Time, Consensus and Governance by Design for Blockchain and DLT

Stéphane Caporali,
Caporali Conseil

December 24, 2020

Abstract

For blockchain governance, data is an asset, and the time parameter is strategic and sensitive. This document aims to explore this issue for blockchain. It analyses the issues related to two properties associated with the parameter time : the ability to measure time (timestamp), and the guarantee of the ordering between events and proposes an innovative approach promoting blockchain technology as an infinite state machine. It is illustrated by the concept of simultaneous states to take into account uncertainty as a transition between two states. After recalling the mechanisms of the consensus algorithm in the case of the PoW, and analyzing the associated issues, this document proposes two data governance functions that can be derived from the notion of time between two blocks : the alert, and the control loop. This paper proposes the abstract concept of temporal structure, understood as the four functions derived from time parameter in support of the data governance strategy for blockchain in an industrial context, and governance by design the resulting approach.

Keywords— Time, blockchain, consensus algorithm, state machine, governance.

1 Introduction

The starting point for the creation of this document has been, since the introduction of the proof of work (PoW) in the Satoshi Nakamoto's founding document [1], the observation of the plurality of different consensus mechanisms in blockchain. There is a risk of fragmentation due to the absence of a technical framework that unifies the whole. Little by little, the idea was born that a level of governance : a governance of the consensus mechanism, is able to bring together all the blockchains despite the differences in the consensus mechanisms. The idea is therefore not that a governance of the consensus mechanism can exist, but that this governance must exist because we need it. In the implementation phase of a blockchain project, a classification of algorithms is a tool that can help support governance. In termination phase, there might be a strategy to terminate a blockchain project, for example the creation of a fork. In the operational phase, finding an approach to governance of the consensus mechanisms is not easy, but something has to be found. This document is an attempt to answer to these questions and make the subject grow in maturity.

2 Time and governance by data

The governance principles must improve the use of data in a more efficient way, for the different operations involved in a system. The operation requires the coordination of the different participating ecosystems in a distributed approach. Time parameter is a sensitive and strategic data. Specifics methods and concepts must be organized around this parameter in order to allow a good governance.

3 Function of timestamp

The representation of time is standardized by ISO (2019)[2] where time is defined as a mark attributed to an instant or a time interval. By extension, associating a measure to a mark does not seem to pose a problem. However, in distributed environment, without a center, problems arise such as for example the choice of the position of the clock. On this subject, Ladleif and Weske (2020) [3] explain that an industrial process needs to dialogue with the blockchain. This authors give the example of a software package that manages invoicing, which needs to read information or ask questions in the blockchain via an oracle. However, there is no natural synchronization between the internal time in the blockchain and the external time. A service that updates a distributed ledger using data from outside of the distributed ledger system called oracle [4] is necessary to allow the transition between off-ledger and on-ledger requests because, according to the authors, blockchain is a closed-world environment without access to global timing information. Thus the measure of time poses important questions described in the document of Jan Ladleif and Mathias Weske beyond the scope of this document and especially the absence of a natural notion of measuring time. However, time measurement is a function associated with time.

4 Function of trust

4.1 Two abstracts point of view of time

Lamport (1978) [5] writes that there are two points of view of time, and gives the example of special relativity where we use a space-time diagram with coordinate diagram where the position of time t is located on the y-axis rather than on the x-axis. This point of view allows to consider the parameter t as a simple label for an event. This suggests that there may be different abstract views of time. As an illustration, imagine that Event is the value associated with an event: for example a consensus has been made and that a value that we call Event is associated with this consensus. Figure 1 is the usual representation where a measure of the event is under the prerequisite of

Figure 1: Case $F(time) = Event$

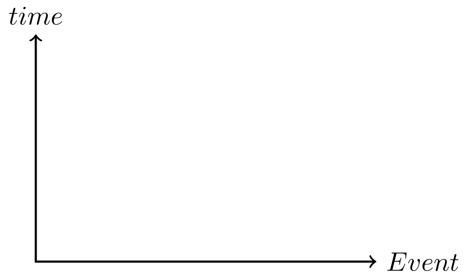

Figure 2: Case $F(Event) = time$

the measure of time. In reverse, in figure 2, t is located on the y-axis rather than on the x-axis. In this second point of view, the measure of the chronology is just a label in the properties of the event.

