Proof of Provision: Improving Blockchain Technology by Cloud Computing

Matthias Pohl, Abdulrahman Nahhas, Sascha Bosse and Klaus Turowski
Faculty of Computer Science, University Magdeburg, Germany

Keywords: Blockchain, Consensus, Proof Algorithm, Proof of Provision, Cloud Computing.

Abstract: Blockchain technology is mainly used in so-called cryptocurrencies and smart contracts. From a technological point of view, securing these applications requires a lot of computing power, which results in high energy consumption. The reason for this is the proof of work algorithm integrated in most cases. In order to promote the problem of high energy consumption and the sustainable use of cloud computing resources, this paper presents a consensus concept for use in a blockchain. We also present the classifications for discussion and give an outlook on a future evaluation.

1 INTRODUCTION

Even though the hype about blockchain technology and Bitcoin & Co. has died down in the meantime[1], there are still many data centers around the world that dig for cryptocurrencies. The energy requirements of Bitcoin mining industry alone estimated to be nearly 43.44 TWh yearly (Kosharnaya et al., 2018), which correspond to the demand of entire countries, as for instance, Peru, Hongkong, or Iraq (International Energy Agency, 2017). Another study predicts that the energy demand will even increase in the future to reach 7.67GW (de Vries, 2018), which is nearly consistent with the demand of Austria 8.2 GW (International Energy Agency, 2017). Yet from a worldwide perspective, a testimony presented to the U.S Senate committee on energy and natural resources suggests that the bitcoin mining process alone constitutes 1 percent of the overall worldwide energy consumption (Narayanan, 2018). The associated CO2 emissions of mining processes of cryptocurrencies can be obviously compared with the CO2 emissions of entire countries, as for instance, Bolivia or Portugal (Stoll et al., 2018).

The reason for this is the integrated use of a hashing algorithm and a deposited puzzle, whose difficulty increases with the increasing computing capacity of the actively mining participants in the Bitcoin network.

A blockchain network basically consists of participants, computing units, which are located in a peer-to-peer network. Each participant has a key that uniquely identifies it and a key pair to perform authorized activities on the network. In the case of a cryptocurrency, it is possible to transfer virtual amounts of money in this network. With the appropriate authorization, a participant can place a transaction of an amount to another participant in the network. To ensure that a participant can only transfer amounts that he has received once before, it is only possible to create a transaction on the basis of an older transaction.

If the transactions are posted correctly, you can also prevent amounts from being transferred more than once. This is made possible by the proofing algorithm and the consensus. After a certain period of time, the proof algorithm determines a checksum of the current accrued and prioritized postings so that these can no longer be changed or deleted. In addition, the result of the checksum calculation of the last time window is added. In the initial bitcoin technology, the proof of work is used here. The checksum is calculated using a hashing algorithm. In order to prevent a malicious recalculation of old transaction bundles, the difficulty of the calculation is influenced by constraints.

The block of transactions and checksum is then distributed in the network and recognized and redistributed by involved participants. If a block has been evenly distributed in the network, a consensus is reached on the current bookings. If two blocks are randomly distributed at the same time, both are initially accepted by all. As soon as a branch can no

[1] https://coinmarketcap.com/currencies/bitcoin/#charts

Pohl, M., Nahhas, A., Bosse, S. and Turowski, K.
Proof of Provision: Improving Blockchain Technology by Cloud Computing.
DOI: 10.5220/0007765605230527
In *Proceedings of the 9th International Conference on Cloud Computing and Services Science (CLOSER 2019)*, pages 523-527
ISBN: 978-989-758-365-0

longer be continued in a coming time window, it is no longer used. Some participants in the network pursuit the calculation of the proof, others participate the confirmation of the consensus and yet others only use the transaction functions as normal participants. The distribution ensures that each participant can monitor the entire accounting process by viewing the distributed ledger (Drescher, 2017; Wattenhofer, 2016). Furthermore, it is possible to link the transactions to conditions that are checked by so-called oracles (e.g. data feeds, user trigger). This allows contracts to be mapped (Szabo, 1997; Wood, 2014).

