Implementation of transactions based blockchain utilizing proof of work consensus

A. Williams Gava

School of computing, Napier University, United Kingdom

40212064@live.napier.ac.uk

Proof of work, web application, blockchain

Abstract

Blockchains have undergone a rapid grow both in commercial and academic circles, creating specialized groups focused in implementing specialised blockchains. This paper aims to provide the basics of implementation of a blockchain and evaluating its components possible alternatives. It is presented the creation of a network to be able to utilize the blockchain in a decentralized structure and understand possible constraints. The results indicated that is it certainly possible to create a blockchain with transaction items and the main issues registered are the correct implementation of distributed network, creating an interface where nodes can communicate and decide the valid blocks mined.

Introduction

Nowadays Bitcoins became a buzzword because of its great popularity that skyrocketed since the 2008. In fact, as today, its circulating supply is quoted as 18 billion units[1], being the most popular cryptocurrency. It is not the only one, many other rose and became valid alternatives to the Bitcoin, each of ones focuses on different improvements: speed, throughput, lightweight. The great popularity is due to the cryptocurrencies' feature as being independent from any central authority, managing to create an electronic monetary system in which transactions can happen automatically and the network is able to autoregulate itself.

This was not a novel idea, in fact it was already proposed in the '90 but it failed and did not see the light until 2008 when Satoshi Nakamoto[2] proposed a model that would facilitate the

creation of a decentralized monetary network. Its model is based on the concept of blockchain which is a reliable and immutable record of transactions based on a distributed peer-to-peer network. It is also known as a public ledger in which all the transactions can be publicly visible and cannot be modified.

Bitcoin is based on a data structure known as blockchain which can be described as a decentralised, peer-validated crypto-ledger that provides a publicly visible, chronological and permanent record of all prior transactions. It can be envisioned as a spreadsheet in which everyone can add a row but cannot update or remove afterwards. They were introduced to solve the problem of double spending, in a system without a central entity that verifies each transaction, creating an environment of trust (no central entity as a bank or Ebay for example)[3].

One of the most known application of blockchain is in cryptocurrencies which are digital currencies that use cryptographical functions to conduct financial transactions. These currencies have a few benefits as not being controlled by a central government, like a bank, low processing fees, and transactions can happen directly between two parties.

Current use of blockchain are focused in financial domain, but other applications are developing in other areas. For example, big data and important information might be applied to blockchains because are secure and distributed, ensuring that it is original, since it is not possible to tamper them. The block chain could also be used to store user reputation information.

The general process of the blockchain can be summarized in a few points:

1. Unconfirmed transaction (or general item) received
2. Node gather a number of transactions into a block
3. Mining following the network's rules (proof of work, proof of stake or others)
4. Successfully mined block is propagated in the network and validated by the nodes
5. Mined block added to the chain of each node if satisfies network's specifications

Materials and Methods

The main components needed to build the blockchain project are: node, blockchain, block, transactions. Each node is responsible of its blockchain and the block mined.

The blockchain has been created following a general and simple architecture that is usually presented in any papers related to blockchain. More specifically, as it is shown from Zheng et. al[4], the blockchain is composed by a series of interconnected blocks having a structure similar as the one proposed below:

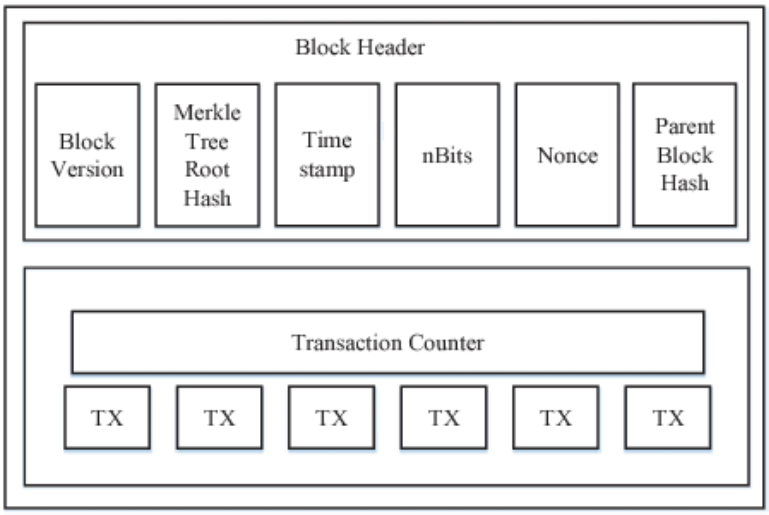

Figure 1 General architecture block implementation

A few modifications have been apported to keep the implementation to a medium level of difficulty while still retaining all the components needed to provide the blockchain's functionalities. For this reason, it was decided to remove the Merkle Tree Root Hash from the implementation because of their purpose and benefit would not be evident in the chosen implementation. As explained in the Bitcoin whitepaper[2], Merkle Trees[5] are very useful in authenticating set of data where the constraint is the large size of the data to be verified and to additionally reclaim disk space. This would not be a restriction on the blockchain implemented because only one transaction per block is added, compared to the 3000 in bitcoin blockchain, because it is wanted to create a simple implementation without adding impractical overheads.