4.2 Blockchain participants as an ecosystem of participants in finite numbers, with an infinity of possible transactions

Characteristics of a participant In this document, we are interested in blockchain participants. For example a miner is a blockchain participant. The participants are not only constituted by the miners but others entities participating in blockchain operations. A participant can be a machine with computational capabilities : for example a smart contract or a script, but it is not necessary and it can be a human being. The question of knowing which entities constitute the participants is open in this document, in particular that can concern on-ledger but also off-ledger entities.

Choice of participants in the blockchain rather than the blockchain as a system for modeling the state machine

Strategic importance of the community of participants Shorish (2018) [6] asks the question of what we mean by strategic in a blockchain, and considers by strategic the consideration by blockchain participants of the actions of other participants. A blockchain is a decentralized ledger, constituted of different parts : a decentralized network, a consensus algorithm, the cryptographic tools especially for the chain structure. For this reason, blockchain technology is the combination of different techniques. For specific functions, describe later in this document, the participants could in addition be described as a state machine. This means that if a participant is a computer which is a turing machine, it could be associated with both an infinite state machine for the same computer system. Blockchain can be seen as the community of participants, with the participants being responsible for the operation of the blockchain and associated with the role of the governance or simply users. Behind one participant there is an entity : for example a system, a process, a device or a human. This link between the participants is a form of trust. However, a technical prerequisite is that participants are entities synchronized with the same ”clock”, as mentioned in section 3.

An event-centric approach Faced with the problem of the infinity of possibles values in the time scale, an idea is to introduce a number of agents which is finite : the description of the system will focus on the interactions between agents at a precise moment. The future and the past are in this case projections from the present.

This is just an approach to how things can be implemented. The interactions between agents are in theory in number infinite but finite in a practical way. This approach conforms to the second case of section 4.1.

4.3 Presentation of the computational model of the infinite state machine

General characteristics of a state machine By computation I mean the act or process of calculating an answer or amount by using a machine [7]. By consequent it is not necessary an amount, it can be an answer. The theory of computation is the basis of computing theory. A state machine is a model of computation, characterized by the fact that a state is modified by an event. For example, an elevator, where pressing the button of a given floor starts moving the elevator to that floor.A state machine, in its simplest form, is without memory (memoryless) : the last event (for example the previous movement of the elevator) is not stored : it is not not because the precedent user wanted to go to the second floor, that the next user has more chance of choosing the second floor, in principle. Keeping the previous operation in memory is useless. The state machine is reactive to the event.

Association between blockchain and state machine : a counter-intuitive choice ? The association of blockchain with the state machine seems a paradox, because computational model of the state machine is memoryless, but on the contrary blockchain has the property of immutability that ISO (2020) [8] indicates as the property wherein ledger records cannot be modified or removed once added. However, blockchain is, in a way, in its description, in accordance with the fundamental properties of the state machine. Blockchain is often described as a particular database, with a write mode, or a read mode but not a modify mode : once written in the chain, a value cannot be deleted. Blockchain is a particular, distributed register, in the sense of accounting where an accounting register is used. However, at least in the case of the proof of work, described later in the document, blockchain is, as the state machine, memoryless : the fact for a minor to be the entity that won the previous challenge does not give more chance to be the next winner [9].

Application to blockchain Brooks (2017) [10] in his blog considers blockchains to be, fundamentally, systems for managing valid state transitions. Petri net is a mode of representation well adapted to distributed environment in the case of discrete events system. These are are the following notation used with Petri net :

- circle is related to an event.
- rectangle is related to a state.

