A Comparative Analysis of Current Cryptocurrencies

Lara Mauri[1], Stelvio Cimato[1] and Ernesto Damiani[2]
[1]*Computer Science Department, Università degli Studi di Milano, via Bramante 65, Crema (CR) – 26013, Italy*
[2]*EBTIC Laboratory, Khalifa University, Abu Dhabi Campus, PO Box 127788, Abu Dhabi, U.A.E.*

Keywords: Cryptographic Protocol, Blockchain, Bitcoin, Ethereum, Ripple.

Abstract: Blockchain technology is having a deep impact on the financial and technical sectors providing a mechanism for the creation of decentralized currencies and a number of applications in different fields. At the core of the technology there is a consensus protocol enabling the maintenance of a distributed ledger. In general current systems are complex schemes that implement a combination of cryptographic algorithm, distributed techniques, and incentive driven behaviour. In this paper we focus on three of the most diffused platforms, i.e. Bitcoin, Ripple, and Ethereum, and provide a comparative analysis of their most important features such as the architecture, the scripting language, the economic and security properties.

1 INTRODUCTION

Digital currencies such as Bitcoin (Nakamoto, 2008) have pioneered a new approach to tracking financial transactions. Bitcoin's breakthrough was to combine existing techniques (distributed ledgers, public-key encryption, consensus protocols, Merkle tree hashing) in an innovative way. The underlying blockchain architecture is essentially a type of distributed ledger, i.e. a digital record of ownership maintained on a distributed network of computers. The identical copies of the digital ledger (a continuously growing list of transaction records) are shared among the network participants and updated automatically every time a new transaction occurs. Since additions to the ledger are decided on a consensus basis by multiple entities, once a transaction is entered in the blockchain, the record is immutable and therefore provides an auditable history of events that cannot be modified. In addition, blockchain is censorship resistant, which means that no actor can prevent a legitimate transaction from being added to the ledger; the integrity of the ledger is maintained by reaching a consensus about its state. Ledgers are the basic consensus technology of transaction at the heart of all modern business models. The traditional approach to certify the correctness of transactions requires complete reliance upon a single centralized authority with formal responsibility over the system. On the contrary, the blockchain architecture acts as an alternative value transfer system that breaks the paradigm of centralized consensus by using a ledger replicated among different nodes in a peer-to-peer network. Such decentralized scheme represents a completely new means of forming consensus reliably across time. Moreover, unlike highly centralised systems in which there is a single point of failure, the blockchain design does not need to rely on a trusted third party thus eliminating the risks that come with data being held centrally. Indeed, the system is designed so that it is maintained collaboratively by the decentralised network of nodes that can initiate direct transactions of data.

Distributed ledgers are highly efficient because authorized ledger changes are immediately reflected in all copies of the record. Also, changes by participants without the necessary permission to modify the blockchain are rejected, so corrupting the ledger is extremely difficult. Although distributed ledger technology (DLT) was created to allow a global network to securely transact and exchange value without the need for a third-party organization in the middle, firms and other institutions are now actively exploring the practical applications for blockchain technology beyond the financial world (Swan, 2015), particularly in the areas of health, science, government, culture and art. Thanks to their immutable nature and processing capability, these digital technologies can be applied to a wide range of industries and services and thus facilitate the grow of a sharing economy.

Contribution In this paper, we study the broad field of DLT. We provide a comparative analysis across

Mauri, L., Cimato, S. and Damiani, E.
A Comparative Analysis of Current Cryptocurrencies .
DOI: 10.5220/0006648801270138
In *Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018)*, pages 127-138
ISBN: 978-989-758-282-0

the most widely diffused platforms, and specifically, we scrutinize the design of Bitcoin, Ethereum and Ripple, with the intent of pointing out the differences in strategies adopted by them. We devote particular attention to implementing choices, functionalities and related concepts. More in details: in Section 2 we consider the alternatives in types of membership allowed by the models, along with other components. Then, in Section 3 we stress the role of blockchain in the actualization of a structural change in the economy. Section 4 analyzes the consensus models used by the different protocols. Section 5 describes the scripting features, followed by an inspection of some peculiarities of the platforms in Section 6. A discussion on some security aspects is presented in Section 7 while we draw some final conclusions in Section 8.

2 BLOCKCHAIN BASED SYSTEMS

The Bitcoin cryptographic currency first popularized the concept of blockchain. Up until Bitcoin and its distributed globally-shared ledger was invented, digital assets were entirely managed by centralized authorities keeping track of all transactions (BitFury Group, 2016). In blockchain-based architectures, the lack of a central entity who exercises concurrency control over the system has shifted this responsibility to a whole network providing a new way of mitigating the risks associated with traditional systems.

Since the emergence of Bitcoin protocol, various distributed consensus models have been proposed with novel means of establishing consensus between parties and divergent areas of focus.

Ethereum is one of the most prominent second-generation blockchain technology, whose intent is to create a new model for building decentralized applications (Buterin, 2014a). The protocol is based on a combination of virtual machines and stack-based architecture, featuring a Turing-complete scripting language, which opens the doors to a hypothetically unlimited range of applications.

Even though the term “blockchain technology” is often treated as a synonym for DLT, blockchain represents one specific type of distributed ledger in which blocks of transactional data are linked to one another with data integrity maintained by cryptography. Consensus ledgers are one of the other possible structures that can be used to implement DLT arrangements and Ripple (Schwartz et al, 2014), in particular, is a type of consensus-oriented distributed database which targets financial use cases. This system is not a blockchain technology, since transactions are not added in the form of blocks that are hash chained to each other. However, similarly to the block-oriented consensus models, the Ripple network updates the ledger through a consensus process between nodes.

2.1 Models and Structural Features

The decentralized paradigm, on which the blockchain technology is based, transfers authority and trust to a network open to all and enables its nodes to transact with varying degrees of control of the ledger. As the technology matured, development efforts by financial institutions have made it possible to adapt this paradigm to existing contexts in which data is not completely transparent. Table 1 outlines the types of membership governing each model, as well as information related to transaction processing and data structure used to efficiently verify the integrity of all the transactions.

