Quality-by-design-engineered pBFT consensus configuration for medical device development

Yaël Kolasa, Thierry Bastogne, Jean-Philippe Georges, Sylvain Kubler

Quality-by-Design-engineered pBFT Consensus Configuration for Medical Device Development

Yaël Kolasa[1], Thierry Bastogne[1], Jean-Philippe Georges[2], and Sylvain Kubler[2]

***Abstract*— Health product development has been lately tainted by wariness in manufacturers, which has reduced trust in the system. It also affects Digital Health were patients' big data flows generated by numerous sensors are subject to increased security and confidentiality to lower the risks incurred. Our aim is to increase trust in the system again by implementing a dedicated Blockchain solution where data are automatically stored, and where each actor in the development process can access and host them. Blockchain has its downside, such as a subefficient management of big data flows. This study is a first step toward defining a Blockchain solution that will not deteriorate the Quality of Service in this particular context by using the Quality by Design approach. We will mainly focus on the time to consensus attribute which affects both of them. From our experiments' results generated after running screening design and surface response design on a practical Byzantine Fault Tolerance (pBFT) simulator, we find that the transmission time and the message processing time are the most impacting factors.**

I. Introduction

In the past few years, health product development has been tainted by numerous trials where some manufacturers of medical devices have been accused of omitting or falsifying data in order to receive marketing approval, thus lowering trust in the system. Digital Health has not escaped this trend with the risks on the patient's personal health data. To this can be added the advent of wearable sensors generating longer and longer personal data and requiring new solutions to protect them and to better ensure their integrity. Our aim is to remove the historical untrusty third party known as the manufacturer whom is by oneself, for now, sending data to the Notified Bodies and European Medicines Agency after collecting them from laboratories. We want to make laboratories' Internet of Things sensors directly store their big data flow into a blockchain hosted by every stakeholder, ensuring for everyone trust in the data generated. The culprit, here, is to define a solution that could manage said flow without sacrificing Quality of Service or Quality of Experience. Blockchain can be a solution, but it has its culprits.

Quality by Design (QbD) is a systematic and dynamic risk-based approach made to lower the risks of failure (being out-of-specifications) by using statistical modeling methods. It is used since 1980 in the automobile industry, and since 2000 in the pharmaceutical area. In this study, we apply this risk-based engineering approach to a blockchain simulator in order to (i) identify the critical design parameters and (ii) determine their settings leading to the expected technical specifications with an acceptable probability. The implementation of QbD requires to apply statistical methods for the design of experiments, which allow us (i) to collect informative data with a minimal number of trials and (ii) to estimate probabilities and risks. We will begin by an overview of Blockchain's structure, then continue by explaining our Quality-by-Design approach and experiments, to finish by the results, and a conclusion.

[1] Yaël Kolasa, Thierry Bastogne are with Université de Lorraine, CNRS, CRAN, F-54000 Nancy, France & CYBERnano, Villers-lès-Nancy ykolasa@cybernano.eu, thierry.bastogne@univ-lorraine.fr

[2] Jean-Philippe Georges, Sylvain Kubler are with Université de Lorraine, CNRS, CRAN, F-54000 Nancy, France jean-philippe.georges@univ-lorraine.fr, s.kubler@univ-lorraine.fr

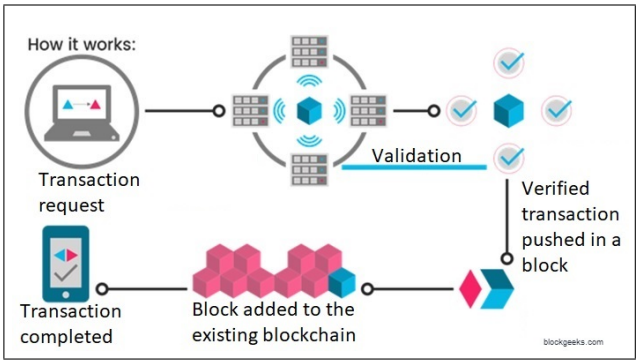

Fig. 1. Mechanics of a blockchain (blockgeeks.com)

II. Blockchain's structure for e-Health

Blockchain is an open, distributed public ledger storing all transactions in a secure and verifiable way. It works as a huge, public, secure and decentralized data-store [1], [2], [3], [10].

