IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 8, Issue 8, August 2019

Stabilize IoT Blockchain using Smart Rewarding Mechanism: Incremental Block Reward

Bahaedinne Jlassi[1], Liuyang Ren[2], Scott Chen[3], Fehri Bilel[4], Ahmad S. Omar[5]

MASc, Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada[1]

PhD Candidates, Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada[2,5]

Professor, Electrical and Computer Engineering, Conestoga College, Cambridge, Canada[3]

PhD, Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada[4]

Abstract: Bitcoin's block reward is dependent on miner hardware profiles such as computation capacity and mining hardware identities, therefore, incentivizing miners to upgrade mining rigs in order to mine more blocks and maximize the payoff. As a result, miners who refuse to upgrade mining hardware suffer decreasing payoff because the mining difficulty increases automatically as the computation power of the entire network grows. However, this phenomenon is highly counter-productive in Internet-of-Things (*IoT*) networks, where *CPU*s are embedded into various devices and hardware upgrading require non-trivial efforts, in addition to the constrained computational and power capabilities. As a result, we propose the Incremental Block Reward (*IBR)* to encourage miners to reuse mining hardware, thereby helping stabilize the network, reduce energy consumption, and encourage new miners (i.e. more devices) to join. The simulation results show *IBR* facilitates even wealth distribution and promotes more fair and long-term mining establishment.

Keywords: Internet-of-Things, Blockchain, Mining Reward, Hardware Upgrade, Monte-Carlo Simulation

I. INTRODUCTION

In Bitcoin, different miners receive the same amount of reward after mining a block based on the allocated computation power which usually drives frequent upgrade to the mining hardware. Consequently, miners periodically buy more powerful mining rigs to compete with each other. In game theory terminology, there is no equilibrium because every miner tends to upgrade hardware to compete with other miners did so. On the contrary, if miners stop investing in new hardware, mining difficulty would stay low and miners are able to gain the same amount of rewards. Hardware upgrade benefits a miner for a while, but makes it harder for everyone to make a profit in the long run. This heavily affects the network stability and introduces greedy behaviour in a network that is built to be cooperative.

Bitcoin mining hardware evolved through: *CPU, GPU, FPGA, and ASIC* [3; 4] . Nowadays, Bitcoin mining difficulty has been pumped up to a prohibitive level, making it difficult for a new miner to enter the mining network. If the same rewarding strategy is applied to an *IoT* Blockchain, the network would not only be unstable, but also discouraging to new joining devices [1;2] . Moreover, replacing mining-dedicated *ASIC* frequently (e.g. every six months [3]) is an unpractical and resource-consuming process. Usually the higher computation capacity an *ASIC* owns, the more power-thirsty it is, therefore more unfit to the nature of small IoT devices that are hindered by limited processing and power capabilities. *IoT* devices are sensor and telemetry nodes that are usually contain microcontrollers and microprocessors with limited processing capability, in addition to being battery-powered.

To address the aforementioned problems, the authors propose the Incremental Block Reward (*IBR*) mining approach in order to discourage miners from upgrading mining hardware, thereby helping stabilize the network, reduce energy consumption, and encourage new miners to join. To the best of our knowledge, the concept of incremental reward block is completely novel. To ensure the resistance towards the ongoing hardware race in the cryptocurrency world, the protocol of rewarding new issued block is designed to compensate the miners that keep utilizing the same mining hardware. Thus, if that same machine (*IoT* device) is used to mine a new block, the protocol would increase the reward of this block. The *IBR* approach is proposed since it suits a consensus algorithm targeted to be run on limited capability *IoT* devices that is hard to upgrade especially for remotely deployed nodes.

II. BACKGROUND

A. Bitcoin's Proof-of-Work

In a Proof-of-Work (*PoW*) consensus algorithm, miners compete in solving a hash puzzle for reward. A block is valid only if its header hash is less than or equal to a threshold, which is inversely proportional to mining difficulty. Mining

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 8, Issue 8, August 2019

refers to the process that a miner tries billions of trillions nonce values in order to find a valid block header hash. The mining node that solves the puzzle first obtains the privilege to propose the next block. Therefore, a node with p percent of the overall hash power of the network has the probability of p to find the next block.
Mining a block consists of eight steps:

1. Listen for transactions and validate them.
2. Listen for blocks and validate them.
3. Maintain a replica of block chain, choose the parent block based on the consensus rule, e.g. longest chain or Greedy Heaviest-Observed Sub-Tree (GHOST) [5] ;
4. Assemble a block;
5. Solve the *PoW*;
6. Publish the block to the network;
7. Wait for the block to be buried deep enough in the blockchain;
8. Block reward becomes spendable;

Fig. 1 shows the three adjustable fields in a block: the nonce, timestamp in the block header, and the extraNonce in the coinbase transaction[6] .

