IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 15, 2020, accepted July 24, 2020, date of publication August 5, 2020, date of current version August 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3014287

Wi-Fi-Dependent Consensus Mechanism for Constrained Devices Using Blockchain Technology

CHINAZAEKPERE E. NGUBO, (Graduate Student Member, IEEE), AND MISCHA DOHLER, (Fellow, IEEE)

Centre for Telecommunications Research, Department of Engineering, King's College London, London WC2R 2LS, U.K.

Corresponding author: Chinazaekpere E. Ngubo (chinazaekpere.ngubo@kcl.ac.uk)

ABSTRACT Consensus mechanisms are at the core of any Distributed Ledger Technology. They define how transactions are validated and verified, how the network resolves conflicts and protects against malicious intent, and how multiple devices on a distributed network with no central authority arrive at the same global view of the ledger. Experience with the consensus mechanisms employed in popular blockchain platforms have warranted years of research into more scalable and energy-efficient consensus mechanisms for a green society using constrained devices. Most of the ''high-security'' consensus mechanisms require the use of computational resources which are simply not available to devices with a constrained nature. Hence, the presence and activity of these devices do not contribute to the security of the ledger, irrespective of how numerous they are on the network. In this paper, we propose a novel consensus scheme, fully dependent on WiFi technology, suitable for constrained devices. Whilst in traditional Blockchain consensus mechanisms, constrained devices do not add to the security of the ledger, the security of our consensus mechanism increases with the number of connected devices.

INDEX TERMS Blockchain, consensus, CSMA/CA, distributed ledger technology, IoT, WiFi.

I. INTRODUCTION

Blockchain technology is a distributed ledger technology (DLT) which relies on cryptographic proofs to secure the ledger while presenting a consistent, global view of the ledger. As an append-only ledger, transactions in the blockchain network append to the ledger in a deterministic manner, i.e. every node on the network executes the same transactions in the same way and arrives at the same results, ensuring the global view.

In early 2016, Christidis and Devetsikiotis [1] detailed the applications of the Blockchain technology to the Internet-of-Things (IoT) sector, which involved among other things the transfer of digital tokenized assets and the usefulness of smart contracts in IoT communication and services. Authentication of IoT devices and the integrity of the data gathered continue to be a major concern as it raises questions about the privacy of the data and the security of the distributed large scale IoT networks [2]. Today, IoT devices support diverse use cases such as farming, supply chains [3], spectrum management [4], etc. A closer look into Blockchain technology shows that it can also provide solutions for IoT in the areas of *interoperability*, *provenance* and *autonomy via smart contracts*.

The production of IoT devices is not limited to a specific vendor; hence, there are bound to be compatibility issues, drags in software updates and services offered and limitations to the use of specific vendor cloud storage. DLT can bridge this gap as the most prominent implementations of DLT are open source, and device-to-device communication can be facilitated via the overlaying peer-to-peer (p2p) network. As sensors and other data-gathering IoT devices transmit data over the network, either for storage or automated use by other devices, the provenance of each data block must be verifiable. Provenance is an important issue in today's data-driven world and is currently a topic of research. Blocks contain the hashes of every transaction accepted on the Blockchain network; and with simply the Merkle root path, transaction hash and block height, it is possible to trace accepted transactions.

The transactions in the block make reference not only to the actions of the constrained device but also contains a signature, resulting from its private key, ascertaining the source of the data and activity. This level of accountability and shared history not only enhances security and confidence in IoT devices but also promotes autonomy, understanding

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaolong Li.

that IoT devices cannot act beyond the reins of the guiding smart contract. The benefits that arise from adjoining these two technologies have resulted in several research works, presenting surveys and use case-specific implementations.

An essential part of DLT is its consensus mechanism. Consensus mechanisms provide the means by which distributed users might agree on the validity of individual transactions. There are various consensus algorithms in the blockchain community; however, the popular mechanisms require a significant expenditure of computing resources, the use of the native cryptocurrency, selected nodes or the use of an additional external device. A suitable consensus mechanism ensures that an attacker has great difficulty when attempting to tamper with the history of the ledger. Figure 1 depicts how blocks are chained together to form a blockchain, and how any manipulation to parent blocks, causes every child block to become invalid and orphaned. Most IoT architectures that include Blockchain technology bypass the communication link for any purpose related to Blockchain consensus and security. The efficiency of the means of communication plays a massive role in transaction throughput and ledger synchronization. Given that wireless connection enables mobility and requires less infrastructure, it is not out-of-reach to imagine that wireless communication and its constraints factor significantly into Blockchain communications and services. But these consensus mechanisms have their limitations, mainly when used in constrained devices.

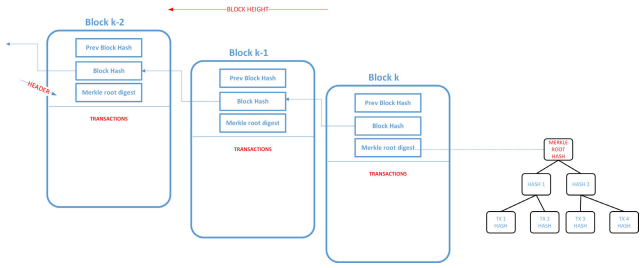

FIGURE 1. Blocks are chained together by citing the header hash of the parent block in the succeeding block header. This ensures that each block can be traced back to the genesis block.

When considering constrained devices and the limitations of their capabilities, DLT has applications that cannot only facilitate processes associated with constrained devices such as authentication and storage but also aid in discovering malicious users within the network. The current means of consensus, however, affects the efficient use of DLT in an IoT ecosystem and also excludes the IoT devices from participating and re-enforcing the security of the ledger. For example, Proof of Work (PoW), in its current form used in Bitcoin is too heavy for constrained devices because of the rate at which the difficulty increases and the computational resource required in hashing a block. Its implementation in Ethereum, referred to as Ethash, is too memory-intensive for constrained devices, as we will show subsequently. Constrained devices are often-battery powered or devices that consume minimal electricity, having limited storage and memory capabilities. Current Blockchain technology offers them no substantial role in the network. Other researchers have noted this limitation and have suggested the use of sink nodes or cluster heads (CH), i.e. nodes in the network with substantially more computational resources to perform the role of a miner. Although, these suggestions have their foundation on known consensus mechanisms, such as PoW.

Constrained devices access the network using wireless links that are low-bandwidth [5]. WiFi technology is the preferred means of connecting devices wirelessly to the existing network infrastructure. The authors in [6] note that there are $\approx$ 432 million access points deployed worldwide, with 623 million predicted for 2023. Also, the recent inclusion of the 6GHz band into the unlicensed spectrum usage for WiFi proves the current level of dependence on IEEE 802.11 technology and its predicted expanse. Hence, expectations lean towards WiFi being the most prominent means of short-range connectivity for constrained devices.

This paper contributes to the growing body of work on Blockchain and IoT in the provision of a lightweight consensus mechanism for constrained devices. Our contribution is two-fold. First, we propose an energy-efficient consensus inclusion scheme for constrained devices based on WiFi technology. Second, we evaluate this consensus scheme, the transaction throughput, bottlenecks and performance against known Blockchain attacks. The motivation for this research stems from a desire to include constrained devices in the consensus process of a Blockchain network, where computing power is of little importance, and given that the estimated number of IoT devices is in its billions, the security of the ledger increases as the number of constrained devices increases.

