Blockchains and the commons

Maria Potop-Butucaru

Blockchains and the commons
Invited Paper ⋆

Maria Potop-Butucaru

Sorbonne Universite, CNRS, LIP6, F-75005 Paris, France

Abstract. Blockchain phenomena is similar to the last century gold rush. Blockchain technologies are publicized as being the technical solution for fully decentralizing activities that were for centuries centralized such as administration and banking. Therefore, prominent socio-economical actors all over the world are attracted and ready to invest in these technologies. Despite their large publicity, blockchains are far from being a technology ready to be used in critical economical applications and scientists multiply their effort in warning about the risks of using this technology before understanding and fully mastering it. That is, a blockchain technology evolves in a complex environment where rational and irrational behaviors are melted with faults and attacks.

This position paper advocates that the theoretical foundations of blockchains should be a cross research between classical distributed systems, distributed cryptography, self-organized micro-economies, game theory and formal methods. We discuss in the following a set of open research directions interesting in this context.

1 Introduction

Blockchain systems became today one of the most appealing area of research motivated mainly by the recent speculations on crypto-currencies such as Bitcoin [72] or Ethereum [85]. A blockchain is a distributed ledger that mimics the functioning of a classical traditional ledger (i.e. transparency and falsification-proof of documentation) in an untrusted environment where the computation is distributed. The set of participants to the system are not known and it varies during the execution. Moreover, each participant follows his own rules to maximize its welfare.

Blockchain systems maintain a continuously-growing list of ordered blocks that include one or more transactions[1] that have been verified by the members of the system, called miners. Blocks are linked using cryptography and the order of blocks in the blockchain is the result of a form of agreement among the system participants. Participants strongly agree only on a prefix of the blockchain, the suffix of the blockchain may be different from one participant to another.

[⋆] This position paper is based on the homonymous ERC Advanced submission [80].

[1] Transaction is used here as a generic name to be adapted to a broad class of use cases. For example, a transaction in Bitcoin [72] or Ethereum [85] can be a transfer of digital money or assets.

Blockchain systems, beyond their incontestable assets such as decentralization, simple design and relative easy use, are not free of incidents and limitations. The most popular incident reported for Ethereum for example was the 60 million dollars ether theft which was possible by simply exploiting an error in the code and the lack of system specification.

A recent scientific analysis, [50], focus on several limitations of the most popular blockchain, Bitcoin, such as: weak security, low quality of services, storage limitations, low throughput and high cost and weak consistency.

Therefore, despite their large publicity blockchains are far from being a technology ready to be used in critical economical applications and scientists multiply their efforts in warning about the risks of using this technology before understanding and fully mastering it. Interestingly, many recent attempts to alarm on vulnerabilities of popular blockchains like Bitcoin are target of defenders brigading.

Nevertheless, once fully mastered, Blockchain systems will be the technical solution for fully decentralizing activities that were for centuries centralized such as for example administration or banking. The applications of tomorrow that potentially will be blockchainized are all different from each other. These applications may range from IoTs to notary passing by administration, banking or health. These applications have various consistency and quality of services requirements. Therefore, we advocate that there will not be only one blockchain but a family of modular blockchains that will have to offer various qualities of services and that will be eventually interconnected.

It should be noted that differently from classical distributed applications, some blockchains have a strong economical aspect since participants should be constantly incited to participate to the system welfare by rewarding their contribution. This contribution is materialized either in the energy spent in solving cryptographic puzzles in order to generate blocks or in the bandwidth spent to route transactions and blocks. If participants massively leave the system then the system collapses, phenomenon known in economy as the *tragedy of commons*. In order to avoid this phenomenon, blockchains have to cross-over new distributed formally verified and proven algorithms with game theory tools and also government rules issued from self-organized micro-economies.

1.1 State of the art

The birth of blockchains systems, as for the case of P2P systems in the early 2000, was in the non academic research. After the releasing of the most popular blockchains (e.g. Bitcoin [72] or Ethereum [85]) with a specific focus on economical transactions their huge potential for various other applications became evident.

Their popularity, transformed blockchains in a huge social experiment that confirmed the fact that blockchains can be a viable alternative for distributed systems of tomorrow. Starting with this point, blockchain area started to became the focus of the academical research.

Interestingly, only recently distributed computing scientist started to investigate theoretical aspects of blockchains and several directions of research can be identified: blockchains based on *proof-of-work* and its alternatives such as *proof-of-stake*, *proof-of-space* or *proof-of-authority*, blockchains using as underlying building block the achievements in classical *practical Byzantine fault-tolerance* and finally *sortition* based blockchains.

The theoretical study of *proof-of-work* based blockchains has been pioneered by Garay et *al* [58]. They decorticate the pseudo-code of Bitcoin and analyze its agreement aspects considering a synchronous round-based communication model. That is, messages sent in a round are assumed to arrive in the next round. This study has been extended by Pass et *al* [77] to round based systems where messages sent in a round can be received later. The major criticisms for the *proof-of-work* approach are as follows: it is assumed that the honest miners hold a majority of the computational power, the generation of a block is energetically costly which yielded to the creation of mining pools and finally, the multiple blockchains that coexist in the system. Interestingly, the two alternatives for *proof-of-work* such as *proof-of-stake* (the power of block building is proportional to the amount of money they own in the system) or *proof-of-authority* (the power of block building is proportional to the amount of authority they own in the system) have not yet been fully analyzed from a theoretical point of view. The line of research that addresses the consensus in proof-of-stake based blockchains is pioneered by Daian *et al.* [51] that proposes a protocol for weakly synchronous networks. The execution of the protocol is organized in epochs. Similar to Bitcoin-NG [53] described below in each epoch a different committee is elected and inside the elected committee a leader will be chosen. The leader is allowed to extend the new blockchain. The protocol is validated via simulations and only partial proofs of correctness are provided.