A. Blockchain mechanics

Blockchain revolves around transactions, as seen in Fig. 1. It implies sending something, whether it is data, or just cryptocurrency, from one address to another one. In the most known public Blockchains such as Bitcoin, each computational step has a cost [5], like a fee, that is added to the transaction.

Each transaction needs to be verified and validated through a consensus[1] process. Once the consensus is achieved, the transaction is stored in a block[2]. Each block is linked to the hash[3] of the previous block, then hashed in turn for the next block to be added [2], [4]. That way, any attempt to modify or delete a block will result in a disrupted chain where one block's hash will not correspond to the previous block in the chain, thus making the blockchain tamper proof [2], [3],

[10]. The blockchain is replicated in each running instance of it in the network, called node, and to modify an existing block, a majority of users must agree to it [1], [2], [10].

B. Blockchain types

Blockchain is mainly known as the technology behind Bitcoin [1]. But there are numerous other types of Blockchain structures. A blockchain can be:

- public or permissionless: fully open, everyone can access it and make transactions, or participate in the consensus process to validate transactions [1], [2], [3], [10], like Bitcoin, or Ethereum ;
- consortium, or hybrid: the consensus process is here controlled by a preselected set of nodes, potentially managed by several organizations [2], [3], [10] ;
- private: centralized, this type of blockchain is controlled by an organization giving the access rights to the system [2], [3], [10].

C. Blockchain consensus

A consensus is a way for different nodes to agree on the validity of a transaction, and to update the ledger with a set of coherent and confirmed facts [2]. Some of the most used consensus are:

- proof of Work: used by Bitcoin, Ethereum [2], [10], users (known as miners) need to verify hashes or solve mathematical problems to validate the transactions. Miners are rewarded with the blockchain's currency for their computational work [4] ;
- proof of Stake: used in Ethereum's Project Sharding. The user must prove that he possesses a certain amount of the currency to be able to validate new blocks in the blockchain [2] ;
- proof of Authority: used in private Ethereum blockchains. One node or more are authorized to validate and add blocks to the chain [2].
- practical Byzantine Fault Tolerance (PBFT): used in private HyperLedger Fabric v0.6. Its base postulate is that less than one-third of the nodes are faulty (f), which means that its network is composed of at least n nodes where $n = 3f + 1$ in order to overcome faulty nodes [7].

D. Blockchain specificities

By definition, Blockchain is well suited for guaranteeing data integrity. Once stored, they are unalterable, which is a crucial point for e-Health. Furthermore, as every piece of data is replicated on multiple nodes, it also raises trust between each actor of the network, whether they are sending the data or managing a node like an hospital.

But this replication technology is also a bottleneck when it comes to multiple sensors collecting data with a high frequency which need to store a great number of transactions within a limited time frame.

In this context, it is crucial to design the Blockchain architecture suited to the characteristics of the targeted e-Health applications. To that aim, we propose a methodology based on the Quality by Design approach.

III. Quality by Design

The target product profile is a blockchain structure devoted to an e-Health application ensuring the safe transaction of long physiological signals between partners. The blockchain shall have several nodes hosted by every stakeholder, like the manufacturer, the notified bodies, or national health agencies. The main problem here is the constant flux of data induced by several sensors sending them to the different nodes. As stated before, blockchain is not primarily made to work with big data flows, like those an ECG could produce, but its advantages in term of trust and data integrity make it worth to overcome the disadvantages.

Because of the e-Health context of our study, our Blockchain type of choice is set to "private" in order to ensure confidentiality, where only whitelisted entities can access it [7]. Therefore, a more efficient consensus than Proof of Work, or Proof of Stake, can be used[7]. pBFT is faster, more resilient to faulty nodes, and an attack does not reduce much its performances.