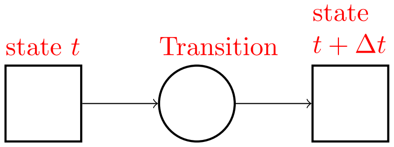

Figure 3: The state machine computational model can be associated with each blockchain participant

An event represents a certain situation occurring at a given point in time. Note that in the context of industrial control system, an event is defined as a change of state or value detected for processing and/or reporting [11].

4.4 The muti-state machine

Problem of uncertainty in the time interval between two blocks It is called a multi-state machine, a state machine whose records can be in multiple states at the same time, which seems contradictory to the work of a deterministic state machine. However we can adopt this representation as an abstract model. We assume at this stage that the time interval between two blocks is constant. For example, 10 minutes for bitcoin. Shorish (2018) [6] explains that if a state machine representation is possible, it must handle the associated uncertainty. The uncertainty could be related to the difficulty of predicting an accurate value of the time interval between two blocks but we agree that we consider the value constant at this stage of the document even though we will go further in paragraph 5 about this question. But even with a precise value, we can consider that the transition is characterized by its inherent uncertainty : in fact the transition is the not yet something and already not something else. To illustrate this fact, I propose that a state machine can have two states at the same time. The originality is that we assume that these states are contradictory : it is only a representation to illustrate the state of incertitude.

Representation of the uncertainty The state/event table, the state diagram, and usual mathematical models used with state machine representation are adapted to finite state machine. In the case of an infinite state machine, form of representations are to be developed. We can however explain what could represent the state of the machine. One event is no more, and another is not yet : We will say that the previous event is no more and that the next event is not yet at the same time: these is a state of uncertainty . A state is only a transition between two events. This conception that highlights the event is in agreement with the second view of time presented in figure 2. There are two options for representing a state :

- First option : two separate events at the same time
- Second option : one event with two values at the same time

4.5 The ordering between events

Definition ISO (2019) [2] defines that a unit of time is the time interval part of the time axis limited by two instants. The ability to generate a unit of time can be described by the function :

$$f(t) = \begin{cases} t+1 & \text{if } t > 0 \\ 0 & \text{if } t = 0 \end{cases}$$

as a minimal recursive function.

Comparison between blockchain and clock ISO (2019) [2] specifies that a clock is a time scale suited for intra-day time measurements. Trubetskoy (2018) [9] in his blog develops the idea of a blockchain comparable to a clock, which is imprecise but generates trust, and in a way plays a role in the community of participants because it generates trust. The author uses the word universal clock, and compares the state of the chain to the ticks of a clock. This analogy with a tick suggests that a clock prints a rhythm, before the measurement of the time parameter. Pérez-Marco (2018) [12] expresses the same idea, that of a universal clock, which has the property of being untamperable and unfalsifiable. He also uses the term tick and identifies the timing of the tick when a new block is validated, which in this case could be compared to the event that precedes a new state. This makes it possible to see more clearly in the modeling of the state machine, and the differentiation between the event (the consensus) and the state (a new unit of time, a new tick) Swann (2015) [13] compares the time between two blocks to a feature. This point is important, because the arises of knowing how to define the time between two blocks, is it a property, a function ? The author explains that the time between two blocks is a privilege function in order to specify a model based on event conditions and states change. This approach is in accordance with the approach developed in this document.

Duality discrete/continuous time In another work, Swann (2016) develops the idea that time can be both discrete and continuous [14]. First of all, in this document we consider time not as a phenomenon, but as a parameter, because the objective is to describe a digital system. A machine does not perceive time in the sense that it has a sense of time, that of a flow. However a machine considers the parameter t as a mark on a scale. Despite this, the parameter time can thus be differentiated from two points of view, depending on whether a given event is considered in a flow of time, or whether the measurement is associated with the event. The second case is linked to the discrete conception of time, that is to say to case number two described in section 4.1.It has an affinity with the model of the state machine described in this document. We note that the conception of a discrete time is not incompatible with the ISO definition (2019) [2] which states that time is defined as a mark attributed to an instant or a *time interval.*

The generation of trust Bonnet, et al. [15] consider that the distributed ledger is a data structure. This approach is interesting, because it leads to the concept of data approach, described in the introduction. Data is not just the content of the transaction. The expression time instant is used to qualify a new unit of time. The state transition function is described as the result of the current state, the events, and the network containing all the nodes. This emphasizes the idea that blockchain can be described from the behavior of all the participants. In summary, blockchain has the property of a clock. This implies, by extension, the ability to measure time on a time scale. The problem developed in section 1 consists in the difficulty of developing a governance of consensus algorithms during the operational phase of a blockchain project. We will see in the next section how the state machine model can be the basis for the development of a technical and terminological framework that supports governance functions of alert and control.

