications based on blockchain are rising,  
 1294 and we plan to do further researches and works on original PoR based blockchain  
 1295 system in the future.

## 1296 Appendix

### 1297 List of abbreviations

1298 The following table describes the significance of various abbreviations and  
 1299 acronyms used throughout the thesis. The page on which each one is defined or  
 1300 first used is also given. Nonstandard acronyms that are used in some places to  
 1301 abbreviate the names of certain white matter structures are not in this list.

	Abbreviation	Meaning	Page
	P2P	Peer to Peer	2
	PoW	Proof of Work	9
	PoS	Proof of Stake	2
	dPoS	delegated Proof of Stake	9
	dPoW	delayed Proof of Work	14
1302	PoET	Proof of Elapsed Time	15
	PoC	Proof of Capacity	18
	PoB	Proof of Burn	18
	PBFT	Practical Byzantine Fault Tolerance	2
	dBFT	delegated	9
	FBA	Federated Byzantine Agreement	9

#### 1303 Author details

1304 <sup>1</sup>LIRIS Laboratory, National Institute of Applied Sciences of Lyon, 20 avenue Albert Einstein, 69100 Villeurbanne,  
 1305 FR. <sup>2</sup>The University of Passau, Innstraße 41, 94032 Passau, Germany.

#### 1306 References

- 1307 1. G. W. Peters, E. Panayi, and A. Chapelle, "Trends in crypto-currencies and blockchain technologies: A monetary  
 1308 theory and regulation perspective," 2015. [Online]. Available: <http://dx.doi.org/10.2139/ssrn.2646618>
- 1309 2. G. Foroglou and A.-L. Tsilidou, "Further applications of the blockchain," 2015.
- 1310 3. A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and  
 1311 privacy-preserving smart contracts," in Proceedings of IEEE Symposium on Security and Privacy (SP), San  
 1312 Jose, CA, USA, 2016, pp. 839–858.
- 1313 4. B. W. Akins, J. L. Chapman, and J. M. Gordon, "A whole new world: Income tax considerations of the bitcoin  
 1314 economy," 2013. [Online]. Available: <https://ssrn.com/abstract=2394738>
- 1315 5. Y. Zhang and J. Wen, "An IoT electric business model based on the protocol of bitcoin," in Proceedings of 18th  
 1316 International Conference on Intelligence in Next Generation Networks (ICIN), Paris, France, 2015, pp. 184–191.
- 1317 6. M. Sharples and J. Domingue, "The blockchain and kudos: A distributed system for educational record,  
 1318 reputation and reward," in Proceedings of 11th European Conference on Technology Enhanced Learning  
 1319 (EC-TEL 2015), Lyon, France, 2015, pp. 490–496.
- 1320 7. C. Noyes, "Bitav: Fast anti-malware by distributed blockchain consensus and feedforward scanning," arXiv  
 1321 preprint arXiv:1601.01405, 2016.
- 1322 8. Castro, Miguel, and Barbara Liskov. "Practical Byzantine fault tolerance." OSDI. Vol. 99. 1999.
- 1323 9. Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. 2008.
- 1324 10. Vasin P. Blackcoin's proof-of-stake protocol v2[J]. URL: [https://blackcoin.](https://blackcoin.co/blackcoin-pos-protocol-v2-whitepaper.pdf)  
 1325 [co/blackcoin-pos-protocol-v2-whitepaper.pdf](https://blackcoin.co/blackcoin-pos-protocol-v2-whitepaper.pdf), 2014, 71.
- 1326 11. Crain T, Gramoli V, Larrea M, et al. DBFT: Efficient byzantine consensus with a weak coordinator and its  
 1327 application to consortium blockchains[J]. arXiv preprint arXiv:1702.03068, 2017.
- 1328 12. Mazieres D. The stellar consensus protocol: A federated model for internet-level consensus[J]. Stellar  
 1329 Development Foundation, 2015.
- 1330 13. Schwartz D, Youngs N, Britto A. The ripple protocol consensus algorithm[J]. Ripple Labs Inc White Paper,  
 1331 2014, 5.

- 1332 14. Chen L, Xu L, Shah N, et al. On security analysis of proof-of-elapsed-time (poet)[C]//International Symposium  
1333 on Stabilization, Safety, and Security of Distributed Systems. Springer, Cham, 2017: 282-297.
- 1334 15. P4Titan. Slimcoin: A peer-to-peer crypto-currency with proof-of-burn, 2014.
- 1335 16. Dziembowski S, Faust S, Kolmogorov V, et al. Proofs of space[C]//Annual Cryptology Conference. Springer,  
1336 Berlin, Heidelberg, 2015: 585-605.
- 1337 17. Larimer D. Delegated proof-of-stake (dpos)[J]. Bitshare whitepaper, 2014.
- 1338 18. Komodo: An Advanced Blockchain Technology, Focused on Freedom
- 1339 19. Solana: A new architecture for a high performance blockchain v0.8.13, 2018
- 1340 20. De Angelis S, Aniello L, Baldoni R, et al. Pbft vs proof-of-authority: applying the cap theorem to permissioned  
1341 blockchain[J]. 2018.
- 1342 21. Gilad Y, Hemo R, Micali S, et al. Algorand: Scaling byzantine agreements for cryptocurrencies[C]//Proceedings  
1343 of the 26th Symposium on Operating Systems Principles. ACM, 2017: 51-68.
- 1344 22. [gochain.io/assets/gochain-whitepaper-v2.1.2.pdf](https://gochain.io/assets/gochain-whitepaper-v2.1.2.pdf)
- 1345 23. YUAN Yong, WANG Fei-Yue . Blockchain: The State of the Art and Future Trends[J]. ACTA AUTOMATICA  
1346 SINICA, 2016, 42(4): 481-494
- 1347 24. [bitcointalk.org/index.php?topic=3026750.0](https://bitcointalk.org/index.php?topic=3026750.0)
- 1348 25. [www.reddit.com/r/Vechain/comments/97zmoy/](https://www.reddit.com/r/Vechain/comments/97zmoy/)
- 1349 26. [www.coingecko.com/fr/pièces/](https://www.coingecko.com/fr/pièces/)
- 1350 27. [www.feixiaohao.com](https://www.feixiaohao.com)
- 1351 28. [coincheckup.com](https://coincheckup.com)
- 1352 29. [blocktivity.info](https://blocktivity.info)
- 1353 30. [bitinfocharts.com](https://bitinfocharts.com)
- 1354 .