We also show that not all the stations within the coverage of a WiFi access point necessarily have to be associated with the blockchain to add to the security of the consensus protocol. Some other works of note in the area of Blockchain-IoT communication include an analysis of the communication traffic for Blockchain synchronization of IoT devices by [5], which approached synchronization from the perspective that the IoT devices were to be validators of blocks. Cao *et al.* [7] analyses the impact of CSMA/CA on the performance and security of a DAG-based DLT.

The rest of this paper is structured as follows: Section II provides details of a selection of popular consensus mechanisms, as observed to be mostly used in research related to constrained devices. In Section III, we discuss the problems associated with the discussed consensus mechanisms in relation to constrained devices and then a description of the contention approach of WiFi. In Section V, we cover the scope of our design and any alterations made to the current understanding of Blockchain technology. We evaluate this design in Section VI, matching claims with proofs and pointing out the limitations of the mechanism; furthermore, we carry out analysis towards discovering the vulnerabilities of the mechanism to prevalent blockchain attacks. Finally, we give our concluding remarks and future work in Section VII.

II. POPULAR CONSENSUS SCHEMES

Among the numerous consensus schemes, we only consider four. PoW because it is the most secure and used of all PoW consensus schemes. Platforms like Ethereum are considering moving to PoS because of the economic benefits and higher transaction throughput, so we also consider PoS. And finally, PoA and Tangle because they appear to be the consensus of choice for constrained devices, as opposed to the aforementioned two.

A. PROOF OF WORK, PoW

The proof of work mining mechanism is a gamified cryptographic principle. The principle is that nodes with an intention to mine blocks and obtain the rewards associated with the blocks and its content must solve a puzzle and achieve the desired result. The nature of the problem is *real-time* i.e. miners become aware of the elements of the puzzle when the parts that make up the puzzle (the block contents) are revealed on the network. This scheme greatly reduces the chances a node can pre-mine blocks for a later date. The ledger remains valid as long as honest nodes own the majority of the hash power in the network.

To solve this puzzle, a node must obtain the Merkle root hash of a select group of transactions from its *txpool* and proceed to obtain a 64-bit value, the *nonce*, B_{no}, which when combined with the mixhash, $B_m x$, is confirmation that the minimum required work has been carried out for the creation of the block. There are various implementations of the Dagger-Hashimoto proof of work, such as Ethash, which is used in Ethereum, although it varies in multiple ways. However, Ethash is the PoW consensus scheme described here for PoW.

Ethash uses a memory-intensive scheme to avoid the known pitfalls of Bitcoin mining. Bitcoin miners are known to use specialised hardware such as Application Specific Integrated Circuits (ASIC) which would give them not only a competitive edge over other miners but also significant control over the ledger. Ethash's memory-intensive mining ensures the process of generating a nonce cannot be done simultaneously for multiple blocks; therefore, making mining ASIC-resistant. From [8], the PoW function can be expressed as

$$\mathrm{mx} = \mathrm{B_m} \wedge \mathrm{no} \leq \frac{2^{256}}{\mathrm{B_d}}, \tag{1}$$

with $(\mathrm{mx}, \mathrm{no}) = \mathrm{PoW}(\mathrm{B_z}, \mathrm{B_{no}}, \mathrm{d})$, where $\mathrm{B_d}$ is the difficulty value of the new block, $\mathrm{B_z}$ is the block header void of the nonce and d is a data set for computing the *mixhash*. With all these information available in the block header, the block header alone is suitable for validating the work done on the block.

The mining rewards associated with blocks has inspired various means by which users might amass hashing power. Since the inception of Bitcoin, mining has been done via CPU, ASIC, GPU, an aggregation of mining resources known as mining pools and sale of mining power via websites such as NiceHash [9]. As of early 2019, [10], [11] noted that a mining pool in China known as "BTC.TOP mining pool" acquired over 50% of the mining power in the network. In 2014, some members of the "Ghash.io" mining pool left because the aggregate mining power was getting close to achieving a 51% ownership of the mining power. The frequency of these situations calls for a reanalysis of the security of the PoW algorithm.

Although better equipment with higher hashing power are used for competitive mining on the Blockchain, these equipment are far from energy-efficient. According to [12], in 2019, Bitcoin, a network of 9315 connected nodes [13], consumed more electricity than the country of New Zealand. [14] notes that for Ethereum, the cost of a single transaction is 34KWh. This value is alarming when considering that if the Ethereum mainnet generates a block every 12 seconds, then in the 10 minutes required to produce 1 Bitcoin block, Ethereum produces 50 blocks, howbeit with significantly greater difficulty. The energy consumed in mining has limited the scalability of the network and has increased the expectations of Casper, Ethereum's Proof of Stake consensus mechanism.

B. PROOF OF STAKE, PoS

The first significant alternative to combat the high energy expenditure in PoW was PoS. Where PoW depended on a provable difficulty, PoS relies on the ownership of cryptocurrency within the network. A user had first to own the native cryptocurrency of the platform and then lock them as a proof for mining. Further implementations of PoS factored in the age of the coin, i.e. how long an account had held the coins to be staked. It was observed that miners could go offline for extended periods and return to the network to mine blocks when the coinage had grown significantly. This was found to not only reduce the security of the ledger but also an attacker could go offline for a long time and return to maliciously change the state of the ledger, having a substantial coinage value. This resulted in proof of activity, which proves that a coin holder has been active in the network and not simply been offline. The transaction throughput of PoS is more significant than that of PoW since its block production does not require high-energy expenditure. PoS is built on proof of equity, i.e. only nodes which have a substantial stake in the network can submit blocks to the network, given that they have the most incentive to be honest.

In (2), $B_p h$ refers to the hash of the previous block, A, the account address, t is the timestamp in UTC, bal(A) references the balance of account A and d is the current difficulty [15].

$$\mathrm{hash}(\mathrm{hash}(\mathrm{B_p h}), \mathrm{A}, \mathrm{t}) \leq \mathrm{bal(A)d} \tag{2}$$

C. PROOF OF AUTHORITY, PoA

While PoW and PoS are designed to combat the attempts of anonymous malicious nodes in public networks, PoA is designed for permissioned networks where the identity of the validators are known. PoA is mostly applied in permissioned

TABLE 1. High-level comparison of popular consensus mechanisms.

Feature	PoW	PoS	PoA	Tangle
Node identity	anonymous	anonymous	anonymous, miners(required)	required
Resource required	Energy	Native token	Energy	none
Network adversary attack (%)	< 51	< 51	< 51	n/a
Most popular platform	Bitcoin	Peercoin	Rinkeby	IOTA
Transactions/second	3-20	25	-	250
Cost Efficency & Performance	Low	Medium	Medium	High

ledger where the identity of the validators must be known. Each round an elected participant mines and produces new blocks. A majority of the authority nodes are required to sign.