5 Function of alert

5.1 The infinite probabilistic state machine

The introduction of the consensus The last section provides a first level of interpretation with the property of ordering between events, illustrated by the model of the state machine. In his work, Shorish (2018) [6] presents the model of the state machine for blockchain and reinforces the idea that formalizing the interpretation of the state machine is a challenge. To go further in this interpretation, we can ask what is the nature of the event that precedes a new state. This cannot be resumed by the generation of a unit of time. But what could this formalization be ? The idea in this section is that the association with state machine (without memory) and blockchain is not obvious, but in fact the states of a machine could be compared to the unit of time, and the event (transition) to the *consensus.*

- circle is related to an event condition. The condition is that the consensus has been made.
- square is related to a state change

Figure 4 show the case of a state machine that is dedicated to manage the state transition : the event is constituted by the consensus.

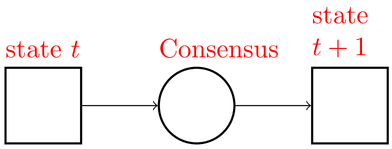

Figure 4: The consensus is the condition of the event

The probabilist component of the state machine Shorish (2018) [6] considers that a transition function between the states of a blockchain is probabilistic by nature. This point is important and shows that the process of consensus introduces a new component into the state machine : its probabilistic component. However, not all the consensus algorithms are probabilistic by nature. But this property illustrates a function which is the alert function. To better understand, the following paragraph recalls the important principles of the proof of work, which is the consensus algorithm taken as an example in this document.

5.2 The consensus algorithm

Clarifications concerning the notion of consensus With blockchain, the consensus algorithm is often confused with other forms of validations, as it is a specific form of the consensus mechanism. For example with smart contracts, there are verification mechanisms, which involves forms of consensus. To see more clearly, we can wonder what precisely is the consensus for blockchain technology. ISO (2020) [8] defines consensus as the agreement between DLT nodes that a transaction is validated and that the distributed ledger contains a consistent set and ordering of validated transactions. It is the function of ordering that is discussed in this document through the example of the proof of work.

The Byzantine Generals' Problem Nakamoto (2008) [1] in its founding document introduces the proof of work (PoW) as a consensus algorithm. To fully understand Proof of work, it is necessary to understand the Byzantine Generals problem. The problem known as man-in-the-middle attack in network security and theorized with the Two Generals Problem argues that it has been proved unsolvable theoretically that a channel between two points can be trusted. A pragmatic approach is to accept the uncertainty of the communication channel. Generalization with more than two generals is the Byzantine Generals' Problem. This is the principle : the Byzantine army must attack a city at a given time. The generals must communicate to agree on what time to attack the city, messengers can be intercepted and the message tampered. The problem is how to successfully trust the channel and obtain a consensus on the precise date when the Byzantine army will attack the city.