In this way, it is possible to replace classical financial institutions, since no third party is necessary anymore to confirm the transaction process and the correctness of the account balances. There exist the hyptothesis that the Bitcoin network consumes more energy than the booking and vending systems of financial institutions. For instance, Swiss banks consume 8.4GWh per year with 3233 devices (Aebischer, 1998). Nonetheless a reduction in energy use due to environmentally harmful factors is desirable.

In this paper, we want to present an approach to reducing energy consumption and the sustainable use of the data centers that have been established so far. This work does not claim to introduce a new cryptocurrency or a new detailed blockchain technology. In section 2, we give an overview of various proofing algorithms currently used by many cryptocurrencies networks and conceptually presented in the literature. In section 3, we will again explain the fundamental problem of the further use of the proof of work and also highlight the problems with alternative proof algorithms. We present a solution concept in section 4 and discuss the integrated approaches as well. Finally, we describe an evaluation concept and give a short summary as well as an outlook for future research.

2 RELATED WORK

In this chapter we will present an overview of existing proof concepts. We will use a literature review with the key terms 'blockchain' and ('proof of' or 'consensus'). We were able to assign the following essential concepts from the literature.

2.1 Proof of Work

The proof of work algorithm has been known since the initial idea of the bitcoin network. According to cryptoslate[2], it is now used by most blockchain technology. The principle has already been explained in the introduction, since it originates from the basic idea of the technology. A set of transactions, together with the hash value of the last transaction bundle and an additional variable, is re-input into a checksum calculation. The checksum must meet a certain condition to be accepted by the blockchain. By changing the variable, you can change the checksum. The process is also called hash puzzle. The many calculations generate the high computing effort. A block that is to be newly created is always appended to the longest chain of the blockchain if there is a branch (Nakamoto, 2008).

2.2 Proof of Stake

The proof of stake goes back to the developments of the PeerCoin (King and Nadal, 2012) and the BlackCoin (Vasin, 2014). In this concept, the process of checksum calculation is changed by limiting the puzzle to be solved by the equation $proofhash < coins * age * target$. The target is predefined per calculation round, while the age of coins depend on mining user's stake. Furthermore, a bookkeeper who calculates the checksum must have an amount of coins to get a lighter puzzle. Participants who have a high amount of coins can solve the puzzle with a higher probability. In the original variant, the age of the coins was taken into account. A further development of the proof of stake is the delegation of the checksum calculation to selected participants per calculation block (Kiayias et al., 2017). The chain with the most used coin days or used coins will be added continuously.

2.3 Proof of Importance

The proof of importance begins with the determination of the importance of the participants in the network. Each participant who holds coins in the network for a certain time and takes a topologically suitable position in the network can thus increase its importance. The importance serves to determine the participants who are allowed to calculate the next block. The calculation of the checksum is based on the proof of work, but the difficulty of the puzzle is so reduced that the puzzle can be solved in about 60 seconds. The continuation of a branch depends on the total importance in it (NEM, 2018).

2.4 Proof of Luck

The checksum is calculated at the proof of luck using a randomized procedure. Each participant who starts the calculation receives a time delay, which is

[2] https://cryptoslate.com/cryptos/proof-of-work/

randomly determined for each participant. The network communication is optimized because the random component does not send so many checksum results through the network. The proof of luck contains a proof of work. If a branch is formed, the chain with the greatest accumulated luck is used in the next step (Milutinovic et al., 2016).

3 PROBLEM

We have already highlighted the problem of increased energy consumption in the Bitcoin checksum calculations in the introduction. Although, Bitcoin mining has its own specially designed data centers, there are other cryptocurrencies that run in standard data centers. The energy consumption of datacenter industry witnessed a gigantic growth in the last years. Statistical analysis suggests a total growth of roughly a hundred percent between 2005 and 2010 with a steadily increasing behaviour (Koomey, 2011). Clearly, the associated CO2 emissions of datacentre's operations reported an immense growth and observed to be the most accelerated growing carbon-footprint compared to various IT-fields (Avgerinou et al., 2017). Accordingly, the worldwide energy consumption and its associated carbon-footprint have triggered extreme anxiety on an international level. In the meantime, this problem stands in the top agendas of many countries. For instance, the European Union has introduced many new regulations in addition to some initiatives and research to reduce the proportion of datacenter's operations of CO2 global emissions (Avgerinou et al., 2017).