The authors of [16] note that PoA is as secure as PoW, while not consuming as much resource in computationally intensive work. PoA systems run with the assumption that $n/2 + 1$ authority nodes are honest, n being the total amount of authority nodes.

D. TANGLE

Tangle is a Directed Acyclic Graph (DAG) based DLT. Being DAG-based, transactions are seen as *vertices* of the graph, and each transaction references two transactions. To reference transactions is to approve the transaction. Unapproved transactions are referred to as *tips*, and each tip is selected for approval based on a weighted random value. A node desiring to submit a transaction to the network, after signing it with its private key must reference two tips, thereby approving them and carry out a PoW calculation. Although PoW is considered resource-intensive, it prevents spam attacks. However, Tangle can achieve high transaction throughput while providing fee-less transactions. The consensus is probabilistic and the confirmation confidence of the transaction gives the acceptance rate of each transaction.

There are other consensus mechanisms suggested for constrained devices such as *Proof of Authentication* [17] which involves the use of fog computing; Proof of Device (PoD), Practical Byzantine Fault Tolerance (PBFT), Proof of Elapsed Time (PoET), Delegated Proof of Stake (DPoS), Proof of Space (PoSp), Proof of Storage (PoSt), Proof of Importance (PoI), etc.

III. CURRENT APPROACH TO CONSENSUS AND CONTENTION

The concept of the Internet-of-Things (IoT) is underpinned by sensors and constrained computing units for data collection and increased service provisioning. Examples of such devices are Raspberry Pi, temperature sensors, pressure sensors, smart light bulbs, etc. These devices are not built with the computational resources to be general-purpose devices. Still, they are equipped with specific tools and calculated computational resources that would enable them to carry out the set tasks. Most IoT implementations in, for example, smart cities and smart homes require that these devices operate over a long period on battery power. Hence, they are designed to be energy efficient.

A. EMBEDDED CONSENSUS PROBLEM

The consensus problem arises from two major features of the blockchain:

1) KNOWLEDGE OF THE LEDGER STATES

Consensus, as stated earlier, is the process by which the nodes in the network validate the rights of an account to perform an action that changes the known state of the ledger. It is also the process by which the nodes verify the correctness of the transactions/blocks they have received. Although light nodes are seen and recognized on the network, they do not add to the security of the ledger as they do not hold locally, the full ledger state information by which to make judgments on fraudulent activities. Most implementations of constrained devices in the DLT space assign these devices as either light nodes nodes due to their limited storage space, hence limiting them from actively participating in consensus and not increasing the security of the distributed ledger. This means, if 99% of a network had thousands of interconnected IoT devices as light nodes, an attacker would only have to dominate and re-write the states of 1% of the nodes on the network.

2) RESOURCES FOR CONSTRAINED DEVICES

While traditional distributed architectures rely on centralized management protocols and pools to make decisions and present the global acceptable view on the states, the consensus mechanisms in DLT perform the same tasks but with increasing computational difficulty and resource strain. Most consensus algorithms designed to combat the limitations of PoW and be more energy-efficient, still require an investment of computational resources which are overpowering for constrained devices. The necessary computational investment for PoW has rendered it unusable in constrained networks, primarily due to the increase in difficulty experienced every 2016 blocks. The process of mining is not only beneficial, at best, if the miner mines on the right fork of the chain. The computational effort required is not merely in finding a value based on the difficulty target but also the denial of sleep due to new transactions/blocks advertised frequently. Furthermore, significant memory of the device is occupied by transactions in the *txpool* or *mempool*. This limits constrained devices from being nodes on the network and further restricts them from being validators on the network. However, given the right use-case and additional resources, we do note there might be an argument for mining, where the rewards associated with mining balance or match the consequences.

It is widely known that if an attacker can amass > 51% of the hashing power in the network, the attacker can successfully attack the network to a degree of α, where α is defined as the $\frac{\text{miningRate}}{\text{blockTime}}$. If a miner does not have sufficient computing resources to gain the advantage in the network, he/she can buy additional hashing power from websites such as [9]. This advantage of mining and the accumulation of mining power not only gives non-resource constrained devices an edge in a PoW based network but also ensures that constrained devices cannot participate nor add to the security of the ledger.

PoS has limited usage in an IoT use case, especially when there is a large amount of IoT devices. For example, in a PoS environment where the native tokens are the acceptable stake, the inclusion of a set of constrained devices in the mining process would require that set be funded with an adequate amount of tokens to lock for the mining process. Furthermore, if coinage is used in the network, the chances of a constrained device to mine blocks resulting from its local cluster would be minuscule as its coins would no longer have a substantial age to compete on the network, after an initial blocking.

The use of *PoA* for IoT devices certainly has its appeals, as there is a significant reduction in the amount of energy spent. Also, an administrator can be assured that only verified nodes are allowed to mine. However, a traditional PoA consensus mechanism requires that a block be verified by more than half of the validators. Depending on the available bandwidth, the proximity of the validators in the p2p network and the delay-sensitive nature of the data contained within the blocks, PoA can be time-consuming.

Tangle is a consensus mechanism designed for IoT devices. It also has its limitations seen its reliance on PoW and the requirement to run multiple times the *tip selection algorithm*. The tip selection algorithm is used to determine the probability that any given transaction would have significant weight, i.e. the work associated with verifying the transaction would not be wasted.

B. CURRENT WIFI CONTENTION APPROACH

To understand subsequent protocol modifications, we now describe the Wifi contention approach. Notably, we focus on the communication structure in an Infrastructure Basic Service Set. We also discuss the key features of WLAN that are instrumental for the consensus mechanism.

Generally, multiple access techniques are divided into three types: contention systems, minimal contention systems and contention-free systems. In contention systems, the ready-to-transmit device transmits the data frame independently of the actions of other stations using the medium. In contention-free systems, all stations are cognisant of the transmission of other stations and hence avoid collisions [18]. While in minimal contention systems, the ready-to-transmit station takes some steps to prevent/reduce the chances of collisions such as reserving the medium for the needed time.

Most devices connected to a WiFi access point use the default Carrier Sense Multiple Access-Collision Avoidance (CSMA/CA) mechanism to transmit and receive frames over the air. This scheme requires devices to listen-before-talk, i.e. observe the channel to detect the presence of on-going transmissions. To manage the wireless medium, there are a number of inter-frame spaces which are used, a backoff counter and a contention window. If the contention scheme of CSMA/CA can be implemented securely, the resulting values could be used to facilitate a proof of work contention scheme, but unlike that seen in Bitcoin and Ethereum.

In our proposed solution, the AP is the miner and, basing the "work" required for consensus on the contention experienced by stations, the AP can produce blocks without expending huge amounts of energy in the process. Due to the use of random backoff values, the contention process in a WLAN is unique and the backoff experienced by any station, in particular, cannot be known prior to joining the Basic Service Set (BSS).

When a station has frames to transmit, it senses the channel, and if the channel is sensed to be free for one DIFS period, it transmits. It waits for a SIFS period, and if it receives an ACK after this period, the frame(s) was transmitted successfully. If the station wishes to transmit again, it listens for a DIFS + random backoff time, to avoid channel capture. Following each IDLE DIFS period are slots, with the slot time represented as β, and transmissions are only allowed at the start of a slot. β is dependent on the PHY layer and is based on the amount of time the station would need to sense transmission on the channel, and also inform the MAC sublayer of the channel state. For the initial transmission, the backoff timer is set to a value within (0, CW-1), where the contention window (CW) is initially set to the minimum contention window value CW_{min}. Collisions may occur when there are packets from multiple stations transmitted within the same β.