A. The Blockchain simulator

In order to conduct the designs of experiments, we chose to use a pBFT simulator. It has the benefit of reducing the costs, and is easier and faster to configure than a real implementation of a Blockchain. The said simulation uses Stochastic Rewards Nets (SRN) with a randomized seed, as seen in Fig. 2, to reproduce the behaviour of HyperLedger Fabric v0.6 practical Byzantine Fault Tolerant (pBFT) consensus and was developed by Sukwhani et al[7]. The pBFT consensus is divided into three phases, the first being the "Preprepare" phase, where the leader of the validating nodes receives a request and makes a proposal of a solution to the other nodes. The second phase called "Prepare" sees the nodes coming to an agreement about the proposal, and the last phase called "Commit" sees the nodes committing their agreement and replying back. They also execute the request on their own to ensure its validity. The SRN has guard functions as seen in table I, such as C_0, which opens a place only when a sufficient number of messages have been received, and said number depends of the number of faulty nodes f.

The simulation allows six inputs that each defines a group of transitions with a distribution law set with identical parameters, which can be seen in table III. This makes the inputs more manageable, where, e.g., each transmission time, which is the amount of time taken by consensus messages between peers[7], is similarly set by a distribution law. It allows to use them together as one factor, which makes the scalability of the simulation easier. For example, we can see on Fig. 2 the SRN for four nodes with 24 transmission

[1]Consensus: rule allowing to validate or not a transaction

[2]Block: unit containing transactions' data

[3]Hash: mathematical functions allowing to transform a string of characters of undefined length into another string of fixed length