The proof of work In the case of proof of work is introduced a nonce which is a arbitrary number from which if we take the hash of the nonce and the message, we get a hash that starts with five zeros. You do this by guessing a large number of tries. This imply to increase the value of the nonce with each attempt. There is no way to find the nonce efficiency, the only way is to guess (the process is memoryless). This is what we call proof of work. The hard part is finding the nonce, but to hash the string is very easy. If the message is tampered and not the nonce, the resulting hash will not start with five zero. Modifying the message means calculating a new nonce. A series of messages is combined into a block and a high number of zero associated with the nonce. If a message is intercepted, it is considered *practicably* impossible to find the nonce, since all the computing power of all the Byzantine army represented an important amount of calculation. There might be a traitor in the group of generals who can give false time attack information. There are a large number of generals. In the case of proof of work, each general must solve a problem to send the message. If a general is the traitor, he has a weak probability to solve the messages twice consecutively. This description is based on the Ivan Liljeqvist youtube channel[16] which clearly describes the concept. Nakamoto (2008) [17] says in a mail written to the Cryptography Mailing List that the proof of work is a probabilistic response to the Byzantine Generals' Problem. Main innovation brought by the blockchain, the proof of work makes it possible to validate a transaction in a distributed environment by solving a cryptographic puzzle. The first of the miners to solve the puzzle wins the challenge. But it is related both to the hash power and to pure luck. The proof of work is a major innovation of blockchain because it introduces *chance*, structurally in the process of determination of the consensus. Previously, work on consensus algorithms in a distributed environment trying to solve the Byzantine generals problem was generally oriented towards a deterministic approach. With proof of work, once the puzzle has been solved, the block is broadcast in the network and verified in order to validate it. In case of conflict, the longest chain wins : it is the rule of the longest chain. Many others forms of consensus are developed, for example proof of stake. But the object of the consensus algorithm remains to validate the ordering between two transactions. It is a function of notarization.

5.3 Time between two blocks

Genesis of the time interval between two blocks In its founding document, Nakamoto (2008) [1] says that ten minutes is the reason of blocktime and that it is for storage reasons : the author estimates, on the one hand, the quantity of memory per year needed for the storage of the chain, and the storage capabilities of computer systems on the other hand, the duration of the block time of 10 minutes being only a reasonable value according to the constraints related to the storage. We can wonder if it is no a part of arbitrary reason in the choice of this parameters. It is a compromise between different things. Nakamoto talks about time between the generation of two blocks, and not only the time of mining. Another interest is that this time is enough longer to take time to recreate a chain in the case of a fork. This is particularly true if the chain is long and this is a protection against a fifty-one percent attack. For this reason a long chain is more secure. Another choice is made for example by the blockchain Ethereum. Siriwardena (2017) [18] explains that the primary motivation of the blockchain Ethereum was to diminish the value of this parameter, in order to make faster the time between two transactions. It is true that for bitcoin and the proof of work, the important time of ten minutes slows down the speed of the transaction and imply scalability issues. When designing a blockchain, there is the choice of the type of consensus algorithm, and another important part is the choice of the value of a time between the generation of two blocks. This shows how important this parameter is.

Terminology We may wonder how to name *the time between two consecutive blocks*. It is important to emphasize that we find different words for same or similar meaning.

Blocktime Swann and De Filippi (2017) [19] use the term blocktime and give the description of blocktime as the time over which a certain number of blocks will have confirmed and consider it as a notion of time specific to the blockchain.

Block produce time On their side, Zhang and Ma (2020) [20] use the expression : "block produce time".

Average Seconds Between Blocks Küfeoğlu Özkuran (2019 [21] differentiate according to the unity of time used and used the terms Average Seconds Between Blocks (ASBB) and Average Hours Between Blocks (AHBB).

Mining time Siriwardena (2017) [18] uses the expression block time and says that block time defines the time it takes to mine a block. Consequently, the meaning is that of mining time.

Block time Ladleif and Weske (2020) [3] specify that the time between two blocks and mining time are two different things and use the word block time for the time between two blocks. The authors integrate into the time interval between two blocks a delay between the initial creation of the transaction and the work of mining through the interesting analysis of the timeline along the blockchain-based execution of a business process.

Expected and average block time Siriwardena (2017) [18] says that there is an expected block time, and an average block time. The differences between the two concepts is important. The expected block time is the value of block time in theory, for a given blockchain, for example ten minutes for bitcoin. The average block time, according to the author, is what we find after a small number of block generated. And this is different.

Cycle time Cycle time is the standardized definition given by ISO (2004) [11] to the time associated with one complete operation of a repetitive process but in the context of industrial control systems. The choice in this document is to use the term *cycle time* as far as possible to express the interval of time between two consecutive blocks because the terminology is not yet stabilized.