In order to overcome the drawbacks of Bicoin, [53] proposes a mix between proof-of-work blockchains and proof-of-work free blockchains referred as Bitcoin-NG. The idea is that the execution of the system is organized in epochs. In each epoch a leader elected via a proof-of-work mechanism will decide the order transactions that will be committed in the blockchain till the next epoch. Bitcoin-NG inherits the drawbacks of Bitcoin: costly proof-of-work process, forks, no guarantee that a leader in an epoch is unique, no guarantee that the leader do not change the history at will if the leader is corrupted.

Later, [66] initiates an alternative to the proof-of-work based blockchains, named Byzcoin. Their research build on top of *practical Byzantine fault-tolerance* [45] enhanced with a scalable collective signing process. [66] is based on a leader-based consensus over a group of members chosen based on a proof-of-membership mechanism. As in Bitcoin, when a miner succeeds to mine a block it is included in the voting members set that excludes one member. This protocol also inherits some of the Bitcoin problems and vulnerabilities. Also Byzcoin voting core can be totally corrupted by a dynamic adversary. More recently, SBFT [60] and Hyperledger Fabric [21] build also on top of [45]. In the same spirit, [49] proposes for the first time a leader-free algorithm to solve Consensus among participants in a consortium Blockchain where the specifications has been adapted to the Blockchain scenario. The same specification is then considered in DBFT [48], an evolution of the consensus algorithm in [49], in Tendermint Consensus algorithm [43]. In the same line of research have been proposed recently SBFT [60] and Hot-Stuff [14].

In order to avoid some of the previously cited problems, Micali [70] introduced (further extended in [29, 46]) the *sortition* based blockchains that completely replace the proof-of-work mechanism by sortition. These works focus again the agreement aspects of blockchains using probabilistic ingredients. More specifically, the set of nodes that are allowed to produce and validate blocks are randomly chosen and they change over the time. Interestingly, the study focuses only on synchronous round-based communication models which do not reflect the reality of blockchain technologies.

In another line of research, Pass *et al.* address in [78] one of the vulnerabilities of Bitcoin studied formally in Eyal and Sirer [54]. In [54] the authors prove that if the adversary controls a coalition of miners holding even a minority fraction of the computational power, this coalition can gain twice its share. Fruitchain proposed in [78] overcomes this problem by ensuring that no coalition controlling less than a majority of the computing power can gain more than a factor $1 + 3\delta$ by not respecting the protocol, where δ is a parameter of the protocol.

A full overview of the agreement protocols designed for blockchain systems can be found in [57].

Another interesting line of research, has been opened by *Decker et al.* [52] related to the blockchains consistency. They propose PeerCensus system that targets to provide the linearizability of transactions. PeerCensus combines, similar to Byzcoin, the proof-of-work blockchain and the classical results in practical byzantine agreement fault tolerance. This line of research has been continued in [20, 24, 47, 22].

All the above mentioned studies leave a huge unexplored space in the theoretical distributed aspects of blockchains. Moreover, even though a strong effort has been recently dedicated to formalizing blockchain systems, it comes to evidence that blockchains still lack of formalization and theoretical understanding of their properties and their level of consistency face to *system asynchrony*, *churn and partitions*, *rational and irrational behaviors* and *multiple types of faults and attacks*. This important drawback limits drastically the integration of blockchains in industrial applications despite the huge interest of the main industrial actors in this technology. In the following we detail open research directions that may help in integrating blockchain solutions in practical applications.

2 Explore novel models of reliability for blockchains

Faults are studied in distributed systems for decades [25] and most of the time in isolation. Interestingly, faults and behaviors are defined in the distributed systems literature in a verbose mode which, in most of the cases leaves the place to the interpretation.

In a very popular paper, *Laprie et al.* [26] the authors describe and classify the distributed system faults, errors and failures. Interestingly, Byzantine Altruistic and Rational model, a.k.a BAR [16]. BAR model identifies three categories of processes: *altruistic*, those who follow a prescribed protocol; *rationals*, those who act in order to maximise their utility function; and *Byzantines*, those who may rationally deviate from a prescribed protocol. This later behavior can be seen as rational Byzantine behavior. In [13] the authors introduce the notion of robustness of a distributed system by introducing the notions of k-resiliency and t-immunity. In a k-resilient equilibrium there is no coalition of k players having an incentive to simultaneously change strategy to get a better outcome. On the other hand, the concept of t-immunity evaluates the risk of a set of t players to have a Byzantine behavior. It should be noted that the property of t-immunity is often impossible to be satisfied in practical systems [12].

In the context of blockchains, Micali *et al* [46] advocate that blockchains should be tolerant to *churn* and a to very powerful *dynamic adversary.* Informally speaking, this adversary "can corrupt any user he wants, at any time; totally control and perfectly coordinate all corrupted users and schedule de delivery of messages". Moreover, Blockchains area brings a new direction of research by exposing *rational behavior* with effects similar to the irrational ones. This type of behavior is extensively studied in economics theories as for example the Elinor Ostrom work [75, 64, 74].