In CSMA/CA, the backoff scheme is exponential, meaning that the value of CW is doubled in the event of an increase. As stated earlier, CW for an initial transmission of a frame is set to the CW_{min}. If an ACK is not received, the station assumes a collision and doubles the CW value. For every unsuccessful transmission, this field is doubled until the CW_{max} value is reached, where $\text{CW}_{\text{max}} = 2^{\omega}\text{CW}_{\text{min}}$. During a re-transmission attempt, a station firstly decrements the backoff counter, as long as the medium is sensed to be idle, i.e. the countdown is paused when the channel state is busy and continued when idle. Upon successfully counting down, the station further senses the medium for a DIFS period before re-transmitting. In the event of a successful re-transmission, the backoff time is reset to the CW_{min}.

The most common deployments of CSMA/CA use the optional Request-to-Send (RTS)/Clear-to-Send (CTS) control frames, as they are known to offer better throughput by having fewer collisions of data frames. Figure 2 depicts the RTS/CTS four-way handshake. There are other frames such as data frames and management frames, but for the scope of this research, we focus on the control frames RTS/CTS. The value in the duration field of the frame is calculated as

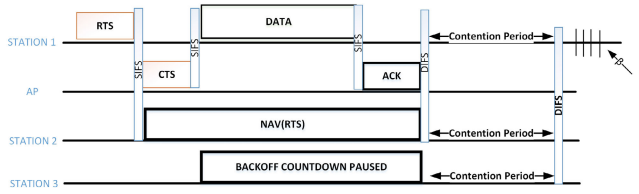

FIGURE 2. High-level overview of the Wifi IEEE 802.11 RTS/CTS Medium Access mechanism.

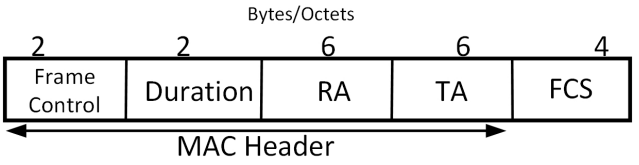

FIGURE 3. RTS control frame.

follows:

$$\mathrm{RTS_d} = \mathrm{SIFS} + \mathrm{CTS} + \mathrm{SIFS} + \mathrm{DATA} + \mathrm{SIFS} + \mathrm{ACK} \quad (3)$$

SIFS, DIFS, ACK, DATA mean Short InterFrame Space, DCF InterFrame Space, Acknowledgement and the frame payload, respectively. As shown in Figure 4, the *Retry* sub-field holds only 1 bit, which is used to indicate if the frame is a result of a re-transmission or not. There is one major cause for frame re-transmission: no-ACK. The distributed nature of the WLAN architecture does not provide global scope to all nodes; for example, in the event of a collision, not all stations are aware of the frames that collided, their size, senders or receivers. The stations are simply aware of the state of the channel, as perceived through physical carrier sense. The value in this field is then vital when determining not only the throughput and collision rate of a WLAN but also in our proposed consensus, as will be demonstrated in later sections.

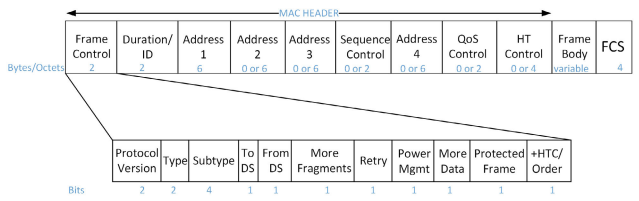

FIGURE 4. MAC frame format [19].

To foster adaptation, we have not made any changes to the IEEE 802.11 standard. Currently, there are no commercial-off-the-shelf WiFi access points that ship with built-in API for accessing any Blockchain platform. This is due to a lack of uniformity in the available Blockchain Technology platforms. Hence, lightweight packages must be installed in the device, acting as the access point.

IV. WiFi DEPENDENT CONSENSUS DESIGN

Blockchain technology, as it is currently known, is built on the chaining of blocks via the use of hashes; we thus lay out a WiFi-dependent architecture where transactions are "blocked". From this point, our understanding of Blockchain technology is built on the workings of the Ethereum blockchain. The core aim of this research is to produce a consensus mechanism where computing power plays no significant role. This aim is to be achieved by taking into consideration specific variables that can be obtained from the WLAN environment and used in conjunction with other values for the proving of individual actions.

A. BLOCKCHAIN CONNECTION DESIGN

The storage and network costs of hosting a local node connected to the Blockchain are quite high, due to the constant growth of the ledger and the bandwidth and data strains on the network. Platforms such as Infura allow users to connect to a remote node rather than bear the costs mentioned above. Here, we adopt a similar approach, but where the local node is hosted on the access point, and the stations access the Blockchain by connecting via Remote Procedure Call (RPC) connection to the ledger on the access point; see Figure 5. This single hop network does not introduce any additional delays as would be experienced in multi-hop networks. This setup allows the AP to filter packets meant for the ledger network from other packets.

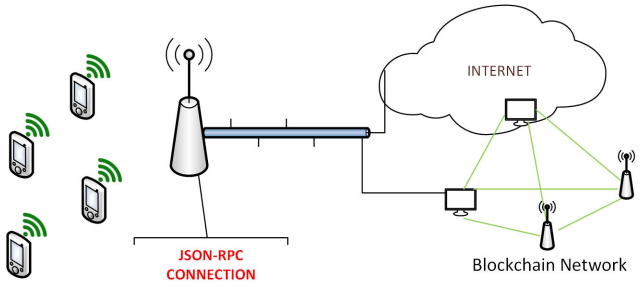

FIGURE 5. JSON-RPC connection from station to Blockchain network. The seconds output link from the AP is for non-Blockchain communication.

A novel approach to our consensus scheme is seen in the relevance of non-ledger associated stations, in the consensus process. Currently, all blockchain consensus mechanisms draw on information obtained from devices/addresses directly associated with the Blockchain. For example, In Bitcoin and Ethereum, the information used for mining originates from within the ledger network. In PoS, this data is the token stake. PBFT relies on the *orderers* within the network, and Tangle uses IoT devices associated with the DAG network. For Tangle, this means if there are ten constrained devices within a BSS, but only one device is associated with the ledger, then only that node contributes to the security of the ledger. The design below presents an alternate approach.

TABLE 2. Transaction parameters.