Each block is connected to the previous ones using the hash of its parent block (previous), creating an immutable chain because modifying a single block of the chain would change its hash invalidating the check with the parent hash stored in his child block. In order to modify the chain, all the successive blocks hashes must be recalculated. Below is presented a figure showing the chain structure utilized

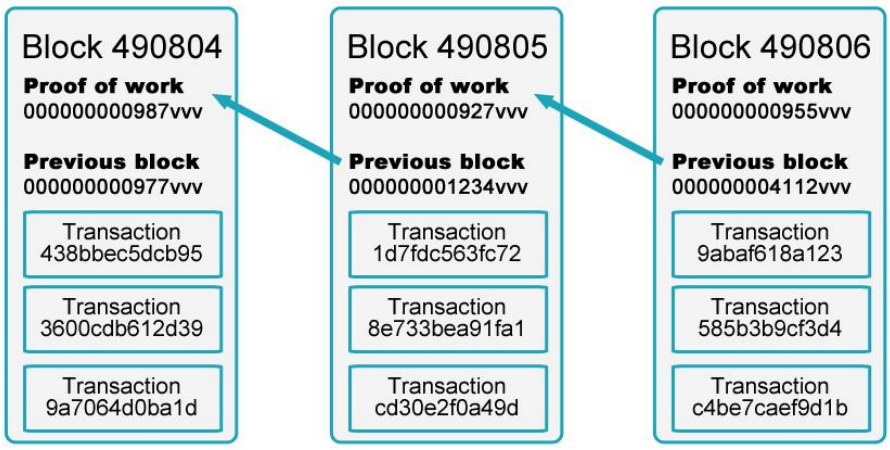

Figure 2 Example chain links hashes

Additionally, the difficulty parameter is stored in the blockchain which is the rule to follow in accepting a valid hash. In a real implementation it is usually the number of leading "0" in the hash, the higher it is, the more difficult is to find it and the more computing power is needed[6].

Each blockchain is managed from an entity called Node which is responsible to give the input to the chain to initiate a mining process. The node itself cannot modify directly internal information belonging to the blockchain, it is a manager of the blockchain which is the only entity equipped with methods to modify itself (mining, modify its difficulty, add blocks, add new transactions). They are implemented as two separated entities to lower the coupling between them and to render the application more flexible to possible changes.

Implemented functionalities

The implementation follows the *general process* presented in the introduction section.

Each block is created adding a transaction from the pool of unconfirmed transactions that is kept from only one selected node of the network. When a transaction is added, all the network will comply with this restriction and check if the actual node is the selected transaction pool, if it is not, the transaction is passed to the right node.

Since the blockchain created is not focused on financial environment, there is no fee to be payed to the node after a successful mine, so some consensus procedures cannot be applied like the proof of stake in which each node has to stake an amount of capital. It was chosen to utilize the proof of work as consensus that involves the application of hash algorithms to find

an hash difficult to create but easy to verify that satisfies certain requirements[7]; it is an easily swappable component and could be used different types of hashes algorithms to learn their speed of performance. Other types of consensus practices could be applied as delegated proof of stake, Ripple, Tendermint, but modification to the general structure of the network would needed to be apported because they need different components block to be run (except of delegate proof of work that is similar as POW)[7].

In a real blockchain, the usual mining is automatically performed by each node and for example in Bitcoin a new block is added every 10 minutes. In the project implementation, it was chosen a different approach in which the mining process is not time-controlled, but it is user-controlled using an interface where a client can request to all the nodes to mine the transaction inputted.

Once the logic of the blockchain was completed, it was needed to create a way to let each node to take part to the network, being able to communicate and share information. It was opted to integrate the blockchain code with web development, setting up each node with a different URL (or port if working locally), using client interfaces to forward requests to each node and run their internal functionalities. Different actions can be requested to the nodes using specific node client routes, enabling communications and concurrent performance.

It is possible that nodes mine a block at the same time, creating two different valid chains and influencing the network with two different chains. This would lead to conflict in the network because of the difference in the stored chains and to solve this problem it was been applied a Fork technique[8] in which the longest valid chain between the two will be chosen as the main

one and the whole network will switch to it. The reason behind this choice is that the longest chain usually means more work done to find the extra hashes and so it is appointed as the accepted one. It is a simple solution, nevertheless it is adopted from real blockchains as Bitcoin, but it can be exploited and create an attack known as selfish mining[9] in which selfish miners keep their mined blocks private, not broadcasting it to the network immediately, only after certain conditions are satisfied. Honest miners might choose to join the selfish miners coalition because of more revenue.