The hierarchy of *Laprie et al.* [26] extended with the BAR model or the (k, t)-robustness model totally covers complex faults experienced in blockchains such as dynamic adversaries, churn, transient faults, rational and irrational behaviors or combinations. Therefore, several research directions need to be explored in this context.

2.1 Blockchains robustness to dynamic adversaries

The *dynamic adversary* that affects blockchains described by Micali in [46] has a Byzantine flavor and has similarities with Mobile Byzantine Adversaries studied in classical distributed systems. Intuitively, a mobile byzantine adversary can move agents from a process to another in order to deviate the process computation. When a process is infected by an adversarial agent, it behaves arbitrarily until the adversary decides to "move" the agent to another process. Most of the literature on Mobile Byzantine Adversaries [27, 44, 56, 76, 81, 83, 40] considered so far *synchronous round-based models* and only between two consecutive rounds, Byzantine agents are allowed to move from one process to another. Hence the set of faulty processes at any given time has a bounded size, yet its membership may evolve from one round to the next. It is obvious that adversaries described so far by the classical distributed literature do not match the Micali's description of dynamic adversary in blockchains. A challenge would be to explore Mobile Byzantine Adversaries decoupled from the synchronous communication of the system. However, this line of research still does not cover the dynamic adversary in blockchains and further research is needed in this direction.

Therefore, *the main challenge will be the formal specification of the robustness of blockchains face to dynamic adversaries.*

2.2 Robustness to rationality and irrationality

Common resources in blockchain systems can be seen at different levels. Participants gain a financial benefit from generating blocks. However, they bring to the system their energy. Moreover, the system itself uses participants as resources since functionalities of the system such as routing, overlay maintenance, mining or agreement, are totally dependent on the presence of the participants. The risk in these systems, as the one advertised recently for Bitcoin, is the fact that participants will leave the system and hence the system collapses. This phenomenon is known in economy as the *tragedy of commons*. Commons have similarities to the fair resource sharing in P2P networks where peers express rational behaviors. Each peer in a resource sharing system gains a certain benefit from using the system and pays a certain cost participating to it. The incentives solutions proposed so far in P2P networks (e.g. [19]) are most of the time evaluated in an empirical model without no formalization. Also these solutions are not designed to cope with dynamic adversaries.

In order to avoid the *tragedy of commons* phenomenon in blockchains, new solutions have to be designed by combining self-organized micro-economies theories (in particular the work of Nobel Prize Elinor Ostrom) with on-the-shelf tools issued from mechanisms and game theories.

A first step would be to understand the effect of various behaviors on blockchain systems. From the game perspective point of view rationality in blockchains has been studies in [30] (for the case of Bitcoin protocol) or [82] (for the case of proof-of-stake protocols). Recently, in [17] the authors explore the robustness of Tendermint consensus core to rational and Byzantine behaviors. They analyze equilibrium interactions between Byzantine and rational committee members and derive conditions under which consensus properties are satisfied or not in equilibrium. However, the proposed framework is not general enough to be applied to other blockchain building blocks.

The challenge here will be to define a unified framework for specifying rational and irrational behaviors all together with mobility of faults and attacks and propose incentive rules tolerant to these behaviors.

One possible solution is first to extend first model proposed in [16] to the specificities of blockchain systems. In [16] the authors define a *Byzantine Altruistic Rational Tolerant* (BART) the protocol that guarantees the specified set of safety and liveness properties in the presence of all rational deviations A protocol is said *Incentive-Compatible Byzantine Fault Tolerant* (IC-BFT) if any rational user is incentivized to follow the prescribed protocol, also in presence of byzantine users. Then, to make practical the model proposed in [13] by relaxing the requirements in terms of t-immunity. Then, propose combined rules resulted from various theories (games theory, mechanisms theory) that will be encoded in incentive rules.

3 Formal abstractions for blockchains consistency

A large number of political, economical and social organisms invoke the possibility of blockchainize their activity. Obviously, the data that will be stored on the blockchain in each of these applications may have various levels of coherency: starting with very strong coherency for the case of banking or notary applications and finishing with weak coherency for applications such as IoTs. Identifying the exact requirements of coherency for representative applications in each class will be core of the current workpackage.

Studying the level of coherency provided by existing blockchains is related to the distributed shared register area. However, the similarity is moderated. A distributed register is a shared variable accessed by a set of processes through two operations, namely write() and read(). Informally, the write() operation updates the value stored in the shared variable while the read() obtains the value contained in the shared variable. The classical registers definitions [67] have been extended to the self-stabilizing area in [36]. This work considers that the system can be hit by arbitrary errors.

It should be noted that none of the above mentioned classical definitions captures the behavior of the popular blockchains such as Ethereum and Bitcoin. That is, values written in a classical register are potentially independent, and during the execution, the size of the register remains the same. In contrast, a new block cannot be written in the blockchain if it does not depend on the previous one, and successive writings in the blockchain increase its size. Also, differently from stabilizing registers, the prefix of the blockchain eventually converges, while no guarantees hold for the last created blocks.

The challenge here is to define new consistency abstractions that will capture the semantics of blockchains.