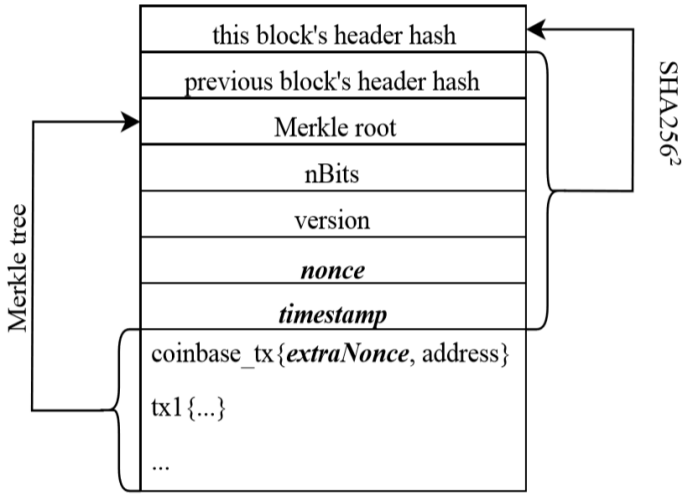

Fig. 1 Block structure with highlighted adjustable fields

The nonce is an arbitrary 32-bit numbers; extraNonce has a variable length of 2 to 100 bytes. Changing extraNonce is much more expensive than changing nonce because the former will propagate through the entire merkle tree up to the merkle root field in the block header.
Timestamp is UNIX timestamp that must be greater than the median of previous 11 block timestamps and less than 2 hours in the future. Thus, the range of the timestamp is roughly 3 hours since, on average, one block comes out per 10 minutes. In other words, a miner can try at most about 10800 timestamps when mining a block.
PoW difficulty is stored in the nBits field in Fig. 1, which is a 32-bit representation of a 256-bit unsigned integer [7] . The nBits value is periodically updated according to block intervals of the past 2016 blocks to ensure the ten minute average block interval. A node calculates the nBits value locally and check the one in a received block against the local version. This block is rejected if the two nBits values do not match. Due to hardware race and network expansion, Bitcoin's difficulty at block height 500,000 is 1.87 trillion times more than that of the genesis block. The hash rate of the entire network is as high as 50 million TH/s [8] .

B. Bitcoin's Rewarding Mechanism

Miners are incentivized by block reward consisting of block subsidy and transaction fees. The first transaction in a block is always a coin base transaction, which credits newly minted Bitcoins (i.e. block subsidy) and collected transaction fees to the miner's account. Block subsidy halves every 210,000 blocks, i.e. four years under the ten-minute average block interval[3] . Block subsidy is 12.5 BTC in 2018.
We do not consider transaction fees in following discussion since it is optionally enforced by Blockchain users. For simplicity, we refer to block subsidy as block reward in other sections of this paper.

C. Intel Software Guard Extensions (SGX)

Intel SGX is a set of extension to Intel *CPU* since processor Core i7 that aims to provide integrity and confidentiality to security-sensitive computation [9] . It sets up a trusted execution environment (TEE), namely an Enclave, where codes

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 8, Issue 8, August 2019

and data loaded inside are protected from reading and writing from outside even by the operating system. *SGX* supports remote attestation by creating quotes—digests of enclaves and digests of output data signed with a unique asymmetric private key. This private key is burned into the processor by Intel during fabrication and thus only accessible to *CPU*. A quote guarantees a remote verifier that a genuine *SGX*-enabled processor is on the other end of the communication channel and the received information is trustworthy [10] .