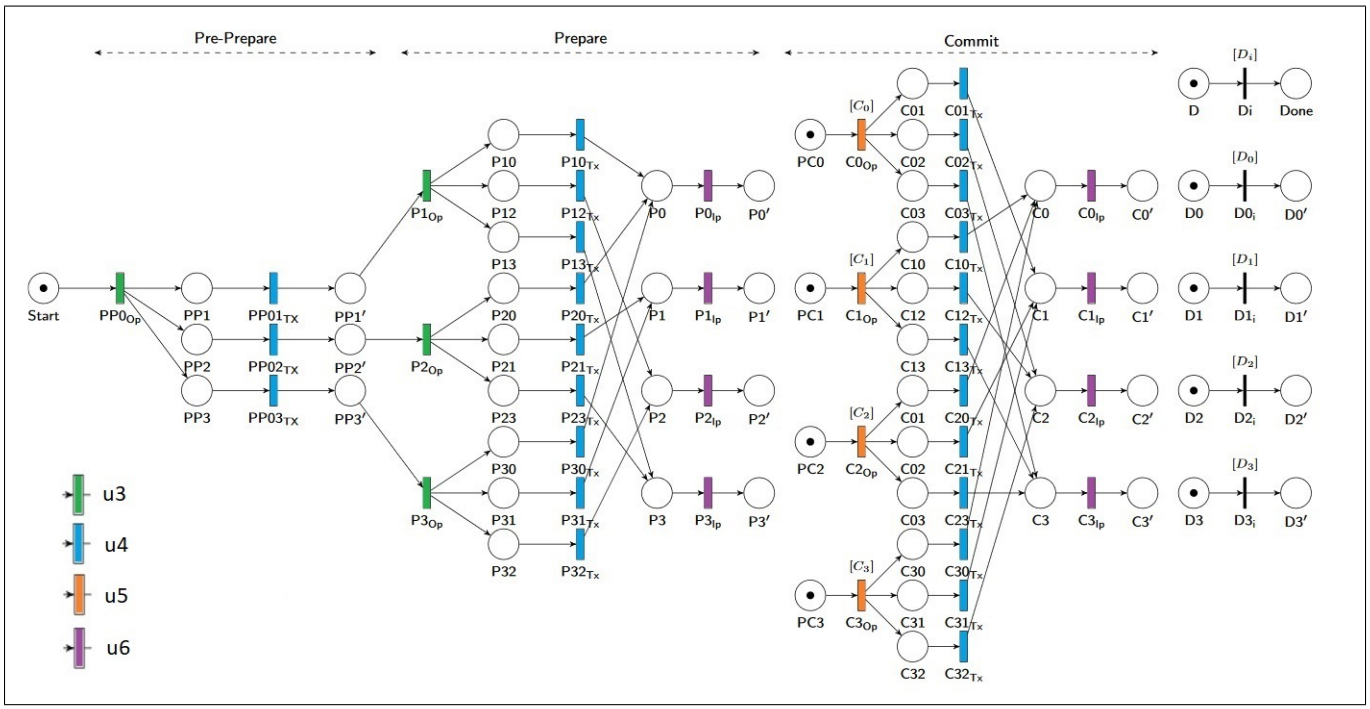

Fig. 2. SRN model for pBFT consensus with four nodes[7].

TABLE I

GUARD FUNCTION FOR SRN MODEL IN FIG.2

Name	Function
$[C_0]$	If #$P0$' $\geq 2f$, return 1, else 0
$[C_x]$, $x \in (1,2,3)$	If #P_x' $\geq 2f-1$, return 1, else 0
$[D_x]$, $x \in (0,1,2,3)$	If #C_x' $\geq 2f$, return 1, else 0
$[C_i]$	If $\Sigma_{y\in(0,1,2,3)}$ #D_y' $\geq 3f+1$, return 1, else 0

TABLE II

CQA FOR AN E-HEALTH BLOCKCHAIN SOLUTION

Specification	CQA
Quality of Experience	Transaction throughput [6] Transaction latency [6]
Integrity	Mathematical proof (pBFT) [8]
Quality of Service	Time to consensus [7]
Costs (€, energy)	Server hardware, sensors, kWh...

transitions that otherwise would be 24 inputs on their own. For 50 simulated nodes, the number of transmission time transitions will be higher than 4700, without speaking of the other kind of inputs seen before. This would make the experiments harder to perform.

B. Critical Quality Attributes

In QbD, Critical Quality Attributes (CQA) are defined, as the metrics, or outputs, from which the system will be monitored and that must stay within an appropriate range to ensure the desired properties. We can see in Table II different CQAs such as transaction throughput[6] or transaction latency[6] which may affect the Quality of Experience for the user, whereas the type of consensus chosen will affect the integrity of the Blockchain. The Quality of Service will be affected by the time to consensus[7] after which a block is considered validated. For a practical solution, the costs are also to be taken into account, such as server hardware, sensors, electricity cost, etc. For our first sets of experiments, we focus on the time to consensus quality attribute which directly impacts the quality of service of the system.

C. Critical Process Parameters

A Critical Process Parameter (CPP) is a process parameter which has a significant impact on a critical attribute when it varies.

It should therefore be monitored or controlled in order to ensure the desired performance. CPPs can be viewed as "inputs" to the system. Thereafter, the CPPs correspond to the inputs of the simulations seen previously, such as transmission time or the number of nodes[7]. The six tested inputs are presented in Tab.III and noted $u1$ to $u6$. They will each affect a specific group of transitions, as seen on Fig.2 with the color coding. The modality values for $u3, \cdots, u6$ have been inspired from the distribution functions defined in [7].

IV. SIMULATED EXPERIMENTS

A. Screening design

The first campaign of *in silico* experiments aims at identifying the most critical factors among the six tested inputs. The design of experiments is based on a Plackett-Burman matrix, generated by AZURAD software[9], in which each factor takes two levels as seen in Table III. The implemented

TABLE III
PROCESS PARAMETERS

	Input	Level 1	Level 2
u_1	Nodes number	20	50
u_2	Faulty nodes number	1	5
u_3	Message preparation time in Pre-prepare and prepare phase	0.01	0.7
u_4	Transmission time	0.01	2
u_5	Message preparation time in Commit phase	0.01	0.2
u_6	Message processing time	0.01	0.4

screening design is composed of eight different conditions replicated ten times with a randomized seed to simulate the randomity of a real system. In [7], Weibull and hypo-exponential distributions are used to precisely simulate the behavior of the pBFT consensus. But to enable the implementation of the Plackett-Burman design, it was needed to change those distribution laws to thin uniform functions, centered on the level 1 and level 2 of each factor with a closed interval of 0.01 around the value, for the inputs $u3$ to $u6$.