3.1 Defining new consistency abstractions for blockchains

The first effort in specifying the properties of permissionless blockchain systems is due to Garay and Kiayias [58]. They characterized Bitcoin blockchain via its quality and its common prefix properties, i.e., they define an invariant that this protocol has to satisfy in order to verify with high probability an eventually consistent prefix. This line of work has been continued by [78]. In order to model the behavior of distributed ledgers at runtime, Girault et al. [59] present an implementation of the Monotonic Prefix Consistency (MPC) criterion and showed that no criterion stronger than MPC can be implemented in a partition-prone message-passing system. On the other hand, the proposed formalization does not propose weaker consistency semantics more suitable for proof-of-work blockchains as BitCoin. In the same line of research, in [20], Anceaume et al. propose a new data type to formally model distributed ledgers and their behavior at runtime. They provide consistency criteria to capture the correct behavior of current blockchain proposals in a unified framework. In parallel and independently of [20], Anta et al [23] propose a formalization of distributed ledgers modeled as an ordered list of records. The authors propose three consistency criteria: eventual consistency, sequential consistency and linearizability.

Providing an unified framework able to capture the specificity of blockchain systems is still an open problem.

Moreover, formalizing the definition of this class of blockchain consistency will help in further proving the correctness and formally verifying algorithms that implement them. The semantic of the consistency can be express in terms of events and partial orders to these events. Note that for the classical consistency criteria the recent work of Gotsman et al. [61] provided a rich formalism based on token systems. However, this formalism should be extended to the blockchain context.

3.2 Design and formally prove new consistency algorithms tolerant to complex behaviors

It should be noted that existing effort for implementing coherency in blockchains (e.g. [58, 70, 29]) concentrate on solving the agreement (consensus) problem. However, it is already folklore that consensus is impossible to solve deterministically in asynchronous environments [55]. As pointed out in the state of the art section implementing blockchain probabilistic consensus in asynchronous environments subject to dynamic faults is still an open problem. The deterministic implementation of registers (even with strong consistency guarantees) in various models characterized by the presence of multiple types of faults (crashes, byzantine, dynamic byzantine or transient) have been investigated in the past [37, 36, 38, 41]. In blockchain systems, recent effort has been directed to both formalizing and implementing consistency criteria in systems prone to faults or Byzantine behaviors [20, 24, 47, 22].

None of the above proposed solutions work with the severe model of blockchain adversary including rationality , irrationality, churns or partitions. Therefore, the implementation of blockchain objects with various consistency guarantees in a *asynchronous environment* with *dynamic models of adversary* when the *size of the network is unknown* is a real challenge that might be mitigated by combining the framework in [41] with abstractions such as *k-quorums* defined in [15] and *sortition* techniques or intersecting sets (i.e. the secure version of the classical distributed quorum systems).

4 Develop correct-by-construction agreement algorithms for blockchains

The core of blockchains technologies is the agreement problem, studied in an environment where participants to the agreement may be controlled by a dynamic adversary. This form of agreement is known in distributed computing as Byzantine Agreement. Briefly stated, it requires that processors, some of which being potentially malicious, start the computation with an initial value and decide on the same value.

Byzantine Agreement, introduced by Lamport *et al.* [68], has been studied for decades in static distributed systems under different aspects (e.g., *possibility*, *complexity*, *cost*) in various models (from synchronous to asynchronous, from authenticated to anonymous) with different methodologies (deterministic or probabilistic).

4.1 Feasibility of blockchain agreement face to complex faults and behaviors

Garay et *al* [58] and [70] pioneered the study of Byzantine Agreement in blockchains. However, their studies are restricted to only round-based synchronous systems.

In [35] the authors study deterministic Byzantine Agreement in environments where the set of nodes controlled by the adversary may change over time. Contrary to other approaches, the model considers that a process previously affected by the adversary may send messages (based on a corrupted state), it will behave correctly in the way it sends those messages: i.e., send messages according to the algorithm. This behavior is very similar to the way the adversary acts in blockchains systems. Interestingly, in order to implement Byzantine Agreement under the assumption of dynamic Byzantine adversary a system needs at least $5f + 1$ nodes while in the case of static Byzantine adversary only $3f + 1$ are sufficient, where f is the number of nodes controlled by the Byzantine. These studies leaves a huge avenue to be explored. First, there is no extension of [35] to round-free environments. Second, in the same model of adversary there is no study related to feasibility of the agreement problem when the adversary movement is decoupled from the synchronous round of computation.

The above works do not implement agreement in asynchronous systems prone to dynamic adversary, rationality or churn.

An interesting challenge would be to explore the asynchronous probabilistic Byzantine agreement in systems prone to dynamic adversary and churn and where processes may have rational behaviors. One of possible solutions would be to considered the methodology proposed in [35] to round free churn exposed systems combined with *sortition* techniques and incentives rules issues from games and mechanisms theories.

4.2 New abstractions for blockchain agreement

Agreement in blockchains has an *Approximate agreement* flavor since the agreement on blockchains should be guaranteed not on an exact value. In systems hit by Mobile Byzantine Adversaries (the closes to the blockchain dynamic adversary) [39] formalized the approximate agreement and prove lower bounds on problem solvability in various dynamic adversary models and further propose an optimal algorithm for approximate agreement in round based systems. The lower bounds range from $n > 3f + 1$ to $n > 6f + 1$ depending on the type of adversary.

The previous results do not cover the blockchain agreement for several reasons: blockchains are not round-based, the adversary is not bounded to the rounds change, the agreement value is not a real value but a prefix of an ever changing blockchain.

Formalizing the bockchain approximate agreement and then proposing solutions in asynchronous environments hit by a dynamic adversary and rationality is the scientific lock here.

