IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 13, 2017, accepted November 18, 2017, date of publication December 6, 2017,
date of current version February 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2779263

Decentralized Consensus for Edge-Centric Internet of Things: A Review, Taxonomy, and Research Issues

KIMCHAI YEOW[1], ABDULLAH GANI[2,3], (Member, IEEE), RAJA WASIM AHMAD[2,4], JOEL J. P. C. RODRIGUES[5,6,7,8] (Senior Member, IEEE), AND KWANGMAN KO[9]

[1]Faculty of Computer Science and Information Technology and Center for Mobile Cloud Computing Research, University of Malaya, Kuala Lumpur 50603, Malaysia
[2]Center for Mobile Cloud Computing Research, University of Malaya, Kuala Lumpur 50603, Malaysia
[3]Academic Sciences Malaysia, Kuala Lumpur 50480, Malaysia
[4]COMSATS Institute of Information Technology, Islamabad 45550, Pakistan
[5]National Institute of Telecommunications, Santa Rita do Sapucaí-MG 37540-000, Brazil
[6]Instituto de Telecomunicações, 1049-001 Lisboa, Portugal
[7]ITMO University, 197101 Saint Petersburg, Russia
[8]University of Fortaleza, Fortaleza-CE 60811-905, Brazil
[9]School of Computer and Information Engineering, Sangji University, Wonju 26338, South Korea

Corresponding author: Kimchai Yeow (yeow@siswa.um.edu.my)

This work was supported in part by the Malaysian Ministry of Education through the High Impact Research Grant of University Malaya under Grant UM.C/625/1/HIR/MOE/FCSIT/03, in part by the Government of Russian Federation under Grant 074-U01, in part by National Funding from *Fundação para a Ciência e a Tecnologia* under Project UID/EEA/500008/2013, in part by Finep with resources from Funttel through the Centro de Referência em Radiocomunicações Project of the Instituto Nacional de Telecomunicações, Brazil, under Grant 01.14.0231.00, in part by the Institute for Information and Communication Technology Promotion grant through the Korea Government under Grant MSIP 2017-0-01705, in part by the Basic Science Research Program through the National Research Foundation of Korea from the Ministry of Science, ICT and Future Planning under Grant 2017030223.

ABSTRACT With the exponential rise in the number of devices, the Internet of Things (IoT) is geared toward edge-centric computing to offer high bandwidth, low latency, and improved connectivity. In contrast, legacy cloud-centric platforms offer deteriorated bandwidth and connectivity that affect the quality of service. Edge-centric Internet of Things-based technologies, such as fog and mist computing, offer distributed and decentralized solutions to resolve the drawbacks of cloud-centric models. However, to foster distributed edge-centric models, a decentralized consensus system is necessary to incentivize all participants to share their edge resources. This paper is motivated by the shortage of comprehensive reviews on decentralized consensus systems for edge-centric Internet of Things that elucidates myriad of consensus facets, such as data structure, scalable consensus ledgers, and transaction models. Decentralized consensus systems adopt either blockchain or blockchainless directed acyclic graph technologies, which serve as immutable public ledgers for transactions. This paper scrutinizes the pros and cons of state-of-the-art decentralized consensus systems. With an extensive literature review and categorization based on existing decentralized consensus systems, we propose a thematic taxonomy. The pivotal features and characteristics associated with existing decentralized consensus systems are analyzed via a comprehensive qualitative investigation. The commonalities and variances among these systems are analyzed using key criteria derived from the presented literature. Finally, several open research issues on decentralized consensus for edge-centric IoT are presented, which should be highlighted regarding centralization risk and deficiencies in blockchain/blockchainless solutions.

INDEX TERMS Blockchain, decentralized consensus systems, directed acyclic graph, edge-centric Internet of Things.

I. INTRODUCTION

Internet of Things (IoT) platforms, such as Azure, AWS, and IBM Watson follows cloud-centric IoT architecture to depict in-cloud analytics via device-to-cloud communication [1]–[3]. Currently, due to their unlimited application, IoT devices are scaling up exponentially.