In this paper we address the enormous environmental pollution caused by the high energy consumption of Proof-of-work-cryptocurrencies and the sustainable use of data centers.

The various concepts for changing the consensus in section 2 have always been motivated by the solution to the same problem. The computing time of the proof of work should be reduced to regulate energy consumption. However, the approaches do not really do justice to the claim, since only the puzzle was simplified and the possible circle of authorized persons for the checksum calculation was regulated. The proof of work still has to be executed. Furthermore, in some concepts (e.g. proof of stake) participants are preferred, so that there is a risk of third party creation.

We therefore continue to address the abolition of the proof of work from the consensus mechanism and the avoidance of preferential treatment of participants.

4 CONCEPT

We now present a consensus concept of a blockchain technology that can be used to provide cloud computing instances in data center. As a starting point we consider a peer-to-peer network in which transactions should take place.

Participants. The participants of this blockchain technology are primarily cloud providers, who always keep the ledger of the blockchain available and can provide cloud computing instances in their own data center. If a subscriber does not fulfill the second condition, he is only active in the network as an auditor of the blockchain. Other participants initially take on the role of the customer who commissions a cloud computing instance.

Based on familiar blockchain concepts, all participants are equipped with a unique ID and a key pair for authorization.

Smart Contract. The provision of cloud computing instances by a provider to a customer is recorded in a smart contract. This contract specifies the time of availability, equipment and the amount due. The amount due is transferred at the end of the contract period or at defined intervals. The smart contract secures the future receipt of the amount.

Availability is checked via a software component running on the cloud computing instance. At short intervals, the software component sends a signal to the smart contract. If a user is active on the instance, the software sends a proactive status. If the instance is available, the status is transmitted as active, otherwise as inactive. If an instance is inactive for too long, defined rules (e.g. fines, invalidity of contract) will apply.

Software Component. The software component consists of a module that transmits the status and an additional module that calculates the checksums of the blockchain transactions. While a customer is using an instance, the software sends a customer signed combination of identifier (e.g. MAC address) and timestamp. This ensures that the user is really using the instance. If the instance is in sleep mode, only a provider signed variant can be sent. Otherwise, the smart contract receives the message that the instance is inactive. The second module of the software is described in more detail in the next section.

Block Creation. A new block containing a set of transactions can be created in a predefined time window (e.g. every 60 seconds). If a time limit is reached, a time-based random value is formed by the blockchain. However, this value can only be read from a current blockchain, i.e. the signals of all running Smart Contracts must be current. This random value is used to select one of all providers that hold smart contracts with the status 'proactive' from the blockchain. This provider in turn selects a customer who is then commissioned to calculate the checksum of the transaction block. The checksum of the last block, a time stamp and the provider and customer ID are added to this block. The checksum of the new block is then encrypted again by the customer and transmitted to the blockchain via the provider. For the checksum calculation, a hash value function can also be used in this concept. A user who has been commissioned with the calculation does not have to accept this order. If this is the case, the system waits for the next time window and the next random value determination. A reward for the calculation serves to motivate the execution.

Synchronisation. The new block is split after creation in the network and added to the blockchain. Due to overlapping calculation times and the simultaneous distribution of a block in the network, branches of the blockchain can occur. If this is the case, the string with the most recent end will be used when creating the next block.

5 DISCUSSION

We first take a look at the extent to which the concept meets the addressed problems. The calculations of the checksum in the consensus could be reduced from indefinitely many to one. Further, if a specific resource is not booked, the data center could shut down the resource. This means that the energy consumption associated with this can be significantly reduced compared to other consensus procedures. By distributing the calculation task, centralization of transaction management is avoided, so that the second problem is also addressed.

Due to the many status transfers, however, a high network load is to be expected. Optimizing traffic data or reducing transmissions by changing the time pattern could help.

Furthermore, a general change in cloud provisioning could be made. The control of the provisioning could run completely over the blockchain, so that the provider only hangs his systems in the blockchain network and only takes care of the hardware. Different provision strategies could be integrated into the blockchain technology to ensure a standard in quality over all smart contracts. A detailed modelling of the smart contracts would also make it possible to integrate different service level agreements (SLA). Control mechanisms necessary for verification can be included in the software component. The current design is only designed for cloud instances such as desktop units, as proactive use is always assumed. Further mechanisms have to be developed to check different instance variants.