5 Develop correct-by-construction overlays and routing algorithms for blockchains

Blockchain underlying overlays and the associate routing are totally unexplored from theoretical point of view. However, the performances of blockchains technologies heavily depend on the performances of the underlying routing process. Recently, Lightning technologies imposed themselves as a viable direction for improving the blockchains throughput. This technology builds on top of blockchains (e.g. Bitcoin) an overlay of secured channels opened by two parties involved in long term multi-transactions. This overlay is further used to route transactions. Although blockchain technologies make strong assumptions on their underlying overlays there is no academic research that focus these overlays. The only prior research on the overlays topic has been developed in the context of dynamic networks such as P2P or wireless networks.

Another interesting point to be explored is the *liveness of the overlay* and more generally of the system. In blockchains the welfare of participants is a crucial factor. When participants desert the system in proof-of-work based blockchains the security of the system sinks which yields to the global sink of the system. This phenomenon is known in economy as the *tragedy of commons*.

5.1 New abstractions for blockchain overlays

Expenders theory proved recently its effectiveness for constructing overlays resilient to churn and partitions. The (node) expansion of an undirected graph is a characterization of the graph robustness. That is, graphs with good expansion are hard to be partitioned into a number of large connected components. In this sense, the expansion of a graph can be seen as a good evaluation of its resilience to faults and churn. However, the expansion of tree overlays is trivially $O(1/n)$. This weakness to faults explains why tree overlays are not pervasive in real applications.

In [65] the authors measured the robustness of tree overlay networks by evaluating their *graph expansion* and proposed a logarithmic algorithm for the construction of a constant degree self* expander that improve the resilience to churn of P2P tree-overlays.

The existing works are not tolerant to dynamic adversaries which can disconnect the overlay before its stabilization. *The unexplored yet research direction concerns the construction of constant degree expenders tolerant to dynamic Byzantine behavior and multiple types of faults. A possible solution would be to extend the methodology in [65] with* sortition techniques.

5.2 New formally verified routing protocols for blockchains

In order to increase the throughput in Bitcoin, the non academical research in blockchains proposed recently lightning routing networks [79]. Secured channels between two or more participants are opened on top of Bitcoin and transactions are routed on top of the virtual network formed by these channels. Routing in lightning networks has some similarities with routing in P2P or mobile wireless networks or delay tolerant networks. Flare, [**?**], for example the most prominent lightning routing was inspired by the wireless ZRP routing.

Interestingly, there is no formal academic research on this topic so far and our preliminary studies show that Flare (and its derivates) present severe limitations such as weak resilience to churn or deadlocks. Moreover, none of these lightning routing protocols has been exposed to multiple types of faults, attacks or dynamic adversaries.

The most studied overlay for routing in classical distributed systems and networking theory is the minimum spanning tree (MST). Research on spanning trees tolerant to multiple faults has been conducted in [31, 34, 32, 33]. None of the above cited algorithms is resilient to dynamic adversaries in conjunction with churn and attacks. The challenge here will be the design of new routing algorithms optimized for the context of lightning networks subject to multiple types of faults, attacks, rationality and dynamic adversaries.

6 Blockchains Interoperability

There are currently several operational systems for achieving interoperability between different blockchains such as Cosmos [3] or Polkadot [10]. These systems can be classified into two categories according to their decentralization level: systems that use a trusted third-party to validate transactions or systems that realize it directly between blockchains without the need of a trusted third-party.

In order to execute an exchange or a *swap* (i.e., a set of transactions between parties), transacting agents (i.e., blockchain users) are provided with a protocol to stick to. A protocol in this case consists of a specific sequence of instructions agents should perform to preserve the ACID properties [69] of the individual transactions or exchanges.

The first atomic swap solution has been proposed for Bitcoin by *Nolan* [73] making use of hash-time locked contracts enabling conditional assets transfers. *Decred* [4] implements Nolan's algorithm on UTXO-based premissionless. In [63] the authors generalize and prove correct *Nolan*'s scheme. Other projects such as *BartherDEX* [7], part of the Komodo project [6], represents a cross-chain solution that matches orders and defines the swap protocol or *Blockchain.io* [2] implements atomic cross-chain swaps by combining centralized components (order matching) with decentralized ones (trade settlement and execution). These projects are not yet formally proved correct.

The academic research focuses on *hybrid* swap protocols, replacing decentralized commitment/locking schemes (hash-locks) with centralized ones, resulting more attractive and efficient. *AC3TW* and *AC3WN* [86] protocols propose atomic cross-chain swaps respectively with centralized and distributed trusted authorities (i.e. witnesses) It should be noted that different swap protocols differ essentially in the involved parties. The set of swap participants can be composed only of the asset owners (e.g., as in [63]) or by owners accompanied by a trusted third party (e.g., as in the AC3TW protocol [86]).

In [28] the authors propose a generic game theoretical framework that formalizes the swap problem and characterize equilibria of two representative recent protocols presented in [73] and [86] respectively. In the case of the protocol proposed in [73] and generalised in [63], following the protocol is the unique subgame perfect equilibrium (in dominant strategies), while in the case of the protocol proposed in [86], following the protocol is a Nash equilibrium.

This work opens several research directions. Swap protocols and more generally blockchain intercommunication are not yet properly formalized and analyzed.

An important challenge in this area is to fully formalize the problem and analyze the robustness of protocols that implement it to both rational and irrational behaviors, dynamic adversaries and attacks and coalitions.