Nonce	$\leq 32bytes$	Tx_n
GasPrice	$\leq 32bytes$	Tx_{gp}
GasLimit	$\leq 32bytes$	Tx_{gl}
To	$20bytes$ address	Tx_r
Value	$\leq 32bytes$	Tx_v
Data	$0 - unlimited$	Tx_d
V	$1(usually)$	Tx_{vs}
R	$32bytes$	Tx_{rs}
S	$32bytes$	Tx_{ss}
Total	$\geq 213bytes$	

B. BLOCKCHAIN TRANSACTIONS

The Blockchain can be considered as a transaction-based state machine, i.e. a machine whose states are transition via the use of transactions. Tx denotes a blockchain transaction. A valid transaction has some minimum required fields: Tx_n denotes the transaction nonce, which is a scalar value indicating the number of transactions issued by an account. Tx_{gp} refers to the price of gas offered from every unit of gas. Tx_{gl} refers to an amount in wei,[1] paid upfront, declaring the maximum amount willing to be paid for computation. Tx_r represents the 20-byte address of the receiver. Tx_v is the value field of the transaction, which determines what amount the sender's account balance will decrement and by which the Tx_r's balance will be credited. The signature is given by three values: **v, r, s**, here represented as Tx_{rs}, Tx_{vs} and Tx_{ss}, respectively.

Transactions are of two types: "contract creation transactions" and "others". Where "others" refer to transactions that result in message calls and transfer tokens. The main difference between the two transactions is the presence of either a d or i field, in the transaction, denoted as Tx_d and Tx_i respectively. Tx_i indicates the presence of an *init* field, which contains the EVM code used to create the structure of the contract. Tx_d refers to a *data* field, used to introduce arbitrary data into a transaction or to interact with a contract function. So the construct of a transaction is given by table 2 and the determinant for Tx_r is dependent on if $\mathrm{Tx_r} = \emptyset$.

One of the primary reasons why a transaction might fail is an Out-of-Gas (OOG) exception. This exception implies that the given gasLimit for the transaction is not equivalent to the computation required.

C. BLOCK

We define a block as a container for the block header and a number of transactions associated with the header. We define a block using (4), where B_H, B_T, B_O represent the *block header, transaction root* and refers to the list of *ommers* (previously known to as uncles).

$$\mathrm{B} \equiv (\mathrm{B_H}, \mathrm{B_T}, \mathrm{B_O}) \tag{4}$$

Each Ethereum block has specific properties: B_p refers to the hash of the parent block. B_O. B_c is the *coinbase* address that mined the block and earned the rewards associated with the block. B_R is the root of all the receipts obtained from executing the transactions. B_s is the state root, the hash of the root node of the state trie. B_b is a 2048 bit array bloom filter, resulting from the transactions in the block. 3-bits of the 2048-bit array are used to represent a transaction. A block also has a nonce, B_{no}. In PoW, B_{no} is a value that when combined with B_m proves the validity of the block. The currently known position of a block on the chain is known as the block number B_n, more commonly referred to as *height*. B_d is the difficulty associated with mining the block. B_m is a 256-bit hash used in conjunction with B_no to confirm that a substantial amount of work was carried out in the mining of the block. A timestamp value B_t is a Unix time representation of the moment of the block's creation. B_e refer to the *extraData* field of the block.

[1] *wei* is the smallest denomination of the Ethereum cryptocurrency, *ether*, and it is the value in which *gas* is calculated.

V. PROPOSED CONSENSUS MODEL

As most Blockchain-IoT architectures, we consider two levels of constrained devices, stations and access point (less resource-constrained). Stations send transactions to the Blockchain via the JSON-RPC connection. The access point on receiving the transactions, blocks the transactions. Each transaction is verified based on the following:

- Valid signature (Tx_vs, Tx_rs, Tx_ss)
- $Tx_n \neq 0$
- $Tx_gl \geq g_0$

The verification of transactions is approached traditionally from two perspectives: full nodes and light nodes. Full nodes have a verification advantage because they host a local version of the full chain. However, light nodes are not able to verify all transactions instantly, but the three criteria given above form the base for transaction verification for both instant and post-mining verification.

To block a transaction using information obtained from activity in the BSS is not a trivial process. The contention process in WLAN provides very little information that can be verified by both the AP and the stations, about the nature of the experienced contention. For example, after station A waits for one DIFS plus a random backoff time for the right-to-transmit, only the station is aware of the backoff time, and no other station can verify this. However, specific values in the field of the frame can be used to ascertain if a frame is a result of a retransmission attempt. Figure 6 does not consider collisions, only successful transmissions and retransmissions.

The effect of a transaction on the local state, σ' is given by (5), where η is the transaction-level state transition function. We assume each station has a set of keys from which to sign transactions, and sufficient balance for $\mathrm{Tx_v}$ and G, thus removing the possibility of an OOG occurring.

$$\sigma'_{\mathrm{t+1}} = \eta(\sigma'_{\mathrm{t}}, \mathrm{T_x}) \tag{5}$$

where $\sigma'_{\mathrm{t+1}}$ is the post transaction state. For consistency, we restrict $\mathrm{G_{txdatazero}}$ and $\mathrm{G_{transaction}}$ to 4 and 21000 gas.

$$\mathrm{g_0} \equiv \mathrm{G_{txdatazero}} + \mathrm{G_{transaction}} \tag{6}$$

$$\mathrm{g_r} \equiv \mathrm{Tx_{gl}} - \mathrm{g_0} \tag{7}$$

G is used to pay the AP, where G is $g_0 * Tx_{gp}$. This works beneficial to the AP in a use case where the AP is public, such as a coffee shop or train station, and the AP is compensated for verifying transactions for every transaction processed. This could also be used to introduce competition among public access point providers, where different access point only process transactions with a minimum Tx_{gp}. For example, AP_1 has a minimum Tx_{gp} of 1, AP_2 has a minimum Tx_{gp} of 5, and AP_3 has a minimum Tx_{gp} of 10, which would result in them being paid 21004 wei, 105020 wei, and 210040 wei respectively, per transaction verification. Such a model, in the event of wide-scale adoption, might lead to an eventual disregard of transaction fees.

1) VALUE REPRESENTATION

An obvious strength of our proposed consensus mechanism is that given all stations have frames to transmit and all stations are operating honestly, it remains impossible or difficult to a large degree, to predict which station will seize the medium for the right-to-transmit. The binary values shown in figure 6 are a representation of the contention period winners, when considering three stations plus the AP, within a time, *t*. For the remainder of this paper, we denote this binary representation as $TXOP_t$, where t is a predetermined amount of time. The size of $TXOP_t$ is a direct representation of the number of stations communicating with the AP and the length of t assigned. For example, if there are five stations currently associated with the AP and t is set to 10 milliseconds, then $TXOP_t$ would be the binary representation of the winners of each contention period, within time, t, from the perspective of the AP. Such a value might be represented as [00001 00100 01000 00010].

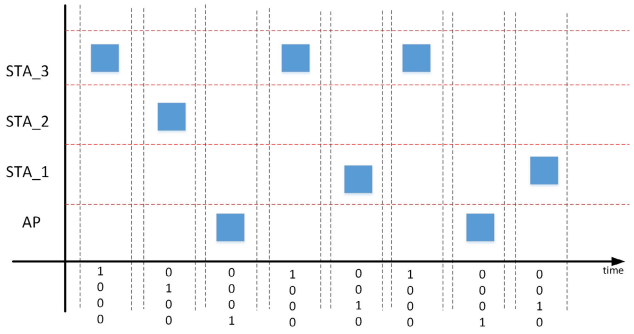

FIGURE 6. Winners of the contention period in BSS within time, *t*.