III. INCREMENTAL BLOCK REWARD (*IBR*)

To our best knowledge, the notion of Incremental Block Reward (*IBR*) presented in this section is a completely new feature. To further increase the resistance against the mining hardware race, the protocol of rewarding newly issued blocks would bias towards the miners who keep their mining operations on the same hardware
IBR compensates miners who stick to old mining hardware by rewarding them more than those who migrate to new hardware. In other words, unlike Bitcoin, *IBR* rewards different miners differently. *IBR* is a two-step protocol: 1) upon receiving a block, a node validates the *CPU* identifier of its miner via remote attestation, and 2) based on the consistency of the *CPU* identifier, calculates the block reward for this specific miner who submitted the block. Fig. 2 illustrates the protocol. All nodes are verifying nodes except for the miner of the new block. It is important to point out that the miner in Fig. 2 also maintains its own UTxO, *CPU:Balance* databases, and the Blockchain so it can be a peer verifier for blocks mined by other nodes, which are not included in this figure to keep the illustration simple.

A. CPU Identifier Verification

A miner retrieves *CPU* identifier via executing system call, e.g. getcpu() in Linux, in an enclave and places it in the block header. A quote (the *CPU* attestation in Fig. 2) is generated by *SGX* to provide integrity, i.e. the miner has not tampered with the result to pretending to use another *CPU*.

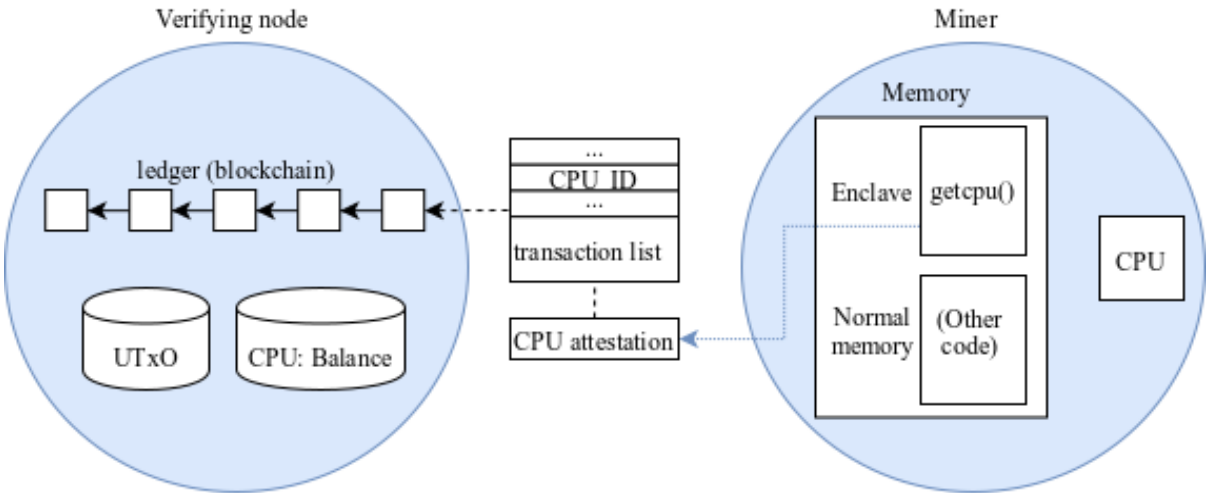

Fig.2 : *IBR* verification protocol

A miner broadcasts their block together with a *CPU* attestation generated by *SGX*. A "verifying node" confirms that the miner did not lie about the *CPU* and calculates the block reward. After the confirmation, the block is added to the local copy of the blockchain if it passes other checks such as transaction validity.

B. Remuneration Mechanism

In *PoW*, remuneration is just an incentive for miners to continuously participate in the consensus process. We assign another goal of encouraging hardware reuse. *CPU* identifiers are used as miner addresses in the new block structure, and each node also maintains a *CPU: Balance* database in addition to the original UT_xO database. The balance of a *CPU ID* indicates the number of times this *CPU* is used to mine blocks.
Like the UT_xO database, *CPU: Balance* database is also derived from the Blockchain and thus consistent among nodes. We use Equation 1 to calculate block reward:

$$R_{ibr} = (1 + B/c) \cdot R \qquad (1)$$

where R_{ibr} is the block reward under *IBR*; *B* is the balance for this *CPU* (account); *c* is constant; *R* is the block reward under Bitcoin's remuneration rule. The optimal methods of keeping a distributed database for *CPU:Balance* data is still under development.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 8, Issue 8, August 2019