7 Conclusions

Blockchains evolve in a very complex environment. Differently, from the classical distributed systems where faults are considered to appear in isolation and to affect the same node of the system during the whole computation, in blockchains environments faults do not follow the same pattern. Blockchains have to face in the same time classical pattens of faults such as crash faults, transient faults, Byzantine faults but also attacks, dynamic faults, churn and selfish or rational/irrational behaviors. Therefore, before addressing the algorithmic core of blockchains a fully characterization of the adversarial environment is necessary. Interestingly, faults and errors in most of the cases (even in classical distributed system) have only a verbose definition. When systems have to be resealed for an industrial or critical economical use automatic verifications and mathematical proofs are necessary. Therefore, verbose definitions are not precise enough. In this paper we discuss five important challenges in this area. The first challenge will be to explore and formalize blockchains robustness. The second challenge will be to formally define universal abstractions for characterizing blockchains consistency. The third challenge will be to provide new correct-by-construction abstractions for agreement in blockchains. The effectiveness of these building blocks will be insured by a formal verification and proof using formal methods tools. The fourth challenge is to develop optimized overlays and communication primitives for blockchains resilient to nodes churn, various attacks and adversary dynamic behaviors and target to avoid the partition or the sink of the system in a *tragedy of commons*. Finally, the formalization of blockchains interoperability is the fifth challenge.

References

1. Blockchain.io (your gateway to the internet of value). accessed on January 10, 2020. https://blockchain.io/.
2. Cosmos: A Network of Distributed Ledgers. accessed on January 10, 2020. https://cosmos.network/cosmos-whitepaper.pdf.
3. Decred cross-chain atomic swapping. accessed on January 10, 2020. https://github.com/decred/atomicswap.
4. Komodo (advanced blockchain technology, focused on freedom). accessed on January 10, 2020. https://docs.komodoplatform.com/whitepaper/introduction.html.
5. Komodo barterdex. accessed on January 10, 2020. https://github.com/KomodoPlatform/BarterDEX.
6. Polkadot: Vision for a Heterogeneous Multi-Chain Framework. accessed on January 10, 2020. https://polkadot.network/PolkaDotPaper.pdf.
7. I. Abraham, L. Alvisi, and J. Halpern. Distributed computing meets game theory: Combining insights from two fields. *SIGACT News*, 42:69–76, 06 2011.
8. I. Abraham, D. Dolev, R. Gonen, and J. Halpern. Distributed computing meets game theory: Robust mechanisms for rational secret sharing and multiparty computation. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*, PODC ?06, page 53?62, New York, NY, USA, 2006. Association for Computing Machinery.
9. I. Abraham, G. Gueta, and D. Malkhi. Hot-stuff the linear, optimal-resilience, one-message BFT devil. *CoRR*, abs/1803.05069, 2018.
10. A. S. Aiyer, L. Alvisi, and R. A. Bazzi. Byzantine and multi-writer k-quorums. In *Distributed Computing, 20th International Symposium, DISC 2006, Stockholm, Sweden, September 18-20, 2006, Proceedings*, pages 443–458, 2006.
11. A. S. Aiyer, L. Alvisi, A. Clement, M. Dahlin, J.-P. Martin, and C. Porth. Bar fault tolerance for cooperative services. In *SOSP '05*, 2005.
12. Y. Amoussou-Guenou, B. Biais, M. Potop-Butucaru, and S. Tucci Piergiovanni. Rationals vs byzantines in consensus-based blockchains. *to appear AAMAS 2020*, abs/1902.07895, 2019.
13. E. Anceaume, M. Gradinariu, and A. Ravoaja. Incentives for P2P fair resource sharing. In *Fifth IEEE International Conference on Peer-to-Peer Computing (P2P 2005), 31 August - 2 September 2005, Konstanz, Germany*, pages 253–260, 2005.
14. E. Anceaume, A. D. Pozzo, R. Ludinard, M. Potop-Butucaru, and S. Tucci Piergiovanni. Blockchain abstract data type. In *Proceedings of the 31st ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, 2019.
15. E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. D. Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolic, S. W. Cocco, and J. Yellick. Hyperledger fabric: a distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference, EuroSys 2018, Porto, Portugal, April 23-26, 2018*, pages 30:1–30:15, 2018.
16. A. F. Anta, C. Georgiou, and N. C. Nicolaou. Atomic appends: Selling cars and coordinating armies with multiple distributed ledgers. *CoRR*, abs/1812.08446, 2018.
17. A. F. Anta, K. Konwar, C. Georgiou, and N. Nicolaou. Formalizing and implementing distributed ledger objects. *ACM SIGACT News*, 49(2):58–76, 2018.
18. A. F. Anta, K. M. Konwar, C. Georgiou, and N. C. Nicolaou. Formalizing and implementing distributed ledger objects. *SIGACT News*, 49(2):58–76, 2018.