5.4 Interest of the historical chart

Chosen approach To better understand the value of analyzing the time between two blocks, it is possible to consult the historical graph of the variation of the cycle time over time. We find on the internet the possibility of seeing the mining time [22] and in another site the block time [23], which corresponds to the time between two blocks that it is called in this document cycle time. Both can have their interest, it is chosen to show the second to fully understand the problem associated with figure 5

Difficulty of interpretation Blocktime corresponds here to what we call cycle time. However, the value is that of the average block time. We see that there is a peak during the fall of 2018 of 14 minutes followed at the same period by an unusually low value of 8 minutes. How to interpret this? Chance is partly responsible of theses small variations on time scale because we have seen that the proof of work is a probabilistic process. However, on larger time scale, there are significant variations linked to other phenomena. Cycle time can be used as a summary parameter in the operation of supervision or management of the blockchain, and this is an idea of this document.

An investigation from the symptoms As we see with the history, the variations of block time are not easy to interpret but it is a symptom that something has happened. *The idea is to accept that while expected block time is useful in designing a clock, the average value of the block time gives us an indication that something has happened.* It could be an alert parameter, comparable to a symptom, which calls for additional investigations. Beyond that, additional work is needed to analyze in more depth how to exploit this parameter.

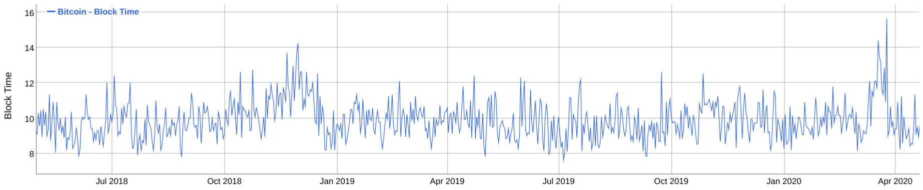

Figure 5: Bitcoin Cycle Time historical chart

6 Function of control

6.1 Blocktime and difficulty parameter

In the previous paragraph, we show the behavior of the variation of the cycle time over time. However, the question arises whether it is possible to have control over this parameter ? Contrary to what one might think, the difficulty in solving the cryptographic puzzle (find the nonce) is not constant, due to the increased performance of the specialized integrated circuit (Asic) used by the minor to calculate, and the mathematical property of the algorithm. There is a mathematical relationship between difficulty D, the hash rate, and the time between two blocks. The hash rate is the unit of the processing power of the minor's hardware, in other words the computational speed expressed in hash/second, and depends on the hardware device used. Zhang and Ma (2020) [20] propose a formulation which takes into account the delay of propagation :

$$\tau_k = \frac{D_k}{h_k} + t_k^p = \frac{D_k}{H_k}$$

with :

- the block produce time T_k
- the expected block produce time τ_k
- D_k is the block difficulty
- h_k is the actual real-time network hash rate
- t_k is the overall delay which includes the propagation delay among peers and the delay for peers to verify the block

The authors propose the nominal hash rate H_k, which offers more convenience for analysis. This formula expresses the mathematical relationship that maintains the block time constant. Küfeoğlu and Özkuran (2019) [21] study the issue of the hash rate readjustment delay and estimate the need to update the difficulty at 14 days. Now that we understand the mechanism, the question arises whether this behavior can be described within a larger theoretical framework.

6.2 Control loop and blockchain

Notion of control loop In many industrial applications, it is necessary to maintain physical measurements despite the disturbances that can influence theses measurements. We design systems where the output is aligned whatever the environment. To propose an approach from the field of control theory, we illustrate by the example of a feedback control loop, whose principle is to regulate the variation of an output signal by a readjustment of the input signal according to the variations of the output signal.