B. Response surface design

The response surface design is used in a second step to identify the cause-effect relationship between the time to consensus and the input factors selected after the screening study. A quadratic response surface model structure was used:

$$\begin{aligned} Y_k &= b_0 + b_1u_1 + b_2u_2 + b_4u_4 + b_6u_6 \\ &+ b_1u_1^2 + b_2u_2^2 + b_4u_4^2 + b_6u_6^2 \\ &+ b_{12}u_1u_2 + b_{14}u_1u_4 + b_{16}u_1u_6 \\ &+ b_{24}u_2u_4 + b_{26}u_2u_6 + b_{46}u_4u_6 + E_k \end{aligned} \quad (1)$$

where the b_i, b_{ii}, b_{ij} are the model coefficients to be estimated from the simulated data. Y_k and E_k are the k-th values of the time to consensus and modeling residual, which is assumed to be a gaussian centered white noise. To estimate the model parameters, a central composite design was implemented and five validation points were added. The whole design was replicated five times.

V. RESULTS

A. Screening design results

We can see on Fig.3 the impact of each factor on the time to consensus. The parameter u_1 seems to reduce the time of consensus when it increases, but to achieve consensus, as seen in Table I, our model need to have at most $2f$ responses (with f as the faulty nodes number). As we have a defined number of faulty nodes in our simulation (Table III), the increase of u_1 allows for more nodes to answer in a short period of time to pass the guard, hence the reduction of time to consensus when the number of nodes increases.

The inputs that are affecting the most the time to consensus in pBFT are the transmission time that accounts for about 45%, the time to process messages accounts for about 25%, the number of faulty nodes accounts for almost 15%, and the total number of nodes accounts for about 13%. These four factors cumulated contribute for more than 95% to the problem of the time to consensus in pBFT. Their relationship with the time to consensus are identified in the second design.

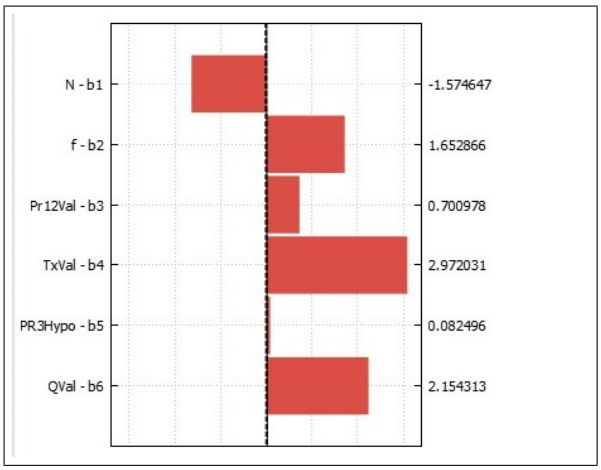

Fig. 3. Graph of the effect of each factor on the time to consensus

B. Response surface design results

Results show an adjusted coefficient of determination of $R_a^2 = 0.998$. As we can see in Table IV, the transmission time (u_3 here) is the most impacting factor with a coefficient of 3.11, followed by the time to process the messages (u_4) with 1.16 and the number of faulty nodes (u_2) with 0.95. The number of nodes (u_1) only scored -0.002. These results correlate the ones obtained with the screening design where they were ranged in the same order of impact on the time to consensus. The number of nodes (u_1) have no significant interaction with any other factor, whereas the number of faulty nodes (u_2) interacts with both the transmission time (u_3) for a coefficient of 0.01 and the time to process the messages (u_4) for a coefficient of 0.88. We can also see an interaction between u_3 and u_4 for a coefficient value of 0.165.