Results

The main advantages of the blockchains advocated by Zheng et al[4] (decentralization, persistency, anonymity and auditability) were mostly all achieved in the blockchain implementation. Decentralization is achieved creating the network of nodes without having a central node that holds the correct blockchain; every node has its own correct blockchain that is validated by the whole network, not from a single entity. Persistency was achieved using the hashing values of the previous block in the chain: it is difficult for a node to tamper the blockchain because it would need to modify the whole chain. Auditability of the transactions was partially achieved because they can easily be tracked and viewed in the chain, but there is not the functionality of verification their validity from each node. Anonymity was not directly implemented, except the absence of names using instead IP addresses for each node and is discussed in the next section.

During the program development, it was noted that the principal difficulties are related to the creation of the network of nodes. The logic applied by each node to mine and manipulate their blocks and chain is straightforward, but on the other hand, the communication between nodes it was revealed intricate, especially because connections can be initiated at any time so it must be taken into account when designing the nodes activities. This led to a few situations that need to be addressed as for example a node is mining a block and a second one contacts it, providing its mined block.

Another difficult task to be perfected is to guarantee the absence of exploitable node' functionalities. This problem is also present in the development of smart contracts which are an extension of blockchain functionality as it is explained by Saad et.al[10] in their paper that explores the possible attacks appliable to blockchains, useful to have an overview of the exploits available. The application needs to be thoroughly tested in various situations and parameters setting, to make sure that a node cannot take advantage of some internal actions, for example if enough check are put in place it could replace the entire chain with a customized one which has its chosen transactions in it, destabilizing the network. Care must also be taken when a node is accepting an external mined block, always assuming that the block could be tampered, so it needs to be fully tested in order to guarantee the correctness and safety when accepting it.

It was noticed that the blockchain application can itself guarantee safety thanks to its structure that makes it very difficult to tamper the chain. This is enhanced by the higher difficulty is utilized in the chain, in fact a node would need in average to put more work to tamper data for high difficulties setting, increasing the impracticality of tampering data and

discouraging it, which is the basic concept that a blockchain using proof of work is based upon[7].

Discussion

The blockchain created is missing communication encryption because its purpose is purely educational. If it would be a real implementation to be used in industry, encryption is a must needed feature because anonymity is one of the fundamental purposes of blockchain. As explained by the profiles addresses are computed from public keys of the user reducing any information that could lead to their identity revelation. Additionally, a combination of public and private keys are used to validate transactions and send them encrypted and check if the transaction has been illegally modified by a third party during the transmission. In the project implemented, it was not used this level of security, in fact the transactions are publicly available to any node and are not encrypted during their transmission[3]. This is surely an improvement that could be applied to the application in a future work, creating a more secure environment for transactions transmission.

Lastly, each node being a single entity working locally, once they share their results, it must be created functionalities that will ensure the adaptation of every node's internal parameters in order to let them output results that are acceptable from the whole network and not only the single node. For example, once a node mines a block, the transactions used must be taken off the whole network, so any other node will not use them in a later block. An other example

would be if the difficulty is updated, the whole network nodes will have to update their internal difficulty.

The final implementation can be argued that is a very simplified view of a real blockchain, but the purpose was to create a project with the main components of any blockchain and the presented project correctly gives a general overview on the structure of a blockchain, implementing the essential features of a public ledger.

Acknowledgements

Thanks to the great many articles in this field that helped in the discovery of blockchain implementation.

References

1. Bitcoin price, charts, market cap, and other metrics. CoinMarketCap. https://coinmarketcap.com/currencies/bitcoin/. Accessed December 7, 2019.
2. Nakamoto S, others. Bitcoin: A peer-to-peer electronic cash system. 2008.
3. Tschorsch F, Scheuermann B. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. *IEEE Commun Surv Tutor*. 2016;18(3):2084-2123. doi:10.1109/COMST.2016.2535718
4. Zheng Z, Xie S, Dai H, Chen X, Wang H. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In: *2017 IEEE International Congress on Big Data (BigData Congress)*. ; 2017:557-564. doi:10.1109/BigDataCongress.2017.85
5. Merkle RC. Protocols for Public Key Cryptosystems. In: *1980 IEEE Symposium on Security and Privacy*. ; 1980:122-122. doi:10.1109/SP.1980.10006

6. Christidis K, Devetsikiotis M. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*. 2016;4:2292-2303. doi:10.1109/ACCESS.2016.2566339

7. Bach LM, Mihaljevic B, Zagar M. Comparative analysis of blockchain consensus algorithms. In: *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. ; 2018:1545-1550. doi:10.23919/MIPRO.2018.8400278

8. Mukhopadhyay U, Skjellum A, Hambolu O, Oakley J, Yu L, Brooks R. A brief survey of Cryptocurrency systems. In: *2016 14th Annual Conference on Privacy, Security and Trust (PST)*. ; 2016:745-752. doi:10.1109/PST.2016.7906988

9. Eyal I, Sirer EG. Majority is not enough: Bitcoin mining is vulnerable. *Commun ACM*. 2018;61(7):95–102.

10. Saad M, Spaulding J, Njilla L, et al. Exploring the Attack Surface of Blockchain: A Systematic Overview. *ArXiv190403487 Cs*. April 2019. http://arxiv.org/abs/1904.03487. Accessed October 29, 2019.

Figure legends