Table 1: Compared analysis of the architecture for the three platforms. Symbols: ‘✓’ means that the protocol has the feature and ‘×’ that it has not.

Blockchain-based systems			
	Bitcoin	Ethereum	Ripple
Permissioned (1) / Permissionless (2)	(2)	(2)	(1)
Public / Private	Public	Public	Public
Trade history recorded	✓	✓	✓
Data structure	Merkle tree	Patricia tree	Merkle tree
Notion of account	×	✓	✓
Multiple senders and receivers	✓	✓	×

Depending on whether or not the network nodes are required to authenticate themselves to participate in the voting process, two main categories of blockchain are distinguished: *permissionless* blockchain and *permissioned* blockchain. The former is a distributed consensus ledger in which no authorization is required to perform transaction processing. There are no restrictions on identities of nodes which act as verifiers and thus anybody can join the network to become a participant in the consensus process as well as create blocks of transactions. Bitcoin and Ethereum are open-ended systems and consequently, any node can take part in the consensus process to

advance the blockchain. Ripple takes a different direction and relies on a permissioned setup. Here only entities belonging to a predefined list of subjects with known identities can become a verifier in the system. Indeed, Ripple requires each node to define a Unique Node List (UNL), which is a set of trusted authorities. During the process execution, validating servers vote on the validity of all transactions received, and each of them consult its list of chosen validators in order to reach consensus. Since validators are preselected by an authority, all the participants in the protocol are known and trusted, meaning that they are also accountable according to laws. Permissioned blockchains are based on trust models and therefore, they better comply with existing regulations, where only trusted members are allowed to participate in the consensus; this configuration maintains compatibility with the traditional finance setting, but at the cost of reduced censorship-resistance. Transaction processing is performed in a more controlled setting, which operates in an inherently biased way. This exposes the system to dishonest behaviours by users who might be involved in fraudulent activities.

Instead, permissionless systems minimize human factor by replacing it with rigorous approaches, which are enforced automatically, thereby eliminating trust involved in interaction between parties and associated vulnerabilities.

According to (BitFury Group and Garzik, 2015), there is a second categorization of blockchain types that is based on the access to the ledger data itself: *public* blockchain and *private* blockchain. In public blockchains, including Bitcoin and Ethereum, all records are publicly transparent. Everyone can participate in the process for determining the next block to be inserted into the chain as well as read transactions. This type of blockchain is considered fully decentralized. At the other end of the spectrum, private blockchains rely on the model of user rights, where the possibility to read or modify the blockchain data is restricted to entities within an organisation. Since write permissions are kept centralized to one organization, this configuration could be defined as a traditional centralized network. However, there are nuances to the private model concept depending on the underlying degree of control. For example, since Ripple is built with a public-based architecture, it is technically a public system. Nevertheless, it can be understood as an intermediate model between the public and the private one because of its closed and monitored nature. Regardless of the type of allowed membership, one of the main benefits offered by distributed ledger technologies is that any asset referenced in a transaction is traceable through the blockchain up until its creation. Bitcoin and other systems which extend on its innovations keep an updated record of all the transactions that ever occurred in the history of the network. Ethereum also has the property that every block contains a copy of both the transaction list and the current *state* of its network. To be more precise, the state is not directly stored in each block; every block header contains only the *state root*, i.e. the root hash of an evolved Merkle tree, called *Patricia tree*, that stores the entire state of the system. These trees record key/value bindings, where the keys are addresses and the values consist of account declarations, and boost efficiency by introducing some modifications to the data structure (Ethereum Wiki (b)). In addition to the state root, every block header in Ethereum contains two other roots related to transactions and receipts respectively. The property such that each block can use the root hash of the Patricia tree to refer to the entire tree is particularly useful when the difference of state between two blocks is fairly small because unchanged data of the tree can be simply referenced without the need to store them twice.

As previously mentioned, Patricia trees are specialized kind of Merkle trees. The original application of Merkle proofs introduced in Bitcoin provides a very efficient process to verify whether a transaction is part of a block (Antonopoulos, 2014). Each header block includes the Merkle root, which is the hash of all the hashes of all the transactions in the block. The efficiency of Merkle trees is leveraged by the *simplified payment verification* (SPV), which allows light clients to verify that a transaction has been accepted by the network by downloading just the chain of block headers, instead of downloading the entire blockchain. However, since SPV nodes do not have all transactions and do not download full blocks, they can prove that a particular transaction is included in a block, but cannot prove anything about the current state (Buterin, 2015).

Multi-level data structures represent also an important scalability feature of the Ripple protocol. The Ripple ledger chain is similar to the blockchain architecture. Every ledger is meant as a notional structure that consists of a previous ledger, a transaction tree (containing the transactions that have taken place since the previous ledger) and a state tree (showing all the account balances, settings, etc.) (Ripple Labs). So, in contrast to blockchain, each Ripple ledger contains only the transactions that created the current ledger from the previous one, rather than the entire transaction history. This means that the ledger chain stores all transactions, so it is

possible to rebuild the trade history and confirm the results of a specific transaction, but the information that a node needs to store to serve in a fully functional way is reduced to just the last ledger state.

Ripple users are equipped with a pair of signing/verification keys to securely send transactions. Any proposed change to the shared global ledger, i.e. every transaction, is signed with the private key of the wallet owner and the corresponding public key is included in the transaction. The ECDSA algorithm is the basis for its public key cryptography and Ripple uses the same elliptic curve specifications as in Bitcoin except for a different leading byte in the address format. Unlike Bitcoin and Ethereum, the current Ripple system only supports transactions involving a single sender and a single receiver. Recently, Moreno-Sanchez et al (Moreno-Sanchez et al, 2017) proposed a protocol called PathJoin, providing an innovative combination of the functionality of Ripple protocol and a distributed threshold signature scheme that overcomes this limitation and performs atomic transactions.