A cross-validation of the model has been made by generating 5 experiments with parameters taken from outside the learning model and they have been repeated 5 times each for a total of 25 experiments. The maximum difference between calculated and experimental results is at 6.26%, which is acceptable. From Fig. 4, we can see the interactions of u_3 and u_4 on the time to consensus for $u_1 = 35$ and $u_2 = 3$. It is clear that u_3 have a greater impact on the time to consensus than u_4 as the time increases faster on the ordinates' axis, but it is worth noting that the highest consensus' time is obtained only when both u_3 and u_4 have high enough values. From Fig. 5, the interactions of u_2 and u_4 on consensus' time for $u_1 = 35$ and $u_3 = 1.005$ can be seen. Having a low u_4 can compensate for a higher u_2, and even if the contrary seems true here, the number of faulty nodes cannot be lowered at will, as it is the main purpose of pBFT to deal with them.

TABLE IV

RESPONSE SURFACE DESIGN RESULTS

THE NUMBER OF * SHOWS THE IMPORTANCE OF THE FACTOR

	Coefficients	Standard deviation	p value %	
$b0$	4.466346	0.000387		
$b1$	-0.002024	0.000326	<0.100000	***
$b2$	0.952098	0.000326	<0.100000	***
$b3$	3.110588	0.000326	<0.100000	***
$b4$	1.157096	0.000326	<0.100000	***
$b1-1$	-0.044445	0.000666	<0.100000	***
$b2-2$	-0.060387	0.000666	< 0.100000	***
$b3-3$	-0.073236	0.000629	<0.100000	***
$b4-4$	0.041841	0.000598	<0.100000	***
$b1-2$	-0.000103	0.000865	90.526826	
$b1-3$	-0.000967	0.000969	32.052306	
$b1-4$	-0.000698	0.001007	48.967590	
$b2-3$	0.012624	0.000969	<0.100000	***
$b2-4$	0.887590	0.001007	< 0.100000	***
$b3-4$	0.165096	0.001007	<0.100000	***

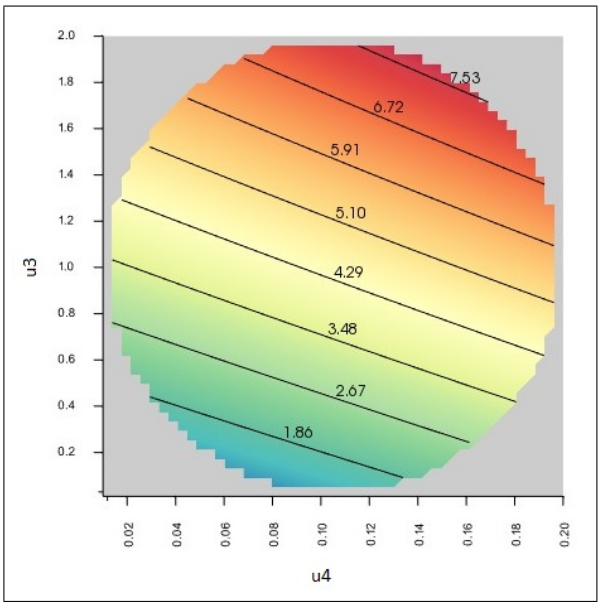

Fig. 4. Response surface, with $u_1 = 35$ & $u_2 = 3$

VI. CONCLUSION & PERSPECTIVES

The time to consensus can directly impact the Quality of Service of the solution by either increasing or lowering the speed of block validation, which defines how much data can be stored within a second. This also impacts the Quality of Experience of the user in this case, where it means, for example, lowering the sample period of the sensor to keep up with the slow time to consensus. In order to keep the time of consensus as low as possible, the transmission time must be kept the lowest by using a efficient network, which could be a limiting factor for sensors network, and a good processor for the processing of the messages should also help to reduce the time to consensus. The number of nodes directly impacts the integrity of the Blockchain by allowing more faulty nodes without compromising the data, but its role on the time to consensus must be further studied because of its intriguing