C. *Security Analysis*

While miners may attempt forging *CPU* identifiers for personal benefits, the act of pretending to be a new piece of hardware wouldn't be beneficial to miners because they would effectively reset their *IBR* record and the respective reward bonuses. However, pretending to be an old hardware while using a more powerful one is lucrative. This is exactly the reason that *IBR* mandates the action of retrieving *CPU* information to be executed under *SGX*. If a miner pretends to use an old hardware, no valid *CPU* attestation can be generated and hence the block would be rejected by other nodes. The authors also suggest a reputation score to be tied to the remuneration mechanism to help further encourage long-term mining establishments, and prohibit fake *CPU* information. The reputation score, *REP*, is directly proportional to the *Ribr* and inversely proportional to the number of times a detected fake *CPU* information usage is found f_{ID}, where alpha is a system parameter that can be configured to tighten or loosen the relationship between the different factors.

$$REP = alpha.(R_{ibr}/f_{ID}) \qquad (2)$$

IV. SIMULATION

As mentioned in the previous section, the main feature of the new *IBR* protocol is increasing reward to those using the same mining hardware. In order to validate this *IBR* feature, a Monte Carlo simulation is carried out. It is a powerful inference tool widely used as a sampling technique to represent the distribution of the variables under investigation [11] . In our case, we'd like to use this technique to demonstrate how *IBR* protocol would bias the wealth distribution towards miners with consistent hardware profile, and how it would help achieve the overall wealth distribution with greater fairness.

A. *Simulation Setup*

The simulation starts with the initiation of the coin supply and fixing the number of mining nodes. The rewarding protocol as discussed in the previous sections is implemented. Next, winning nodes are picked randomly, and the reward is granted accordingly, as defined by the rewarding protocol. In one instance, all nodes might have the same probability of mining any given block, i.e. all nodes have equi-probable chances of winning a block. In another instance, nodes with higher performance have a higher chance of winning. For each iteration (iteration meaning a block has been mined and the corresponding reward is given), the balances of miners is recorded. At the end of the simulation, the distribution of wealth among different nodes is recorded.

B. *Simulation Result*

In all our simulations, the total number of coins offered is fixed at 10 million. *R* in Equation (1) is set at 100 coins, and the constant c is set at 10,000. We use this baseline to emulate various different market wealth distributions under different market hardware upgrade regulation with and without IBR governance. In order to make sense of the positive effect of *IBR*, we must first examine the most ideal market situation where hardware upgrades are strictly prohibited. As shown in Fig. 3, under this controlled market scenario, an IBR-free algorithm ensures a more even wealth distribution than an *IBR*-enabled algorithm. This may seem counter-intuitive at first, because *IBR* changes the probabilistic distribution of wealth by biasing towards miners with consistent mining hardware.

However, it is important to note that such strictly controlled cryptocurrency market do not existed. In reality, an *IBR*-free algorithm – Bitcoin as a good example – biases wealth towards miners with substantial hardware upgrades, thus skewing the distribution of wealth much more unfairly than the misleadingly ideal *IBR*-disabled curve in Fig. 3.

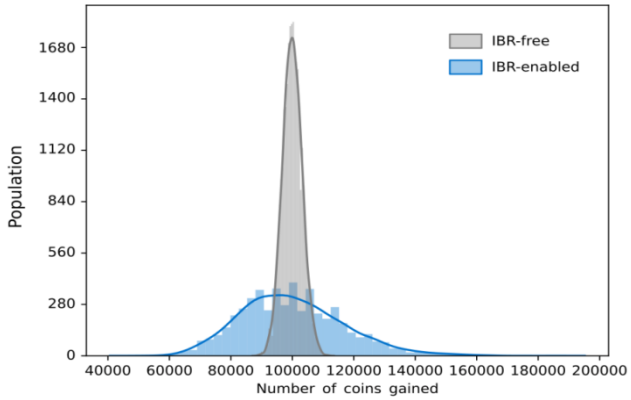

Fig. 3: Wealth distribution when hardware upgrade is strictly prohibited

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 8, Issue 8, August 2019

With Fig. 3 as a theoretical baseline, we now introduce different variations of permitted hardware upgrade into the mining market. Under *IBR*-enabled scenario, miners who are permitted to upgrade their hardware capabilities would give up their *IBR* reward to gain a 200% increase in their mining hardware performance. Fig. 4 illustrates different percentages of miner populations who are allowed for hardware upgrades.