Instead of explicitly have user identification, the Bitcoin system uses public key security in which bitcoins are linked to public keys through *unspent transaction outputs* (UTXOs). The UTXO model requires that each transaction input has a referral to the output of a previous transaction and the concept of user balance is a derived construct created by the cumulative amount of unspent bitcoins associated with the corresponding public key of the user.

So, the Bitcoin format of transaction implies that the concepts of accounts and balances are essentially defined as states of ownership. This peculiarity cannot be found in the Ethereum ecosystem, in which, on the contrary, there is a built-in notion of higher-level concepts such as accounts. In particular, Ethereum manages two types of accounts: *externally owned accounts* (EOAs) and *contract accounts*.

The externally owned account defines the basic form of account, which is controlled by a private key and akin to the abstract concept of Bitcoin accounts. EOAs have no code and are directly controlled by real human beings; therefore, the simple ownership of the private key associated with an EOA gives the ability to send *ether* (Ethereum's intrinsic currency) and messages from it.

In contrast, contract accounts, on the other hand, are controlled entirely by code and their code is executed whenever they receive instructions in the form of a transaction from an EOA

.

3 ECONOMICS

Unlike traditional payment systems, which typically involve the transfer of funds denominated in a sovereign currency, Bitcoin has its own unit of account called *bitcoin*. In essence, blockchain-based assets could be classified as bearer assets, i.e. the ownership of an asset amounts to nothing more than the knowledge of the corresponding private key. Bitcoins are native tokens with no reference to any underlying commodity and for this reason they have no intrinsic value (Murphy et al, 2015); they have no physical form and their supply is not determined by a central bank. As shown in Table 2, currently, about 17 million BTC are in circulation (CoinMarketCap). However, the Bitcoin monetary policy is based on artificial scarcity, i.e. bitcoins are created in such a way that their numbers are inherently scarce. As it stands now, the total value of Bitcoin currency units in circulation represents 79% of the total coins that will be produced. Indeed, the supply of bitcoins is programmed to grow at a steady rate and the total amount of BTC that can be created is capped at 21 million, which is predicted to be reached in approximately 2140.

Table 2: Compared analysis of the economic features for the three platforms. Symbols: '✓' means that the protocol has the feature and '×' that it has not.

	Economics		
	Bitcoin	Ethereum	Ripple
Native currency	BTC	ETH	XRP
Circulating supply	16,684,587 BTC	95,779,681 ETH	38,622,892,459 XRP
Maximum supply	21 million BTC	×	100 billion XRP
Deflationary nature	✓	×	✓

The finite monetary base is a clear indication of the fact that the currency has an inherent tendency to be deflationary. Since in Ripple the number of XRP (Ripple's native currency) starts off at an all-time maximum of 100 billion, its ecosystem has the same deflationary behaviour. However, while the Bitcoin network creates new bitcoins through the mining process and the number of BTC in circulation increases more and more slowly as time passes (until eventually stabilizing at the permanent amount of 21 million), Ripple has no fair mechanism for coin introduction into the network. As a result, XRP is

designed to be a scarce asset and its supply decreases each time a transaction fee is paid; to date, there are about 39 billion XRP in existence available to the public (CoinMarketCap). Another important component of the Ripple network to consider is the fact that it natively supports cross-currency payments, thereby allowing parties to transact in the currency they desire. Therefore, it does not only promise to facilitate the exchange between currencies within its network, but also aims to be integrated with existing financial infrastructures, becoming a bridge between traditional banking and emerging electronic payment mechanisms (Rosner and Kang, 2016; Swanson, 2015).

Currently there are many industry leaders evaluating blockchain technology and almost every major financial market institution in the world is doing DLT research. Banks and financial market leaders are adopting blockchain solutions dramatically faster than initially expected. Two market study reports from the IBM Institute for Business Value (Bear et al., 2016a, 2016b) found that in 2017, 14% of financial markets institutions and 15% of banks expect to have blockchains in production and at commercial scale. According to the studies, more than 70% of these early adopters are prioritizing blockchain efforts in order to change the role of hierarchies and trust. By replacing top-down control with distributed consensus, blockchains are seen as a means to create new business models and access new markets.

Ethereum (the second platform for market capitalization, after Bitcoin) represents one of the most exciting ongoing developments in the blockchain scenario (Davidson et al, 2016). Ether (Ethereum's value token), is meant as a system resource needed to execute smart contracts, i.e. the mechanism carrying out the direct exchange of values between untrusted parties. Unlike the other two protocols, Ethereum has infinite monetary base, meaning there is no hard limit for this particular coin; the issuance of ether occurs at a constant annual linear rate via the mining process, but the rate of growth of the supply is not constant. The amount of ether available at the present time and circulating in the market is roughly 96 million (CoinMarketCap).

4 CONSENSUS MECHANISMS

Blockchain technology enables entities to transact directly even though they do not trust each other, without the need to send proposed transactions to a centralized third-party acting as a trusted intermediary. In contrast to traditional centralized systems, participants collectively maintain the common shared ledger in a decentralized fashion allowing a faster reconciliation between transacting parties. Each node keeps an individual copy of the blockchain, whose state is updated as newly-mined blocks are added. However, nodes may have completely inconsistent view of data recorded on the blockchain due to the divergent order in which transactions are listed in the replicas. Therefore, participants need to coordinate among each other in order to determine the legitimate ledger and guarantee the consistency of the system across all nodes.

Achieving consensus in a distributed setting is not a trivial task due to the strong dynamism of the system and the numerous actors involved. In addition, the network has to be able to operate properly even in the presence of malicious nodes that can cause Byzantine faults. These failures were first identified and described by (Lamport et al., 1982) as the *Byzantine General's Problem*. The original problem description characterizes the case of a group of generals, each commanding a portion of an army, trying to formulate a common plan of action for attacking the surrounded enemy city. Generals communicate only through messengers to reach a mutual agreement. However, some generals might be traitors and hence have the potential motivation to lie and distort messages (such an adversarial environment is comparable to the situation of a distributed system).