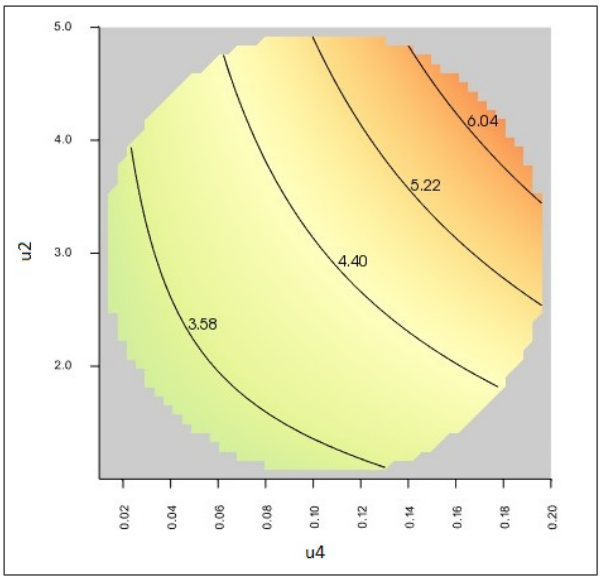

Fig. 5. Response surface, with $u_1 = 35$ & $u_3 = 1.005$

results. This work also emphasizes an original application of the Quality-by-Design engineering approach to *in silico* studies for safety assessment of technologies related to the Internet of Things.

Further studies on pBFT with more flexible transitions and realistic behavior, but also on a whole Blockchain simulation are on the way, to make a complete analysis of impacting factors on defined quality attributes such as Quality of Experience. It will then help to define a sturdy Blockchain solution for e-health innovative solutions.

REFERENCES

[1] M. Benchoufi et P. Ravaud, « Blockchain technology for improving clinical research quality », Trials, vol. 18, No. 1, p. 335, déc. 2017.

[2] L. Leloup, Blockchain, la révolution de la confiance. Eyrolles, 2017.

[3] A. Zhang et X. Lin, « Towards Secure and Privacy-Preserving Data Sharing in e-Health Systems via Consortium Blockchain », J Med Syst, vol. 42, No. 8, p. 140, août 2018, doi: 10.1007/s10916-018-0995-5.

[4] H. Shafagh, L. Burkhalter, A. Hithnawi, et S. Duquennoy, « Towards Blockchain-based Auditable Storage and Sharing of IoT Data », arXiv:1705.08230 [cs], mai 2017.

[5] X. Zheng, R. R. Mukkamala, R. Vatrapu, et J. Ordieres-Mere, « Blockchain-based Personal Health Data Sharing System Using Cloud Storage », in 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom), Ostrava, 2018, p. 1-6.

[6] Hyperledger Performance and Scale Working Group, « Hyperledger Blockchain Performance Metrics - White paper » https://www.hyperledger.org/wp-content/uploads/2018/10/HL_Whitepaper_Metrics_PDF_V1.01.pdf.

[7] H. Sukhwani, J. M. Martinez, X. Chang, K. S. Trivedi, et A. Rindos, « Performance Modeling of PBFT Consensus Process for Permissioned Blockchain Network (Hyperledger Fabric) », in 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), Hong Kong, Hong Kong, 2017, p. 253-255, doi: 10.1109/SRDS.2017.36.

[8] M. Castro et B. Liskov, « Practical Byzantine Fault Tolerance », in Proceedings of the Third Symposium on Operating Systems Design and Implementation, New Orleans, USA, 1999, p. 14.

[9] Azurad software, Version "expert". AZURAD SAS, Marseille, France, 2019.

[10] J. Yang, M. Onik, N.-Y. Lee, M. Ahmed, et C.-S. Kim, « Proof-of-Familiarity: A Privacy-Preserved Blockchain Scheme for Collaborative Medical Decision-Making », Applied Sciences, vol. 9, No. 7, p. 1370, avr. 2019, doi: 10.3390/app9071370.