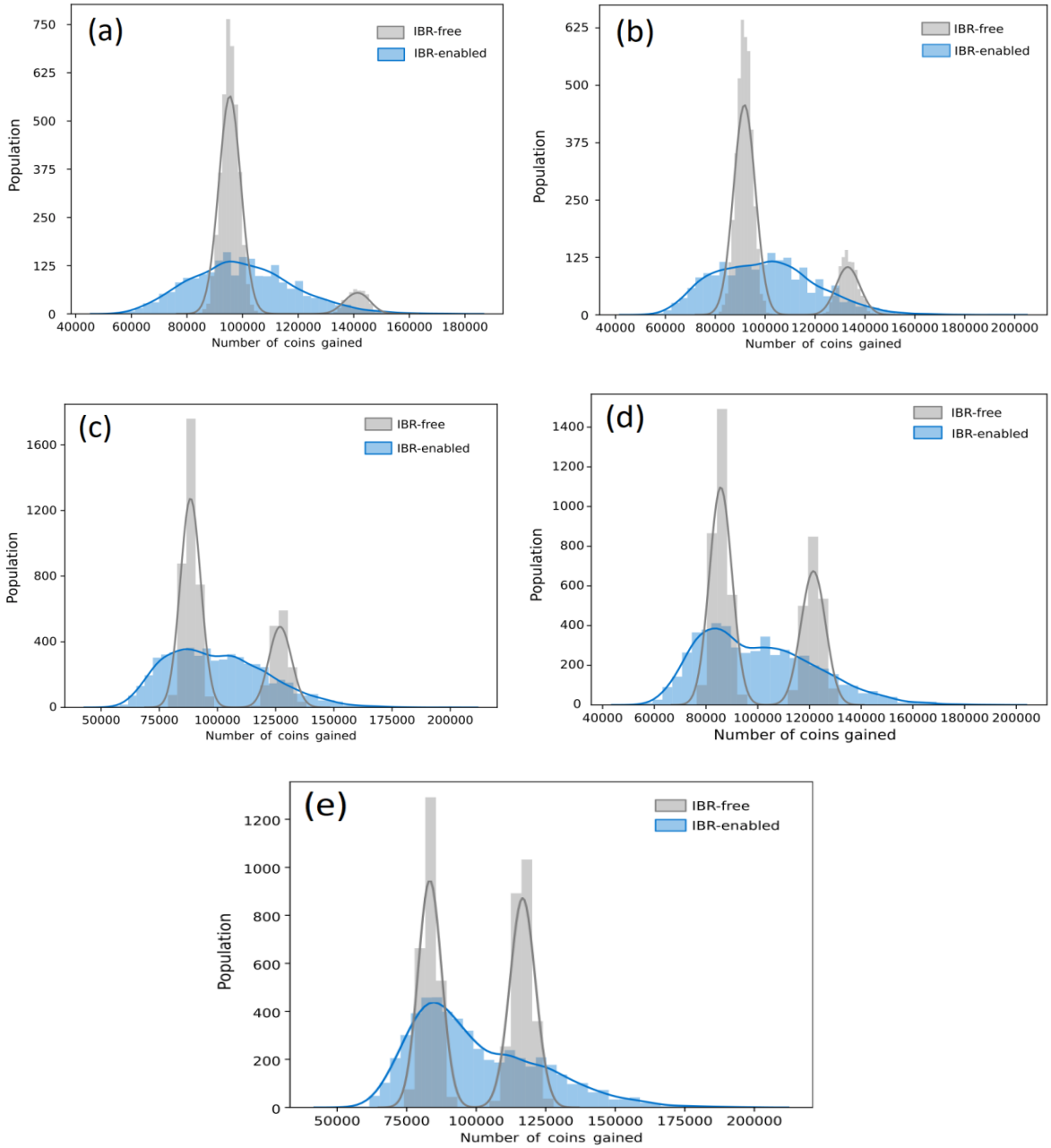

Fig. 4: Five different percentages of miner population who are permitted to upgrade their mining hardware performance by 200%. a) 10%, b) 20%, c) 30%, d) 40%, and e) 50%.