19. H. Attiya and J. Welch. *Distributed Computing: Fundamentals, Simulations and Advanced Topics (2nd edition)*. John Wiley Interscience, March 2004.
20. A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. on dependable and secure computing*, 1(1), 2014.
21. N. Banu, S. Souissi, T. Izumi, and K. Wada. An improved byzantine agreement algorithm for synchronous systems with mobile faults. *International Journal of Computer Applications*, 43(22):1–7, April 2012.
22. M. Belotti, S. Moretti, M. Potop-Butucaru, and S. Secci. Game theoretical analysis of Atomic Cross-Chain Swaps. In *40th IEEE International Conference on Distributed Computing Systems (ICDCS)*, Singapore, Singapore, Dec. 2020.
23. I. Bentov, R. Pass, and E. Shi. The sleepy model of consensus. *IACR Cryptology ePrint Archive*, 2016:918, 2016.
24. B. Biais, C. Bisière, M. Bouvard, and C. Casamatta. The blockchain folk theorem. *The Review of Financial Studies*, 2019.
25. L. Blin, S. Dolev, M. G. Potop-Butucaru, and S. Rovedakis. Fast self-stabilizing minimum spanning tree construction - using compact nearest common ancestor labeling scheme. In *Distributed Computing, 24th International Symposium, DISC 2010, Cambridge, MA, USA, September 13-15, 2010. Proceedings*, pages 480–494, 2010.
26. L. Blin, M. Potop-Butucaru, and S. Rovedakis. A super-stabilizing log(n)log(n)-approximation algorithm for dynamic steiner trees. *Theor. Comput. Sci.*, 500:90–112, 2013.
27. L. Blin, M. Potop-Butucaru, S. Rovedakis, and S. Tixeuil. A new self-stabilizing minimum spanning tree construction with loop-free property. *Comput. J.*, 59(2):225–243, 2016.
28. L. Blin, M. G. Potop-Butucaru, and S. Rovedakis. Self-stabilizing minimum degree spanning tree within one from the optimal degree. *J. Parallel Distrib. Comput.*, 71(3):438–449, 2011.
29. F. Bonnet, X. Défago, T. D. Nguyen, and M. Potop-Butucaru. Tight bound on mobile byzantine agreement. *Theor. Comput. Sci.*, 609:361–373, 2016.
30. S. Bonomi, S. Dolev, M. Potop-Butucaru, and M. Raynal. Stabilizing server-based storage in byzantine asynchronous message-passing systems: Extended abstract. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, Donostia-San Sebastián, Spain, July 21 - 23, 2015*, pages 471–479, 2015.
31. S. Bonomi, M. Potop-Butucaru, and S. Tixeuil. Stabilizing byzantine-fault tolerant storage. In *2015 IEEE International Parallel and Distributed Processing Symposium, IPDPS 2015, Hyderabad, India, May 25-29, 2015*, pages 894–903, 2015.
32. S. Bonomi, A. D. Pozzo, and M. Potop-Butucaru. Tight self-stabilizing mobile byzantine-tolerant atomic register. In *Proceedings of the 17th International Conference on Distributed Computing and Networking, Singapore, January 4-7, 2016 and to appear in TCS 2017*, pages 6:1–6:10, 2016.
33. S. Bonomi, A. D. Pozzo, M. Potop-Butucaru, and S. Tixeuil. Approximate agreement under mobile byzantine faults. In *36th IEEE International Conference on Distributed Computing Systems, ICDCS 2016, Nara, Japan, June 27-30, 2016*, pages 727–728, 2016.
34. S. Bonomi, A. D. Pozzo, M. Potop-Butucaru, and S. Tixeuil. Optimal mobile byzantine fault tolerant distributed storage: Extended abstract. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing, PODC 2016, Chicago, IL, USA, July 25-28, 2016*, pages 269–278, 2016.

35. S. Bonomi, A. D. Pozzo, M. Potop-Butucaru, and S. Tixeuil. Self-stabilizing mobile byzantine-tolerant regular register with bounded timestamp. *SRDS 2017*, abs/1609.02694, 2016.
36. E. Buchman, J. Kwon, and Z. Milosevic. The latest gossip on bft consensus. *arXiv preprint arXiv:1807.04938*, 2018.
37. H. Buhrman, J. A. Garay, and J.-H. Hoepman. Optimal resiliency against mobile faults. In *Proceedings of the 25th International Symposium on Fault-Tolerant Computing (FTCS'95)*, pages 83–88, 1995.
38. M. Castro and B. Liskov. Practical byzantine fault tolerance and proactive recovery. *ACM Trans. Comput. Syst.*, 20(4):398–461, Nov. 2002.
39. J. Chen and S. Micali. Algorand. *arXiv preprint arXiv:1607.01341*, 2017.
40. V. Cholvi, A. F. Anta, C. Georgiou, and N. C. Nicolaou. Brief announcement: Implementing byzantine tolerant distributed ledger objects. In J. Suomela, editor, *33rd International Symposium on Distributed Computing, DISC 2019, October 14-18, 2019, Budapest, Hungary*, volume 146 of *LIPIcs*, pages 40:1–40:3. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.
41. T. Crain, V. Gramoli, M. Larrea, and M. Raynal. Dbft: Efficient byzantine consensus with a weak coordinator and its application to consortium blockchains. *arXiv preprint arXiv:1702.03068*, 2017.
42. T. Crain, V. Gramoli, M. Larrea, and M. Raynal. (Leader/Randomization/Signature)-free Byzantine Consensus for Consortium Blockchains. http://csrg.redbellyblockchain.io/doc/ConsensusRedBellyBlockchain.pdf (visited on 2018-05-22), 2017.
43. K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. E. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, D. Song, and R. Wattenhofer. On scaling decentralized blockchains - (A position paper). In *Financial Cryptography and Data Security - FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers*, pages 106–125, 2016.
44. Daian, R. Pass, and E. Shi. Snow white: Provably secure proofs of stake. *IACR Cryptology ePrint Archive*, 2016:919, 2016.
45. C. Decker, J. Seidel, and R. Wattenhofer. Bitcoin Meets Strong Consistency. In *Proceedings of the 17th International Conference on Distributed Computing and Networking Conference (ICDCN)*, 2016.
46. I. Eyal, A. E. Gencer, E. G. Sirer, and R. van Renesse. Bitcoin-ng: A scalable blockchain protocol. In *13th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2016, Santa Clara, CA, USA, March 16-18, 2016*, pages 45–59, 2016.
47. I. Eyal and E. G. Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *Financial Cryptography and Data Security - 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers*, pages 436–454, 2014.
48. M. J. Fischer, N. A. Lynch, and M. S. Paterson. Impossibility of distributed consensus with one faulty process. *Journal of the ACM (JACM)*, 32(2):374–382, 1985.
49. J. A. Garay. Reaching (and maintaining) agreement in the presence of mobile faults. In *Proceedings of the 8th International Workshop on Distributed Algorithms*, volume 857, pages 253–264, 1994.
50. J. A. Garay and A. Kiayias. Sok: A consensus taxonomy in the blockchain era. *IACR Cryptol. ePrint Arch.*, 2018:754, 2018.
51. J. A. Garay, A. Kiayias, and N. Leonardos. The bitcoin backbone protocol: Analysis and applications. In *Proc. of the EUROCRYPT International Conference*, 2015.