Previously, we described how frames could collide and need to be re-transmitted, and how this is indicated on the *Retry* field of the frame. To take cognisance of this field, we have included the use of a salt value. The salt value is a binary number which is used to change the value of a field marked as re-transmitted. For example, in figure 6, if the winner of the third column sent a frame, with a "1" in the *retry* field of the frame, the salt is subtracted from the binary value. If the salt is determined to be 1101, it is subtracted from 0001, which using two's complement, results in 10100. This also aids in preventing the desired outcome of an attacker

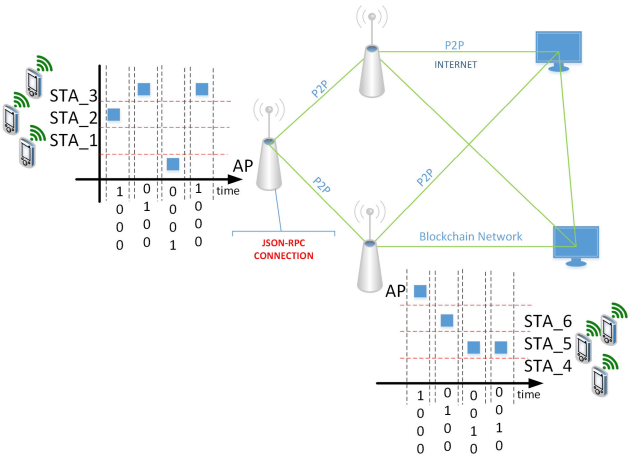

FIGURE 7. Proposed Blockchain Connection among multiple mining access points and the global associated Blockchain network.

when an attacker attempts to seize the medium by brute force, continually sending frames to the AP. If at least one of those frames collide with a frame from another station, the intended value of the $TXOP_t$ will not be achieved.

2) PROPOSED BLOCK HEADER STRUCTURE

The major identifiers of the amount of work spent in mining a block, as seen in Ethereum, is in the *nonce* and the *difficulty* field. Our proposed block header replaces these two fields with *TXOP* and a *Salt*, and is given in table 3.

TABLE 3. Proposed Block Header Structure with included TXOP and salt.

Proposed Block Header Structure
"size": int
"gasUsed": int
"gasLimit": int
"logsBloom": ""
"timestamp": int
"TXOP": Binary TXOP
"salt": Binary Salt
"hash": Block Hex Hash
"miner": Ethereum Address
"number": Block Number (int)
"receiptsRoot": Block Receipt Hash
"sha3Uncles": Hash of uncle Blocks
"transactions": Transaction Hash []
"transactionsRoot": Hash of Transaction Root
"parentHash": Block Hex Hash for previous Block

3) HASHING

The concept of hashing is already in use in Blockchain technologies. A hash function is one which takes data of arbitrary length as input and outputs data of a fixed size. For consistency and ease of implementation, we use keccak-256 as the hashing function in this research. Of course, depending on the use case, any suitable hash function can be used.

VI. EVALUATION

In this section, we evaluate the efficiency of the consensus mechanism and compare it to the PoW. The core aim of

this research is to produce a consensus mechanism where computing power plays no significant role, and this is to be achieved by taking into consideration certain variables that can be obtained from the WLAN environment and be used in conjunction with other values for the proving of certain actions.

In an attempt to observe the impact of Blockchain mining on a Raspberry Pi, we conducted an experiment on a Raspberry Pi 3B+, installing the necessary software required to mine within a permissioned PoW blockchain network. The results show that under no configurations within the confines of the current Ethereum block can a 64-bit SoC @ 1.4GHz, 1GB LPDDR2 SDRAM Raspberry Pi 3B+ mine at least one block with no transactions successfully. We note that other attempts at blockchain mining done on Raspberry Pi involved external devices such as the *Bitmain AntMiner U3* and *LKetc*.

The consistent error encountered is "Failed to generate mapped ethash dataset", and it was due to "fatal error: out of memory". The results show that while only 7.9 of the available memory was used, 100% of the available CPU resource was consumed, which caused the crash. We further set the difficulty value to 0, and the same error was encountered. This sufficiently proves that the Raspberry Pi 3B+, which is one of the higher-powered, resource-constrained devices, is not capable of mining on a PoW blockchain without external assistance.

As stated in the previous section, commercial-off-the-shelf access points are not able to participate in Blockchain lottery-based consensus mining. Hence why most research papers consider the sink nodes or cluster nodes as devices with significantly more computational resources, bypassing the communication link via which the constrained devices communicate their transactions and data to the ledger.

A. DCF PERFORMANCE

Here we present an analytical view of the DCF scheme in CSMA/CA. For simplicity, we assume that the channel is not noisy, and there is no maximum retry limit. In (8), n to represent a fixed set of stations and p is the probability that the frames from a station collide with the frames of at least one of the remaining n-1 stations. Hence, the probability of no collision within a slot, assuming no hidden stations! A successful transmission is then given by $1 - \mathrm{p}$.

$$\mathrm{p} = 1 - (1 - \tau)^{\mathrm{n}-1} \tag{8}$$

Let $\mathrm{i} \in (0, \omega)$ represent the backoff stage. The probability of transmission in a randomly selected time-slot, once the backoff countdown is done, regardless of the backoff stage, where $\mathrm{W} = \mathrm{CW}_{\min}$, is given as:

$$\begin{aligned} \tau &= \sum_{i=0}^{\omega} \mathrm{b}_{\mathrm{i},0} \\ &= \frac{\mathrm{b}_{0,0}}{1-\mathrm{p}} = \frac{2(1-2\mathrm{p})}{(1-2\mathrm{p})(\mathrm{W}+1) + \mathrm{pW}(1-(2\mathrm{p})^{\omega})} \end{aligned} \tag{9}$$

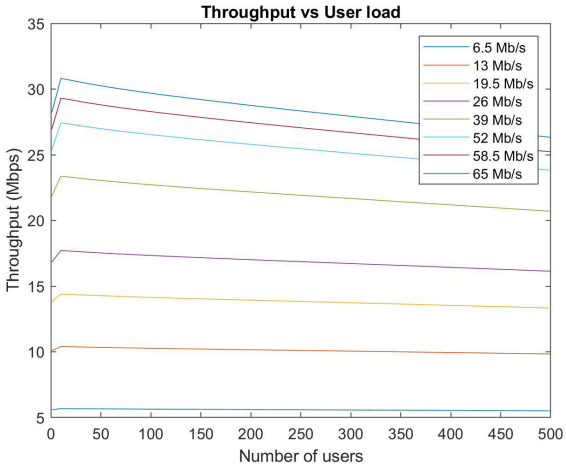

FIGURE 8. Throughput against user load.