1. The observation is related to the measure to control.

2. The reflection determine the difference between the measure we observe and what we want.
3. Function of the difference, an action is to be taken

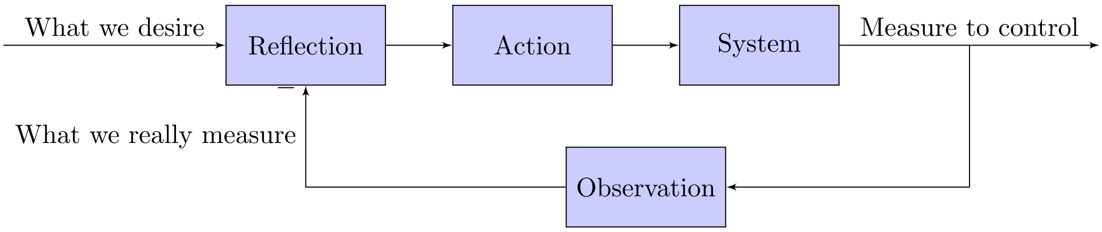

Figure 6: Functional structure of a control loop

Adaptation to the case of PoW If we adapt to the case of blockchain, this gives the result of figure 7. Δt in input corresponds to the expected time between two block and δt is the variation of this difference. However, in a practical terms, a corrective action by adjusting the difficulty D is required almost 15 days, and not at each time a new block is created.

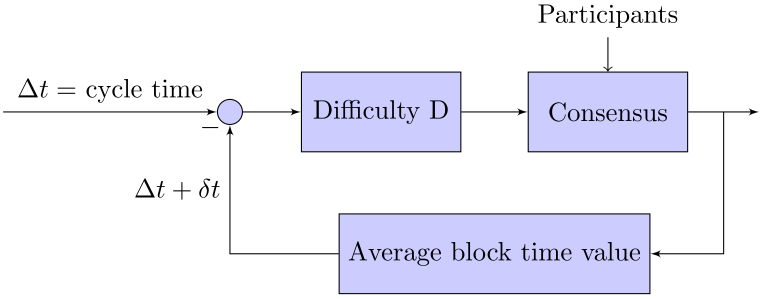

Figure 7: The adjustment of the difficulty allows to keep constant the time interval between two blocks.

Adaptation to others types of consensus The main idea is that something (the input) must remain constant. This is not necessary Δt and the idea of the adjustment parameter D as a parameter necessarily intrinsic to the protocol can be discussed. An adaption to others forms of consensus algorithms must be made from the foundations of the model.

Governance by design The control loop is a general principle that can give birth to new research ideas in order to develop the field of control system with the integration of the blockchain case, and vice versa. We must not forget that blockchain is a technology that must allow high-level mechanisms of self-governance. As we say that blockchain is secure by design, we can say that blockchain is (or should be) governable by design : it is by building on the properties of the protocol that control mechanisms could be put in place.

7 Industrial requirements and possibles actions

In the blockchain life cycle, the exploitation phase asks specific questions and involves the following requirements :

- The governance of the consensus algorithm during the blockchain operation phase improves the sustainability of the project. This implies long-term maintenance and *resilience* capacity for long-term operations. I use the term resilience in the sense of control system as tolerance to fluctuations. We lack experience concerning theses aspects.
- To validate this approach to other types of consensus : all consensus algorithms, in particular in permissioned mode, and all models of consensus are concerned and this requires further examination. Shorish (2018) [6] develops this idea and cites the two conditions of a transition function between blockchain states, the second being to be consensus agnostic.

This document offers a first approach to theses issues. Here are some examples of concrete actions to be taken :

- To improve the terminological framework, taking into account existing standards, in particular concerning the mechanisms of self-regulation of an industrial or digital system. The choice of the expression cycle time in this document is an example of what can be done.
- To model the interactions between on-ledger and off-ledger participants, for different phases of the blockchain life cycle, because understanding the different roles is strategic in governance issues. To what extend, and by what means can these interactions be linked to governance support functions ?

8 Conclusion

We have seen in this document that it is possible to associate the properties of timestamp and ordering between events with the time parameter in a blockchain system. By extension, the alert and control functions can be used depending on the characteristics of the cycle time. All this constitutes an abstract blockchain structure, which I call *temporal structure of the blockchain* constituted by this four functions : timestamp, ordering, alert, and control. The purpose of this document is to draw attention to these points and to lead to further work. A next step involves ensuring the continuous improvement of the data governance functions by developing the associated technical framework.