From a general perspective, one of Bitcoin's main contribution is a solution to this problem by means of a consensus mechanism that safeguards the stability of the network. The consensus method enforces a temporal and unambiguous ordering of transactions in the system, thereby ensuring a unique authoritative view of the world state. *Proof-of-Work* (PoW) is the mechanism introduced by Bitcoin to reach a consensus on the distributed shared state. In essence, the algorithm requires each node to show that it has performed some amount of work in order for a set of new transactions (block) to be accepted by the network members. The PoW consists in the search for a specific hash digest that is less than a certain difficulty level fixed by the system. The idea behind proof-of-work is simple and involves repeatedly hashing the header of the newly formed block, which the participant is wishing to publish, together with a nonce until a solution with desirable properties emerges; practically, the goal is to find a hash starting with a certain number of zero-bits. Since the next block always includes the PoW hash of the last mined block, the structure resulting from the consensus process is a hash chain which grows incrementally (hence the name of *blockchain*).

Bitcoin's PoW scheme laid the theoretical foundation for modern consensus models and new ones are still emerging providing innovative functions and properties (Cachin and Vukolić, 2017). As Table 3 outlines, Ethereum currently proposes *Ethash* as its specific proof-of-work algorithm (Ethereum Wiki (a)). It requires to randomly create a dataset (initialized by the current blockchain length), which forms a DAG (directed acyclic graph) and then attempt to solve a specific constraint on it. Each Ethereum client has to generate the DAG, which is regenerated every epoch, i.e. every 30000 blocks.

Table 3: Compared analysis of the consensus models for the three platforms. Symbols: '✓' means that the protocol has the feature and '×' that it has not.

Consensus mechanisms			
	Bitcoin	Ethereum	Ripple
Verification method	PoW	PoW (Ethash)	Consensus
Involved nodes	All	All	Validators
Hash algorithm	SHA-256	Keccak-256	SHA-512Half
Bock time	10 min	~14 sec	5-10 sec
Transaction rate	Low	Low	High
Probabilistic transaction finality	✓	✓	×
Mining reward	✓	✓	×
Transaction fee	✓	✓	✓

Although the proof used in Ethereum is similar to Bitcoin PoW protocol, the point in Ethash is that it uses a different cryptographic primitive for its hashing function, called Keccak-256 (a variant of the FIPS 202 based standard (SHA-3) (FIPS 202, 2015), instead of relying on the double SHA-256 hash algorithm.

Proof-of-work represents the technique commonly used for achieving fault tolerance in a distributed system. However, there are models that use a different approach to demonstrate consistency of transactions. Among them, Ripple is a current application of Byzantine agreement which opted for a voting scheme in order to advance the state of the shared ledger. It implements a round-based process called *consensus*, whose objective is to make it possible for all the nodes to agree on which transactions to include in the last closure of the ledger. A crucial aspect of this protocol is the fact that only a group of nodes can participate in advancing the ledger, thus acting as validating servers, and the system is based on trust relationships, since each server only considers the proposals from the nodes in its UNL. In an iterative process, validators vote on the validity of transactions received and the ledger is finalized when a super-majority of 80% of votes from servers in the UNL is reached. After a consensus round completes, validating servers calculate a new version of the ledger and transmit their results to the network. In particular, they publish a signed hash of the ledger, called *validation*, whose purpose is to ensure that all participants derive the same ledger. The hash function used in Ripple is known as SHA-512Half. It provides a hash value that is calculated by applying SHA-512 to some contents; then the result is truncated to the first 256 bytes. Since in Ripple there is no process of solving the proof-of-work cryptographic puzzle, consensus is fast and ledgers are validated in seconds. As a new ledger is closed approximately every 5 seconds, Ripple takes on average from 5 to 10 seconds to confirm a transaction. This is significantly different from Bitcoin block time. Indeed, a new block is generated roughly every 10 minutes. The Ethereum network, by comparison, produces a block every 14 seconds on average.

One of the important factors that requires close attention when evaluating a blockchain platform is perhaps the transaction finality (Buterin, 2016). This criterion indicates whether a transaction included in the blockchain is truly finalized, i.e. there is no way to revert that operation. Typically, decentralized systems provide this property probabilistically. Because blocks are calculated in a distributed setting, two independent nodes can discover a new block and broadcast it almost simultaneously. This leads to the creation of a temporary *fork*, an occurrence that can be exploited by a malicious node to intentionally cause a reorganization in the ordering of transactions. Even though Bitcoin solves this problem by requiring that users accept the longest branch (i.e. the branch involving the highest amount of computational effort) as the true one, in practice, there is no system that offers truly complete finality. However, a user can consider a transaction practically final by means of the number of confirmations, which represent the blocks depth. In the specific case of Bitcoin, transactions are confirmed after the generation of six consecutive blocks and hence, a payment is confirmed after one hour on average. As a result, in systems operating in the same way as Bitcoin, transaction finality is a probabilistic concept that is heavily dependent on the waiting time required by a transaction before being considered as confirmed and then final. Both Bitcoin and Ethereum are incapable of dealing with high transaction rates because of such

a slow transaction confirmation mechanism. Ripple, conversely, is a model designed for high performance and immediate transaction finality, since once a transaction is included in the last validated ledger, it is confirmed fast and cannot be reversed.

An important function in maintaining the immutability of the blockchain system is that of incentives. As it is known, DLT works through means of game theory, since the consensus process requires co-operation between actors with unaligned interests. The game-theoretic approach plays a critical role in achieving a balanced strategy to maintain a unified version of the global ledger. As such, PoW-based systems cannot function without economic rewards for nodes participating in the creation of new blocks (BitFury Group, 2015). Bitcoin incorporates incentive mechanisms, which come in the form of mining rewards and transaction fees. The first node that successfully solves the PoW and gets to add its block to the blockchain can collect the *block reward*. Currently, it amounts to 12.5 bitcoins. This issuance is determined algorithmically: the reward halves every 210,000 blocks (roughly every four years). The next halving is expected to occur in June 2020 (Bitcoin Clock). Ethereum introduced a similar wealth distribution mechanism by which the successful PoW miner receives a static block reward of 5 ETH. The beneficiary also receives the gas expended by executing the transactions in the block, where *gas* is the fundamental network cost unit at the base of each transaction. Moreover, there is an extra reward equal to 7/8 of the static block reward for including *uncles* (i.e. stales blocks), with a maximum of two uncles allowed per block. The approach adopted by Ripple differs completely from the models just described as it does not provide a direct monetary reward for nodes support. The reason behind such lack of an incentive mechanism lies in the fact that the XRP were premined and the protocol does not have a mining process.