With the network being saturated, if we consider that there is at least one transmission in the selected slot, the probability of the transmission, given that each station has an equal probability of τ to transmit, is:

$$\mathrm{P_{tr}} = 1 - (1 - \tau)^{\mathrm{n}} \tag{10}$$

Hence, $\mathrm{P'_{tr}} = 1 - \mathrm{P_{tr}}$ is the probability there is no transmission in the time-slot. The throughput is then considered as the time spent for the payload to move from source to destination stations. This is based on the principle that only one stations transmits within the randomly selected time-slot, and is expressed as:

$$\mathrm{P_S} = \frac{\mathrm{n}\tau(1-\tau)^{\mathrm{n}-1}}{\mathrm{P_{tr}}} = \frac{\mathrm{n}\tau(1-\tau)^{\mathrm{n}-1}}{\tau(1-\tau)^{\mathrm{n}}} \tag{11}$$

Therefore the probability of collision is given as:

$$\mathrm{P_C} = 1 - \mathrm{P_S} \tag{12}$$

We then define $\mathrm{T_C}$ and $\mathrm{T_S}$ as the time the medium is sensed during a collision and a successful transmission, respectively, by the other stations.

$$\begin{cases} T_S = RTS + SIFS + \triangle + CTS + SIFS + \triangle \\ \qquad + H + E[P] + SIFS + \triangle + ACK + DIFS + \triangle \\ T_c = RTS + DIFS + \triangle \end{cases} \tag{13}$$

$\triangle$, $E[P]$ and H are used to indicate the propagation delay, average packet payload size and packet header, respectively. Where H includes the PHY header, $\mathrm{PHY_{HDR}}$ and MAC header $\mathrm{MAC_{HDR}}$. With the length of a time-slot given as β, the throughput of the network which is the fraction of the time to transmit the payload within a slot, compared to the length of the slot [20], can be expressed as:

$$\mathrm{S} = \frac{\mathrm{P_S P_{tr} E[P]}}{\mathrm{P'_{tr}}\beta + \mathrm{P_{tr} P_S T_S} + \mathrm{P_{tr} P_C T_c}} \tag{14}$$

B. CONSENSUS EVALUATION

For the transaction structure, we maintain the same transaction structure, as seen in Ethereum table 2. For the purpose of evaluation, we limit the nature of the transactions to simply the transfer of cryptocurrency, which requires 21000 gas, as stated earlier. Figure 9 provides a possible interaction between stations contending for the channel and if we isolate only successful frames, we can obtain the random binary sequence, which can be hashed with the block header to produce the block hash.

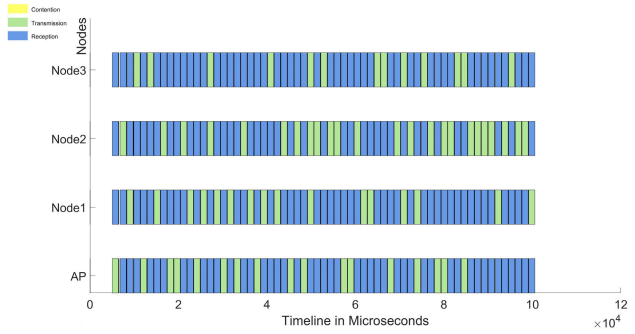

FIGURE 9. MAC frames transmission in CSMA between 3 stations and an AP.

Since we view the WLAN architecture as an IBSS, and we generate the binary data from the perspective of the AP, i.e. all successful transmissions, from and to the AP, we can assume a Bernoulli distribution over a variable array size.

$$P(X = x) = p^x(1-p)^{1-x} \tag{15}$$

Claim 1: The proposed consensus consumes minimal resources.

Proof: As stressed earlier, PoW requires the solving of a puzzle that requires a considerable energy investment, PoS requires the miners to lock tokens for a length of time, and PoA requires the block to be signed by a fraction of the authority nodes. Our proposed consensus mechanism does not require proofs that show the expenditure of electricity, nor the locking of stakes, nor the signatures of multiple validators. Our consensus mechanism is built on the number of stations associated with an access point and the result of the contention over the right-to-transmit.

Claim 2: Higher transaction throughput is achieved without compromising security.

Proof: The popular implementations of PoW such as seen in Bitcoin and Ethereum set the mining difficulty to reflect a block interval that would have a low probability of producing forks, which also places a cap on the transaction throughput of the network. The results obtained from our experiments show that transaction throughput can be measured in milliseconds, which is faster than most consensus mechanisms known to us.

Claim 3: Maintains a level of decentralisation, while not depending on trusted nodes.

Proof: Most suggested consensus mechanisms for constrained devices are dependent on trusted nodes within the network. While our architecture requires that the nodes submitting transactions be connected to a WiFi access point, we do not consider any significant centralisation issues. Decentralisation is still evident in the network, and while the access points carryout out the work of creating proofs on one level of abstraction, any compliant AP can be used. Hence, the AP's are viewed as non-trusted.

C. SIMULATION TEST

We evaluate the above claims via simulation. For this simulation, we use a transaction size of 220bytes, and this transaction size is obtained from the *web3.eth.getRawTransaction()* of a simple 21000 gas *web3.eth.sendTransaction()*. We assume that each node always has a transaction to submit.

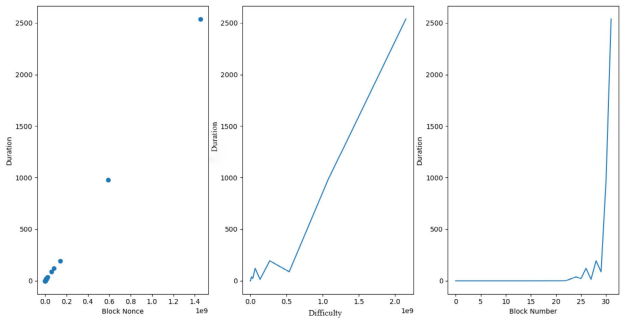

FIGURE 10. PoW mining.

Figure 10 shows PoW mining for an increasing difficulty starting from 2^0 to 2^{32}.

In a use-case where multiple miners are required, the results would be replicated on each miner, demonstrating the resource wasted alluded to in previous sections.

Unfortunately, the time to process the transaction and have it included in a block far exceeds the SIFS period; hence a receipt cannot be sent back with an adjustment made to the ACK frame. Therefore, the time to receive the receipt for a transaction, per station is given as

$$2T_S + M_p \tag{16}$$

D. TRANSACTION THROUGHPUT

Transaction throughput is affected by the constraints of the communication medium. The Transaction per Second (TPS) of the network is calculated based on (17). The TPS is limited by multiple access mechanism in use [7].

$$\text{TPS} = \frac{\text{number of new transactions}}{\text{confirmation delay}} \tag{17}$$

E. MINING CONSTRAINTS

Both reasons for PoW as provided in the yellow paper are covered here:

1) IT SHOULD BE HARD FOR A SINGLE MINER TO MINE BLOCKS, SUCH THAT IT OVERCOMES THE NETWORK

The throughput of the DCF scheme, makes it impossible for a single AP to make outrageously-linear profits as an increase in the number of stations submitting transactions within the BSS (assuming all stations are saturated), will lead to a decline in the performance of CSMA/CA and ultimately the performance of the cluster, as it would lead to multiple retransmissions and if a max-retry is assumed, then an eventual drop of packets.

2) MINING SHOULD BE ACCESSIBLE TO AS MANY USERS AS POSSIBLE

Accessibility is also covered by the millions of access points in the world, which would also generate income from Tx processing.