52. A. Girault, G. Gössler, R. Guerraoui, J. Hamza, and D.-A. Seredinschi. Monotonic prefix consistency in distributed systems. In *International Conference on Formal Techniques for Distributed Objects, Components, and Systems*, pages 41–57, Berlin, Germany, 2018. Springer.
53. G. Golan-Gueta, I. Abraham, S. Grossman, D. Malkhi, B. Pinkas, M. K. Reiter, D. Seredinschi, O. Tamir, and A. Tomescu. SBFT: a scalable decentralized trust infrastructure for blockchains. *CoRR*, abs/1804.01626, 2018.
54. A. Gotsman, H. Yang, C. Ferreira, M. Najafzadeh, and M. Shapiro. 'cause i'm strong enough: reasoning about consistency choices in distributed systems. In *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20 - 22, 2016*, pages 371–384, 2016.
55. M. Herlihy. Atomic cross-chain swaps. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, pages 245–254. ACM, 2018.
56. C. Hess and E. Ostrom. Understanding knowledge as a commons. *From theory to Practice*, 2007.
57. T. Izumi, M. Potop-Butucaru, and M. Valero. When expanders help self-healing distributed r-tree overlays. In *IEEE 12th International Symposium on Parallel and Distributed Computing, ISPDC 2013, Bucharest, Romania, June 27-30, 2013*, pages 143–150, 2013.
58. E. Kokoris-Kogias, P. Jovanovic, N. Gailly, I. Khoffi, L. Gasser, and B. Ford. Enhancing Bitcoin Security and Performance with Strong Consistency via Collective Signing. In *Proceedings of the 25th USENIX Security Symposium*, 2016.
59. L. Lamport. On inter-process communications, part I: basic formalism and part II: algorithms. *Distributed Computing*, 1(2):77–101, 1986.
60. L. Lamport, R. Shostak, and M. Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, July 1982.
61. B. A. Lewis, P.M. and M. Kifer. *Databases and transaction processing: an application-oriented approach.* Addison-wesley Reading, 2002.
62. S. Micali. Algorand: The efficient and democratic ledger. *arXiv preprint arXiv:1607.01341*, 2016.
63. S. Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf, 2008.
64. T. Nolan. Re: Alt chains and atomic transfers. accessed on January 10, 2020. https://bitcointalk.org/index.php?topic=193281.msg2224949#msg2224949.
65. E. Ostrom. *Governing the commons.* Cambridge university press, 2015.
66. E. Ostrom and J. Walker. *Trust and reciprocity: Interdisciplinary lessons for experimental research.* Russell Sage Foundation, 2003.
67. R. Ostrovsky and M. Yung. How to withstand mobile virus attacks (extended abstract). In *Proceedings of the 10th Annual ACM Symposium on Principles of Distributed Computing (PODC'91)*, pages 51–59, 1991.
68. R. Pass, L. Seeman, and A. Shelat. Analysis of the blockchain protocol in asynchronous networks. In *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II*, pages 643–673, 2017.
69. R. Pass and E. Shi. Fruitchains: A fair blockchain. In *Proceedings of the ACM Symposium on Principles of Distributed Computing, PODC 2017, Washington, DC, USA, July 25-27, 2017*, pages 315–324, 2017.
70. J. Poon and T. Dryja. The bitcoin lightning network. https://lightning.network/lightning-network-paper.pdf, 2016.

71. M. Potop-Butucaru. Brace: Blockchains and the commons. submitted to ERC Advanced program, 2017, Proposal ID : 788886 (internal reference number: SEP-210446727) Call : ERC-2017-ADG Type of action : ERC-ADG Topic : ERC-2017-ADG http://pagesperso.lip6.fr/Maria.Gradinariu/spip.php?article23.
72. R. Reischuk. A new solution for the byzantine generals problem. *Information and Control*, 64(1-3):23–42, January-March 1985.
73. F. Saleh. Blockchain Without Waste: Proof-of-Stake. SSRN Scholarly Paper ID 3183935, Social Science Research Network, Rochester, NY, Jan. 2019.
74. T. Sasaki, Y. Yamauchi, S. Kijima, and M. Yamashita. Mobile byzantine agreement on arbitrary network. In *Proceedings of the 17th International Conference on Principles of Distributed Systems (OPODIS'13)*, pages 236–250. Springer, December 2013.
75. G. Wood. Ethereum: A secure decentralised generalised transaction ledger. http://gavwood.com/Paper.pdf (visited on 2018-05-22).
76. V. Zakhary, D. Agrawal, and A. Abbadi. Atomic commitment across blockchains. *Proceedings of the VLDB Endowment*, 2020.