On the other hand, the second Bitcoin revenue stream is represented by the voluntary *transactions fees* associated with a block. If the value of a transaction input exceeds the value of the output, the net difference in value may be claimed as a transaction fee, which serves as an incentive to make sure that a particular transaction will get included into the next block. Also in Ethereum's environment, the variable fee is a way to prioritize some transactions over others. Fees in Ethereum are denominated in gas, which represents the amount of ether that covers the cost of executing a transaction. So every transaction must contain, alongside its other data, two further gas related fields: a *gasprice* value and a *startprice* value. The latter defines the maximum amount of gas that the transaction sender is willing to pay. Ether is used to purchase gas, which is bought prior to the execution of a transaction at a certain price. At the end of the transaction, if not all the gas is consumed, unused gas is refunded in ether to the sender's account; thus, ether is converted freely to and from gas as required. Instead, if the gas runs out while the transaction is being executed, this is treated as an exception: the state is completely reverted, but the ether used to purchase the gas is not refunded.

In Ripple each transaction submitted to the network requires a transaction fee specified in XRP. This cost is designed to increase based upon the transaction load (currently, the minimum transaction cost is 0.00001 XRP). The novelty associated with this kind of fee is the fact that, unlike the other two protocols, there are no beneficiaries and therefore, it is not paid to any party. Once a transaction is included in a validated ledger, the exact amount of XRP specified by the fee parameter is irrevocably destroyed. Consequently, this approach creates a sort of artificial scarcity that makes XRP more valuable and drives a tendency toward the concentration of wealth.

5 SCRIPTING

The Bitcoin UTXO model characterizes how transactions move value from transaction inputs to transaction outputs. Since bitcoins are thought of as unspent outputs associated to a public key, transfers occur in a chain of transactions consuming and creating UTXO. Each input of a transaction refers to a given previous output, which is defined by a script. So transaction validation is achieved through the execution of a scripting language, whose goal is to ascertain under which condition is it possible for a user to spend the outputs (Table 4).

Table 4: Compared analysis of the scripting features for the three platforms. Symbols: '✓' means that the protocol has the feature and '×' that it has not.

Scripting			
	Bitcoin	Ethereum	Ripple
Built-in script language	✓	✓	×
Smart contract implementation	✓	✓	×
Turing-completeness	×	✓	×

Specifically, Bitcoin has a locking script which specifies the spending conditions and an unlocking script which satisfies such requirement and allows the output to be spent (Bonneau et al., 2015).
However, the scripting functionalities of the underlying technology behind Bitcoin can be used for application domains that go beyond currency. For example, the expressiveness of the blockchain technology can be used to implement what is known as *smart contract*, a concept introduced by Nick Szabo in 1997 (Szabo, 1997). Smart contracts represent a technical advancement to the practice of law, which formalize the contractual obligations into programming code and verify them in a self-executing way, eliminating ambiguity problems of natural languages. Since contracting parties can structure their relationships without involving a trusted third party, the decentralized scheme of math-based currency systems like Bitcoin, Ethereum and Ripple makes distributed ledger networks suitable for smart contracts. In 2014, Ripple released the first prototype for *Codius*, a platform that uses the concept of smart oracles (Schwartz and Thomas, 2014) to implement smart contracts. However, the development of the project was discontinued nearly a year later (Thomas, 2015). Instead, Ethereum is specifically designed as a multipurpose platform, featuring smart contract functionality. One of the key points of this system is that the protocol implements a completely programmable blockchain enabling the execution of an unlimited variety of user-customizable smart contracts. Because of its resilience to tampering, these computer programs can be leveraged to automatically conduct transactions or perform specific actions on the Ethereum blockchain. The three protocols differ markedly in terms of scripting: Ethereum's programming language has a Turing-complete nature, Bitcoin has several limitations, including non-Turing-completeness, and Ripple does not even have a scripting language. The Bitcoin network has purposefully omitted Turing-completeness, meaning that the scripts do not support all computation (e.g. iterative loops). This restriction has the function to avoid undesired behaviours related to a problem in computer science known as the *halting problem*, according to which there is no possible mathematical way to determine whether a computer program will halt or continue to run forever. Despite the fact that Bitcoin's scripting is realized by a simple stack-based language supporting basic operations, the protocol implements a weak version of smart contracts. Thanks to an unlimited ability to implement rich logic, Ethereum takes the scripting features of Bitcoin blockchain to the next level. The protocol is designed to execute code of arbitrary algorithmic complexity, created for any purpose users deem necessary, where the only limit is given by the imagination of its developer and associated resources. User-defined operations need to be interpreted and this task is handled by the *Ethereum Virtual Machine*, which constitutes the key part of the execution model (Wood, 2014).

6 PLATFORMS PECULIARITIES

As Table 5 illustrates, the platforms have some differences in the technical features and peculiarities that derive from the different types of consensus mechanism they adopt.

Table 5: Compared analysis of the platforms peculiarities. Symbols: '✓' means that the protocol has the feature and '✗' that it has not.

Platforms peculiarities			
	Bitcoin	Ethereum	Ripple
Determinism	✓	✓	✓
Energy consumption	Wasteful	Wasteful	Efficient
ASIC resistance	✗	✓	✓

At the core, DLT creates opportunities of leveraging consensus-oriented models for transaction validation within the context of a distributed ecosystem. A consensus mechanism is the way in which the nodes agree on the value of a proposed change that then updates the shared ledger, and can be described as the set of rules and procedures that ensures consistency and authenticity. Following this idea, the logic for distributed ledger transaction processing must be deterministic because each participant in the consensus process has to be able to find the same result when verifying a transaction. Otherwise, each node would produce different outputs causing consensus failure.
Ripple makes extensive use of the concept of determinism, especially when servers communicate with each other to agree upon the finality of the state after processing the transactions as a unit. Indeed, since all nodes must publish precisely the same ledger, it is imperative that they converge on the same transaction set. If there are multiple conflicting transactions in the same round, participants can sort and execute them in a deterministic way following pre-defined rules.