F. CONFLICT RESOLUTION

Conflict resolution is one of the major functions of consensus mechanisms. For simplicity, we assume that all devices capable of producing blocks have the same level computing resource, i.e. no one device vastly out-powers another.

1) IDENTITY SECURITY

A common practice in the Blockchain community is to have a wallet with numerous addresses and issue transactions with by alternating the originating address. The idea around this is, the activity of a user is difficult to trace and track if the user uses a new address for each transaction. But the model we present appears to defeat the purpose of that level of reasoning. It can be argued that even though a station sends 10 transactions with 10 different source addresses, if the AP is malicious, the AP can always know the 10 transactions originated from the one station, by matching the transactions to both the Source Address (SA) and the Transmitting STA Address (TA) fields of the MAC frame. Hence, reducing the security and privacy of the network.

Fortunately, there are methods by which a device can use a different MAC address for each transmission. By altering not only the MAC address used for each session in the WLAN but also the source address for each Blockchain transaction, a significant level of privacy can be maintained.

2) ATTACKS

This subsection covers some of the known attacks associated with PoW.

a: 51% ATTACK

Needless to say, our architecture does not allow for a 51% attack, as there are no resources required for mining that an attacker can acquire.

b: DOUBLE-SPENDING ATTACK

The double-spending attack is when a malicious account attempts to spend the same Unspent Transaction Output (UTXO) in two different transactions. For example, a node in an attempt to double spend, may connect to a set of peers, publish a transaction, drop all the peers with the hope of connecting to a different set of peers and publish the same transactions. The aim here is to transfer the same set of UTXO to two different addresses. In this scenario, the use of TXOP and salt can be used to quickly validate and communicate blocks, making it difficult to double spend.

VII. CONCLUSION

We have taken measures to evaluate our design and ensure that its performance matches its claims. However, we note that within the Blockchain community, there is a hesitation to adopt consensus mechanisms that rely on information or devices that are external to the Blockchain network, for example, consensus mechanisms that require the use of an external device to provide the necessary proofs for consensus. Still, we propose our consensus mechanisms because the results are promising, and regardless of global acceptance, can be used within the suggested use-case.

The core aim of this research was to produce a consensus mechanism where computing power plays no significant role, and this was achieved by taking into consideration certain variables that can be obtained from the WLAN environment and be used in conjunction with other values for the proving of certain actions. The wealth of information enclosed within IEEE 802.11 frames allows for the expansion of our consensus mechanism to varying degrees. Additional data could be extracted from the other fields of the MAC layer frames such as *Sequence Control, More Fragment, Power Management*, etc.

REFERENCES

[1] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7467408/

[2] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2017, pp. 618–623.

[3] S. Aich, S. Chakraborty, M. Sain, H.-I. Lee, and H.-C. Kim, "A review on benefits of IoT integrated blockchain based supply chain management implementations across different sectors with case study," in *Proc. 21st Int. Conf. Adv. Commun. Technol. (ICACT)*, Feb. 2019, pp. 138–141.

[4] S. Bayhan, A. Zubow, and A. Wolisz, "Spass: Spectrum sensing as a service via smart contracts," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Oct. 2018, pp. 1–10.

[5] P. Danzi, A. E. Kalor, C. Stefanovic, and P. Popovski, "Analysis of the communication traffic for blockchain synchronization of IoT devices," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–7. [Online]. Available: https://ieeexplore.ieee.org/document/8422485/

[6] *Facts and Stats—Wireless Broadband Alliance*. Accessed: Apr. 22, 2020. [Online]. Available: https://worldwifiday.com/about-us/facts/

[7] B. Cao, M. Li, L. Zhang, Y. Li, and M. Peng, "How does CSMA/CA affect the performance and security in wireless blockchain networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4270–4280, Jun. 2020.

[8] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger Eip-150 revision," Ethereum, Tech. Rep., 2019. Accessed: Feb. 12, 2019. [Online]. Available: http://gavwood.com/Paper.pdf

[9] *Hash Power Marketplace | NiceHash*. Accessed: Mar. 20, 2020. [Online]. Available: https://www.nicehash.com/marketplace

[10] *Bitcoin Cash Mining Pool Gains 53% Hashrate and Control; 51% at Risk | Cryptopolitan*. Accessed: May 1, 2020. [Online]. Available: https://www.cryptopolitan.com/bitcoin-cash-mining-pool-gains-53-hashrate-and-control-51-at-risk/
[11] *Single Mining Pool Gains 50% of Bitcoin Cash Hashrate*. Accessed: May 1, 2020. [Online]. Available: https://www.ccn.com/bitcoin-cash-single-mining-pool-controls-half-of-hashrate/
[12] *Bitcoin (BTC) Network Electricity Consumption Chart*. Accessed: May 1, 2020. [Online]. Available: https://www.blockchainanalytics.pro/btc/electricity-consumption/
[13] *Global Bitcoin Nodes Distribution—Bitnodes*. Accessed: Oct. 9, 2019. [Online]. Available: https://bitnodes.earn.com/
[14] *Ethereum Energy Consumption—QAN Blockchain Platform*. Accessed: May 1, 2020. [Online]. Available: https://qanplatform.com/ethereum-energy-consumption/
[15] *Proof of Stake Versus Proof of Work*, BitFury Group, Amsterdam, The Netherlands, 2015, vol. 1, pp. 1–26. [Online]. Available: https://bitfury.com/content/downloads/pos-vs-pow-1.0.2.pdf
[16] Q. He, N. Guan, M. Lv, and W. Yi. *On Consensus Mech. Blockchain/DLT for Internet Things*. Accessed: Feb. 26, 2019. [Online]. Available: http://user.it.uu.se/~yi/pdf-files/2018/sies18.pdf
[17] D. Puthal and S. P. Mohanty, "Proof of authentication: IoT-friendly blockchains," *IEEE Potentials*, vol. 38, no. 1, pp. 26–29, Jan. 2019.
[18] F. Mazda, *Telecommunications Engineer's Reference Book*, 2nd ed, F. Mazda, Ed. London, U.K.: Butterworth-Heinemann Ltd, 1993.
[19] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Standard IEEE Std 802.11-2016 (Revision IEEE Std 802.11-2012), IEEE Computer Society, 2016, p. 3533. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?punumber=7786993
[20] G. Bianchi, "Coordination Function," *Analysis*, vol. 18, no. 3, pp. 535–547, 2000. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=840210

CHINAZAEKPERE E. NGUBO (Graduate Student Member, IEEE) received the B.Sc. degree from Bowen University, Iwo, Nigeria, in 2014, and the M.Sc. degree from Coventry University, U.K., in 2016. He is currently pursuing the Ph.D. degree in telecommunications with King's College London. His current research interests include distributed ledger technology, IEEE802.11, and the Internet of Things.

MISCHA DOHLER (Fellow, IEEE) is currently a Full Professor in wireless communications with King's College London, driving cross-disciplinary research and innovation in technology, sciences, and arts. He is a Fellow of the Royal Academy of Engineering, the Royal Society of Arts (RSA), and the Institution of Engineering and Technology (IET) and a Distinguished Member of Harvard Square Leaders Excellence.