Through Fig. 4, the advantage of *IBR* governance becomes extremely clear. Under an *IBR*-free market, miners who upgraded their hardware capabilities enjoys a drastic increase in wealth, and as the percentage of permitted miners increases, the wealth gained by the restricted miners have been increasingly compressed. This phenomenon models very closely what happens in the real-world cryptocurrency market, but the scale of hardware upgrade in reality is much more out-of-proportion. Those who paid for strong mining hardware enjoy large amounts of coins gained, whereas those with weaker mining hardware would see close-to-zero probability to gain any wealth through mining. Ultimately, all the market monopolization and manipulation seen nowadays in the cryptocurrency market can be attributed to the outrageously uncontrolled hardware upgrade race. On the other hand, under the *IBR*-enabled market, the wealth distribution is consistently maintained in a much more normal distribution as seen in Fig. 3, regardless of

how many miners are permitted to upgrade their hardware. This is because *IBR* opens up a possibility for the free market to find its own wealth distribution balance - through a virtual loyalty program. Miners have two options to guarantee probable wealth accumulation under *IBR*-either stay loyal to the mining hardware, or offset the loss of the loyalty reward by purchasing a more expensive piece of mining hardware. In this sense, *IBR* presents a highly feasible model as regular loyalty reward program commonly available in existing enterprise models, where individuals enjoy an option of gaining advantages without paying more. Thus, we have successfully demonstrated that *IBR* is a highly practical approach to discourage hardware race, thereby achieving a much healthier market wealth distribution with reasonable fairness, and promoting energy conservation in a grander scheme.

V. FUTURE WORK

IBR itself is not pool-resistant. A pool manager could join the blockchain network as an individual node and outsource the *PoW* to a pool of workers behind him. The huge hash power enables a pool to mine more blocks signed by the same processor than small solo miners. Although pool-resistance is out of the range of this paper, the authors suggest a solution by moving *PoW* solving code into enclave in the future.

VI. CONCLUSION

In this paper, we proposed *Incremental Block Reward* (*IBR*) to encourage miners to reuse mining hardware and lessen the effect of high spending on mining hardware for blockchain networks. The simulation results of many different network configurations show that *IBR* reduces the effect of hardware upgrade and thus facilitates a fair distributed, energy efficient and stable blockchain networks.

REFERENCES

[1]. X. Huang, P. Craig, H. Lin, & Z. Yan, "SECIOT: a security framework for the internet of things," Security and communication networks, vol. 9, no. 16, pp. 3083–3094, 2016.
[2]. P. Zhang, Z. Yan, and H. Sun, "A novel architecture based on cloud computing for wireless sensor network," in Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering. Atlantis Press, pp. 472–475, 2013.
[3]. A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.
[4]. M. B. Taylor, "The evolution of bitcoin hardware," Computer, no. 9, pp. 58–66, 2017.
[5]. Y. Sompolinsky and A. Zohar, "Secure High-Rate Transaction Processing in Bitcoin,"
[6]. Bitcoin.org, "Coinbase Input: The Input Of The First Transaction In A Block." *Accessed: 2018-10-18*
[7]. Bitcoin.org, "Target nBits", *Accessed: 2018-10-18*.
[8]. Bitcoin.org "Hash rate." *Accessed: 2018-10-18*.
[9]. V. Costan and S. Devadas, "Intel sgx explained.," IACR Cryptology ePrint Archive, vol. 2016, no. 086, pp. 1–118, 2016.
[10]. F. Schuster, M. Costa, C. Fournet, C. Gkantsidis, M. Peinado, G. Mainar- Ruiz, and M. Russinovich, "Vc3: Trustworthy data analytics in the cloud using sgx," in Security and Privacy (SP), 2015 IEEE Symposium on, pp. 38–54, IEEE, 2015.
[11]. R. Y. Rubinstein and D. P. Kroese, Simulation and the Monte Carlo method, vol. 10. John Wiley & Sons, 2016.

BIOGRAPHY

Bahaedinne Jlassi is the CEO of Demystify. He is a Blockchain technology expert with a vision to popularize it around the world. Baha is an Electrical and Computer Engineer with two master degrees (MASc) from the University of Waterloo and the Superior Institute of Technology in Montreal (ETS). Baha has more than 10 years of experience in both industrial and academic fields. He worked as a lab instructor at the University of Waterloo for years teaching digital electronics, VHDL, FPGA, embedded system design and Physics. As an engineer, he has supervised and managed multiple projects in Electronics, Information Technology and Telecommunication. Baha is a hungry reader, an avid traveler, a fitness passionate, a debutant backpacker, an acceptable cook and an adventure seeker.