Bitcoin requires nodes to solve a cryptographic puzzle, which is computationally hard by design, in order to add a new block to the blockchain. The PoW-based consensus forces participants to expend computational resources toward the purpose of ensuring the safety of the network. Since this effort has a measurable cost associated to it, Bitcoin relies on incentive mechanisms to induce nodes to solve the proof-of-work. Therefore, the PoW concept as used in Bitcoin comes along with a few drawbacks.

First of all, due to the need for enormous amounts of computational resources, the protocol consumes a lot of electricity, and hence it is wasteful in terms of energy expenditure (O'Dwyer and Malone, 2014). Secondly, since mining profitability depends on factors such as hash rate and cost of electricity, the competitive nature of the process has led miners to develop more powerful and cost-efficient customized hardware. As a result, the mining ecosystem has come to be dominated by ASICs. These are a kind of integrated circuit specifically built for a certain purpose (in Bitcoin's case the task consists in computing uniquely the SHA-256 hash function), whose power is to achieve massive gains through parallelization. With the introduction of ASIC implementations, the probability of being the first to successfully find a valid nonce has become very low for a regular user. Consequently, in order to increase their chance of winning the mining competition, participants rely on mining pools, which are groups of miners that contribute to perform the block validation jointly; in case of win they split the mining reward according the contributed processing power. Today, the majority of bitcoin mining is done in data centers, by large powerful companies that take control of the mining power. The emergence of mining pools, therefore, has caused a departure from the original Bitcoin idea of decentralization (Nakamoto, 2008).

By using Ethash, Ethereum combats mining centralization (a problem that does not affect Ripple ecosystem, since it does not require mining to achieve consensus). Its mining algorithm is meant to be ASIC-resistant, meaning that ASICs are no more efficient than general-purpose computers at mining. Ethash is designed to be memory hardness, a property where the time to compute a valid PoW derives from the amount of memory required to hold data. The function takes a very large amount of memory and this reduces the power of specialized hardware solutions, thereby encouraging individuals to use their GPUs and allowing a tangible decentralized mining process. However, the consensus process is based on the proof that a particular amount of computation has been expended in finding a value less than a pre-defined target threshold and hence, also Ethereum wastes a lot of energy. Conversely, Ripple is more environment-friendly thanks to the absence of mining. Indeed, the energy cost of its consensus process only derives from the processing power needed to update the ledger and verify the transactions.

In light of the energy expenditure, one of the proposed improvements to the current design of Ethereum involves the update of the PoW scheme (Buterin, 2014b). The idea is to move to *Proof-of-Stake* (PoS), i.e. an approach in which the probability to create a block is proportional to a user's ownership of cryptocurrency in the blockchain system. The algorithm is designed to overcome the downsides of Bitcoin-like PoW mining process by requiring all nodes to compete with their node fraction of stake. So it is based on the distribution of digital currency within the system rather than computational resources. The proposal for this new scheme, known as Casper, is still in testing, but a rough implementation guide has recently been released (Ethereum Research).

7 SECURITY ASPECTS

By design, once a transaction is added to the blockchain and confirmed, it can never be reversed. The existence of multiple shared copies of the same ledger makes these systems inherently harder to attack than centralized solutions. However, even though data integrity is one of the key point of blockchains, distributed ledgers are not invulnerable to cyber-attack because legitimate changes to the global record can be made in principle by anyone.

Table 6: Compared analysis of the security aspects for the three platforms. Symbols: '✓' means that the protocol is resistant in principle to the attack and '×' that it is not.

Security			
	Bitcoin	Ethereum	Ripple
Double-spending attack	✓	✓	✓
Sybil attack	✓	✓	✓
51% attack	×	×	✓
DoS attack	✓	✓	✓

The double-spending problem refers to a failure case of digital cash schemes where two separate transactions sent into the network spend the same

digital currency (a double spend is hence a deliberate fork). Bitcoin solves this problem by chronologically ordering blocks of transactions into an ongoing chain of proof-of-work that is visible to all users. Therefore, by implementing a confirmation mechanism and imposing the rule that bitcoins can be spent only from UTXO, Bitcoin naturally defends against it. Similarly, Ethereum uses its PoW-based algorithm to prevent the risk that a user could concurrently spend the same unit of currency in several transactions.

However, the previously mentioned Bitcoin's SPV method, is not always effective in resisting double-spending attacks. In (Karame et al, 2012), authors investigate the problem and demonstrate that these attacks can be performed in spite of the measures recommended by Bitcoin developers.

Ripple offers an alternative solution to the double-spending problem through its consensus process. It is resistant to the attack because transactions are considered validated only when an overwhelming majority of validators sign validations for the same ledger. Since the process requires agreeing on an order for the transactions, if two transactions are a double spend, the attack is solved simply by agreeing on which of the two transactions comes first (the other is considered invalid, and hence not applied).

As shown in Table 6, all three platforms are immune to Sybil attacks (Douceur, 2002). Reputation-based systems are susceptible to this type of attack, in which a malicious agent assumes multiple pseudonymous identities with the goal to gain a disproportionately large influence. By acquiring control over a substantial fraction of the system, the adversary is able to manipulate the voting outcomes.

Bitcoin and Ethereum prevent this attack by means of PoW. The ability of block generation is proportional to the computing power, and not to the number of counterfeit identities (Tschorsch and Scheuermann, 2015). Therefore, the capability of the attacker is determined by the number of blocks it can produce. The way in which Ripple avoids Sybil attacks is based on a strategy that is conceptually opposed to the approach adopted by Bitcoin, because Ripple does not work in an open environment where anyone can participate in the consensus process. Indeed, given that it implements a permissioned system, a Sybil-resistant strategy is superfluous. Instead of evaluating proposals from all validators in the network, a validating server needs to query only its UNL. The sole fact that consensus is reached by means of trusted relationships guarantees Sybil resistance.