References

[1] Satoshi Nakamoto. A peer-to-peer electronic cash system. *Bitcoin.–URL: https://bitcoin. org/bitcoin. pdf*, 2008.

[2] ISO. Iso 8601-1:2019(en), date and time — representations for information interchange — part 1: Basic rules, 2019. (Accessed on 03/26/2020).

[3] Jan Ladleif and Mathias Weske. Time in blockchain-based process execution. *arXiv preprint arXiv:2008.06210*, 2020.

[4] ISO. Iso/tr 23455:2019(en), blockchain and distributed ledger technologies — overview of and interactions between smart contracts in blockchain and distributed ledger technology systems. (Accessed on 11/26/2020).

[5] Leslie Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications*, 1978.

[6] Jamsheed Shorish. Blockchain state machine representation. Technical report, Center for Open Science, 2018.

[7] Cambridge English Dictionary. Meaning of computation in english. (Accessed on 04/13/2020).

[8] ISO. Iso 22739:2020(en), blockchain and distributed ledger technologies — vocabulary. (Accessed on 11/21/2020).

[9] Gregory Trubetskoy. Blockchain proof-of-work is a decentralized clock. `https://grisha.org/blog/2018/01/23/explaining-proof-of-work`, January 2018. (Accessed on 12/21/2020).

[10] Samuel Brooks. Blockchain: the infinite state machine. `https://medium.com/@samuel.brooks/blockchain-the-infinite-state-machine-ffc39f32e182`, April 2014. (Accessed on 09/01/2020).

[11] ISO. Iso 16484-2:2004 - building automation and control systems (bacs) — part 2: Hardware, 2004. (Accessed on 05/04/2020).

[12] Ricardo Pérez-Marco. Blockchain time and heisenberg uncertainty principle. In *Science and Information Conference*, pages 849–854. Springer, 2018.

[13] Melanie Swan. Magic blockchains, but for time? blocktime arbitrage. `https://ieet.org/index.php/IEET2/more/Swan20151202`, December 2015. (Accessed on 09/01/2020).

[14] Melanie Swan. A new theory of time: X-tention is simultaneously discrete and continuous. `https://ieet.org/index.php/IEET2/more/Swan20160428`, April 2016. (Accessed on 04/10/2020).

[15] François Bonnet, Quentin Bramas, and Xavier Défago. Stateless Distributed Ledgers. working paper or preprint, June 2020.

[16] Yvan on Tech. Bitcoin and byzantine generals - youtube. `https://www.youtube.com/watch?v=kE51N84hBxU`, May 2017. (Accessed on 10/30/2020).

[17] Satoshi Nakamoto. Bitcoin p2p e-cash paper. `https://www.metzdowd.com/pipermail/cryptography/2008-November/014849.html`, November 2008. (Accessed on 10/23/2020).

[18] Prabath Siriwardena. The mystery behind block time. `https://medium.facilelogin.com/the-mystery-behind-block-time-63351e35603a`, October 2017. (Accessed on 04/30/2020).

[19] Melanie Swan and Primavera De Filippi. Towards a Philosophy of Blockchain . *Metaphilosophy*, 48, October 2017.

[20] Shulai Zhang and Xiaoli Ma. A general difficulty control algorithm for proof-of-work based blockchains. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3077–3081. IEEE, 2020.

[21] Sinan Küfeoğlu and Mahmut Özkuran. Energy consumption of bitcoin mining. Technical report, Faculty of Economics, University of Cambridge, 2019.

[22] Data.bitcoinity.org. Average time to mine a block in minutes. `https://data.bitcoinity.org/bitcoin/block_time/5y?f=m10&t=l`. (Accessed on 12/09/2020).

[23] Bitinfocharts.com. Bitcoin block time historical chart. `https://bitinfocharts.com/comparison/bitcoin-confirmationtime.html`. (Accessed on 04/30/2020).