In the following section we give an outlook on how to proceed.

6 CONCLUSION

For further investigating our hypothesis and evaluating the presented concept, we will rely on the Design Science Research evaluation pattern presented in (Sonnenberg and vom Brocke, 2012). Briefly, the authors presented a generalized evaluation pattern that comprises four evaluation stages, which lies between four main activities: problem identification, design, construction and use. The presented argument flow and the analyzed literature strongly points out the extensive use of computational resources for mining process due to the inefficient adopted consensus mechanisms (e.g. Bitcoin). Accordingly, CO2 emissions of this industry will keep rising in the future which already pose significant problem. The initial design of the presented concept will be further analyzed from a security point of view to insure that the presented consensus mechanism will overcome attacks such as 51%-attack or Double block generation. Furthermore, a simulation model will be built to investigate the feasibility of the concept to enhance the operations of cloud providers to reduce their energy consumption using the concept of proof-of-provision, which provide a higher form of transparency through incorporating the SLA conditions and violations. In the simulation, we will demonstrate the role of dynamic scaling of virtual machines and virtual machines live migration to reduce energy consumption in datacenters. One can argue that the transparency is a major problem that demotivate cloud services consumer from accepting specific forms of services as for instance, a dynamic scaling of virtual machines or hibernating computing resources if they are not in use. In the third and fourth evaluation phases we will develop a prototype to demonstrate the applicability of the concept and further investigate the mentioned security matters.

REFERENCES

Aebischer, B. (1998). Energieverbrauch von Automaten und Energiesparmoeglichkeiten.

Avgerinou, M., Bertoldi, P., and Castellazzi, L. (2017). Trends in Data Centre Energy Consumption under the

European Code of Conduct for Data Centre Energy Efficiency. *Energies*, 10(10).

de Vries, A. (2018). Bitcoin's Growing Energy Problem. *Joule*, 2(5).

Drescher, D. (2017). *Blockchain Basics*. Apress, Berkeley, CA.

International Energy Agency (2017). Key World Energy Statistics 2017.

Kiayias, A., Russell, A., David, B., and Oliynykov, R. (2017). Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. In Katz, J. and Shacham, H., editors, *Advances in Cryptology – Crypto 2017*, volume 10401. Springer International Publishing, Cham.

King, S. and Nadal, S. (2012). PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Technical report.

Koomey, J. (2011). Growth in data center electricity use 2005 to 2010. Technical report.

Kosharnaya, Y., Yanchenko, S., and Kulikov, A. (2018). Specifics of Data Mining Facilities as Energy Consumers. In *Proceedings of the XII International scientific and technical conference Dynamics of Systems, Mechanisms and Machines (Dynamics)*. IEEE.

Milutinovic, M., He, W., Wu, H., and Kanwal, M. (2016). Proof of Luck: an Efficient Blockchain Consensus Protocol. In *Proceedings of the 1st Workshop on System Software for Trusted Execution - SysTEX '16*, Trento, Italy. ACM Press.

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Technical report.

Narayanan, A. (2018). United States Senate, Committee on Energy and Natural Resources Hearing on Energy Efficiency of Blockchain and Similar Technologies.

NEM (2018). Technical Reference.

Sonnenberg, C. and vom Brocke, J. (2012). Evaluations in the Science of the Artificial – Reconsidering the Build-Evaluate Pattern in Design Science Research. In Peffers, K., Rothenberger, M., and Kuechler, B., editors, *Design Science Research in Information Systems. Advances in Theory and Practice*, volume 7286. Springer Berlin Heidelberg, Berlin, Heidelberg.

Stoll, C., Klaasen, L., and Gallersdoerfer, U. (2018). The Carbon Footprint of Bitcoin.

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

Vasin, P. (2014). BlackCoin's Proof-of-Stake Protocol v2. Technical report.

Wattenhofer, R. (2016). *The science of the blockchain*. Inverted Forest Publishing, 1st edition.

Wood, D. G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Technical report.