In the current landscape, blockchain design is inherently vulnerable to a 51% attack (Swan, 2015). Blockchain paradigms rely on the assumption that the majority of nodes act honestly, but in principle it would be possible for a single malevolent node or organization to amass a large amount of computational power and disrupt the stability of the process. In particular, if an attacker controls >50% of the mining power, he can create an independent branch maintaining a fork. Thus if this attack is successful, a malicious actor can manipulate the ledger to his advantage and double spend his own currency, thereby rewriting blockchain history. Empirical evidence shows that this attack is infeasible for any single user, as it would require enormous computational power to recompute all proofs for all the previous blocks in the chain. However, although Bitcoin itself is purely decentralized, the declining incentive to mine leads to centralization of the mining function. (Beikverdi and Song, 2015) argue that this trend toward centralization increases the risk of a 51% attack. Another research (Eyal and Sirer, 2014) proves that the selfish attack (i.e. a method by which a coordinated group of nodes increase their returns by not publishing a valid solution to the rest of the network), facilitates the grow in size of the colluding group because honest miners strategically decide to join the selfish miners.

Ethereum shares the same weaknesses as Bitcoin about the 51% attack, but its ASIC-resistant design makes the protocol more resistant to the attack.

Ripple, as it is known, does not rely on distributed computational power to protect the integrity of the network. It replaces the vote per computing power of the miners notion of PoW based consensus mechanisms with the vote per validator (Karame and Androulaki, 2016). The underlying assumption is that the majority of Ripple nodes will not collude to manipulate the voting result. Actually, if 80% of validating servers collude, it is possible to confirm a fraudulent transaction. Thereby, the Ripple equivalent of Bitcoin's 51% attack is when a group gets control over a sufficient number of validators to cause consensus fail. As participants specifically select their own validating servers, the probability of success of the attack is very low. However, in the event that the majority of validators become malicious, they can rewrite the entire system transaction history (Armknecht et al., 2015).

Finally, since distributed ledger technologies are based around a public ledger of information maintained by a network of computers around the world, adversaries could broadcast large amounts of transaction spam in an attempt to disrupt the normal operation of the network. The most visible consequence of such attack is the creation of excess load on the network, which causes difficulties in

processing legitimate transactions. To mitigate denial-of-service (DoS) attacks, Ripple enforces a transaction fee. Moreover, unlike other architectures, Ripple requires each account to have a small reserve of XRP for the creation of ledger entries in order to protect the network from abusive creation of ledger spam. The current minimum amount needed to fund a new address is 20 XRP. This creates a strong disincentive against ledger spamming because any attacks aimed at wasting network bandwidth become very expensive for malicious agents.

A denial-of-service attack on Ethereum blockchain would imply that a malicious actor utilizes the network resources improperly in an attempt to interfere with the miners' ability to quickly settle legitimate transactions. Due to the Turing-complete nature of the contracts, Ethereum is inherently vulnerable to DoS attacks, as an attacker could perform a successful attack by sending transactions that loop forever. However, it protects the network against DoS attacks through the use of gas. Gas limits the number of computational step a transaction execution is allowed to take, and hence ensures that there can be no infinite loop. Similarly, Bitcoin has some DoS prevention built-in. Theoretically, it is immune to hostile infinite loops because its scripting language is non-Turing-complete. Actually, Bitcoin is vulnerable to DoS attacks (Vasek et al, 2014).

8 CONCLUSIONS

In this paper, we presented a survey specifically targeting the distinguishing features of three of the most diffused platforms, i.e. Bitcoin, Ethereum and Ripple. The compared analysis of these systems focused on their common points, as well as differences in how they maintain the integrity of data recorded on the shared ledger, by grouping them into six main categories. The work includes an accurate description of the different consensus and incentive mechanisms adopted by the platforms for securing the network. Also we examined the scripting features, security aspects and impact on the economy, as well as related concepts.

Whereas blockchain is still in its emergent and immature technological phase, the increasing interest on it is showing the importance and awareness of distributed ledgers as one of the most promising technology that will have a pervasive impact on the future of many sectors of our socio-economic systems. Indeed, the emergence of the blockchain technology could give rise to the next generation beyond the internet, potentially leading to the creation of new types of economies. Blockchain's ability to catalyse transparency is based on the way it leverages a global peer-to-peer network to guarantee integrity of value exchanged between parties without the need for central authorities. Therefore, by providing a way of recording transactions securely, distributed ledger technology offers the opportunity to reimage how the financial system can work. However, it should not be understood only as a disruptive technology, but also as a foundational technology that offers the possibility to create new foundations for the social infrastructure (Iansiti and Lakhani, 2017).

REFERENCES

Antonopoulos, A. M. 2014. *Mastering Bitcoin: Unlocking Digital Crypto-Currencies*. O'Reilly Media, Sebastopol.

Armknecht, F., Karame, G. O., Mandal, A., Youssef, F. and Zenner, E. 2015. Ripple: Overview and Outlook. In: *Proceedings of International Conference on Trust & Trustworthy Computing*, 9229:163–180.

Bear, K., Drury, N., Korsten, P., Pureswaran, V., Wallis, J. and Wagle, L. 2016a. *Blockchain rewires financial markets: Trailblazers take the lead*. Executive Report - IBM Institute for Business Value, [online] Available: https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBP03469USEN&

Bear, K., Drury, N., Korsten, P., Pureswaran, V., Wallis, J. and Wagle, L. 2016b. *Leading the pack in blockchain banking: Trailblazers set the pace*. Executive Report - IBM Institute for Business Value, [online] Available: https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=GBP03467USEN.

Beikverdi, A. and Song, JS. 2015. Trend of centralization in Bitcoin's distributed network. In: *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2015 16th IEEE/ACIS International Conference on*, pp. 1–6.

Bitcoin Clock, [online] Available: http://bitcoinclock.com/

BitFury Group 2016. *Digital Assets on Public Blockchains*. White Paper, [online] Available: http://www.the-blockchain.com/docs/bitfury-digital_assets_on_public_blockchains.pdf.

BitFury Group 2015. *Incentive Mechanisms for Securing the Bitcoin Blockchain*. White Paper, [online] Available: http://bitfury.com/content/5-white-papers-research/bitfury-incentive_mechanisms_for_securing_the_bitcoin_blockchain-1.pdf.

BitFury Group and Garzik, J. 2015. *Public versus Private Blockchains - Part 1: Permissioned Blockchains*. White Paper, [online] Available: http://bitfury.com/content/5-white-papers-research/public-vs-private-pt1-1.pdf.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. A. and Felten, E. W. 2015. SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies. In:

Proceedings of the 36th IEEE Symposium on Security and Privacy (SP'15), pp. 104–121.

Buterin, V. 2014a. *A Next-Generation Smart Contract and Decentralized Application Platform*. White Paper, [online] Available: https://github.com/ethereum/wiki/wiki/White-Paper.

Buterin, V. 2014b. *On Stake*, [online] Available: https://blog.ethereum.org/2014/07/05/stake/

Buterin, V. 2015. *Merkling in Ethereum*, [online] Available: https://blog.ethereum.org/2015/11/15/merkling-in-ethereum/

Buterin, V. 2016. *On Settlement Finality*, [online] Available: https://blog.ethereum.org/2016/05/09/on-settlement-finality/

Cachin C. and Vukolić, M. 2017. *Blockchain Consensus Protocols in the Wild*. Technical Report arXiv:1707.01873, IBM Research – Zurich, [online] Available: https://arxiv.org/abs/1707.01873.

CoinMarketCap, *Cryptocurrency Market Capitalizations*, [online] Available: https://coinmarketcap.com/

Davidson, S., De Filippi, P. and Potts, J. 2016. *Economics of Blockchain*, [online] Available: http://dx.doi.org/10.2139/ssrn.2744751.

Douceur, J. R. 2002. The Sybil Attack. In: *Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02).*

Ethereum Wiki (a), *Ethash*, [online] Available: https://github.com/ethereum/wiki/wiki/Ethash.

Ethereum Wiki (b), *Patricia Tree*, [online] Available: https://github.com/ethereum/wiki/wiki/Patricia-Tree.

Ethereum Research, *Casper Version 1 Implementation Guide*, [online] Available: https://github.com/ethereum/research/wiki/Casper-Version-1-Implementation-Guide.

Eyal I. and Sirer, E. G. 2014. Majority Is Not Enough: Bitcoin Mining Is Vulnerable. In: *Financial Cryptography and Data Security*, 8437:436–454.

FIPS 202, 2015. *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions*, [online] Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf.

Iansiti M. and Lakhani, K. R. 2017. 'The Truth about Blockchain', *Harvard Business Review* 95(1):118–127.

Karame G. O. and Androulaki, E. 2016. *Bitcoin and Blockchain Security*. Artech House, Norwood.

Karame, G. O., Androulaki, E. and Capkun, S. 2012. Double-spending fast payments in Bitcoin. In: *Proceedings of the 2012 ACM Conference on Computer and Communication Security*, pp. 906–917.

Lamport, L., Shostak, R. E. and Pease, M. 1982. The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401.

Moreno-Sanchez, P., Ruffing, T. and Kate, A. 2017. PathShuffle: Mixing Credit Paths for Anonymous Transactions in Ripple. In: *Proceedings on Privacy Enhancing Technologies*, 2017(3):107–126.

Murphy, E. V., Murphy, M. M. and Seitzinger, M. V. 2015. *Bitcoin: Questions, Answers, and Analysis of Legal Issues*, Congressional Research Service, [online] Available: https://fas.org/sgp/crs/misc/R43339.pdf.

Nakamoto, S. 2008. *Bitcoin: A peer-to-peer electronic cash system*, [online] Available: https://bitcoin.org/bitcoin.pdf.

O'Dwyer, K. J. and Malone, D. 2014. Bitcoin mining and its energy footprint. In: *25th IET Irish Signals and Systems Conference and China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014)*, pp. 280–285.

Ripple Labs, *The Ledger*, [online] Available: https://ripple.com/build/ledger-format/#tree-format.

Rosner, M. T. and Kang, A. 2016. 'Understanding and Regulating Twenty-First Century Payment Systems: The Ripple Case Study', *Michigan Law Review*. 114(4): 649–681.

Schwartz E. and Thomas S. 2014. *Smart Oracles: A Simple, Powerful Approach to Smart Contracts*, [online] Available: https://github.com/codius/codius/wiki/Smart-Oracles:-A-Simple,-Powerful-Approach-to-Smart-Contracts.

Schwartz, D., Youngs, N. and Britto, A. 2014. *The Ripple Protocol Consensus Algorithm*. White Paper, [online] Available: https://ripple.com/files/ripple_consensus_whitepaper.pdf.

Swan, M. 2015. *Blockchain: Blueprint for a New Economy*. O'Reilly Media, Sebastopol.

Swanson, T. 2015. *Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems*, [online] Available: http://www.ofnumbers.com/wp-content/uploads/2015/04/Permissioned-distributed-ledgers.pdf.

Szabo, N. 1997. 'Formalizing and securing relationships on public networks', *First Monday* 2(9).

Thomas, S. 2015. *Codius - One Year Later*, [online] Available: https://codius.org/blog/codius-one-year-later/

Tschorsch, F. and Scheuermann, B. 2015. Bitcoin and beyond: A technical survey on decentralized digital currencies. *IACR Cryptology ePrint Archive 2015:464*.

Vasek, M., Thornton, M. and Moore, T. 2014. Empirical analysis of denial-of-service attacks in the bitcoin ecosystem. In: *Financial Cryptography and Data Security*, 8438:57–71.

Wood, G. 2014. *Ethereum: A secure decentralised generalised transaction ledger*. Yellow Paper, [online] Available: http://gavwood.com/paper.pdf.