According to a Gartner and Cisco report, this exponential rise will reach 25 and 50 billion respectively by 2020 [1], [4]. Unfortunately, the cloud-centric IoT provides cloud computing at the far center of the network, which is infeasible in many application scenarios that require optimal latency, bandwidth, and connectivity. For instance, Industrial IoT applications for smart factories, smart farms, smart grids, and smart cities are not well-suited for cloud-centric architectures [1], [4]–[6]. Henceforth, the trend is geared towards edge-centric IoT which provides fog and mist computing in the vicinity [4].

Fog computing extends elastic computation, networking, storage and analytics, across the cloud, to network-edge nodes like gateways and cloudlets [7], [8], whilst mist computing extends even further; beyond the ''fog''; to absolute-edge or endpoints like sensors and actuators [1], [9]. To encourage sharing of edge (e.g., fog and mist) resources and foster a distributed and decentralized infrastructure, a trustless and immutable public ledger based on a decentralized consensus system (DCS) for edge centric IoT is therefore imperative. The design of this public ledger is based on either blockchain or blockchainless directed acyclic graph (DAG) technology to characterize the framework for facilitating transaction processing and coordination between the devices involved. Evidently, there are only three surveys related to DCSs, as discussed in [10]–[12]. In comparing the surveys, [10] only addressed a tool for researchers and practitioners alike for how to develop a taxonomy. Alternatively, [11] and [12] discussed the basic Nakamoto protocol and its applications but overlooked exploring other technologies such as DAG. The survey reported in [11] pertained to broader aspects of technologies and was not specifically related to IoT. Moreover, [12] provided insight into only blockchain and smart-contract technologies for IoT but not on the more realistic and practical edge-centric IoT, which involves M2M micro or nano-payments. Besides, neither [10] nor [12] analyzed state-of-art systems in detail. The main contributions of the present study include:

(i). Present an extensive literature review of state-of-the-art DCSs for edge-centric IoT with their pros and cons.

(ii). Propose and design a thematic taxonomy for DCSs for edge-centric IoT to categorize the literature based upon the common features among these systems.

(iii). Analyze existing methods to highlight the crucial facets and characteristics of edge-centric IoT DCSs. Lastly, some open research issues are put forward.

The organization of the paper is as follows. Section II presents the background with emphasis on recent edge-centric IoT features and decentralized consensus models. Section III illustrates a thematic taxonomy of DCSs for edge-centric IoT. Section IV is a debate on state-of-the-art DCSs and a critical analysis of DCSs. Section V presents a debate on a few compelling open research issues. Section VI concludes the paper by indicating future trends in this research domain.

II. BACKGROUND

This section discusses edge-centric IoT features and decentralized consensus. Throughout this article, we differentiate the keywords ''decentralized consensus'' and ''decentralized consensus system'' (DCS). The former term refers to a consensus protocol and the latter (DCS) refers to its implementation with architectural concerns.

A. EDGE-CENTRIC IoT FEATURES

It is comfortably opined that network-edge (e.g. gateways) and absolute-edge (e.g. sensors) are getting more in their functionality while security ''accidents'' in the cloud will only become more frequent and lethal. Henceforth, the arc of history clearly supports the inevitability of a more autonomous edge-centric IoT, which comprises fog (network-edge) and mist (absolute-edge) computing architectures [1], [13].

1) FOG COMPUTING ARCHITECTURE

Bonomi, Milito, Zhu and Addepalli [7] defined the characteristics of network-edge or fog computing and proved that the fog is an appropriate platform for various IoT applications, including connected vehicles and smart cities. To avoid or mitigate the problems of legacy-cloud/mobile-cloud computing (e.g., high latency, limited QoS, and poor throughput), fog computing is proposed in the context of IoT. The goal of the fog computing paradigm is to shift the typical services of cloud/mobile-cloud computing to the network edge. The elastic computation, storage, and networking across the cloud are extended by fog computing architectures through to the network-edge [7]. Fog computing requires massive geographically distributed implementation. As a result, all entities within a fog-based network are resource-constrained for financial reasons. On the other hand, computing technologies such as batch job processing; are not sensitive to delays and demand more resources. Traditional cloud computing systems are employed to execute these type of jobs more successfully than the emerging fog nodes [1], [7], [13]. Therefore, it is presumed that cloud and fog computing can exist side by side. The target users of both technologies are different but they two complement each other in some scenarios. However, fog computing mainly relies on network-edge servers and gateways, but it is incapable of exploiting resources available in absolute-edge.

2) MIST COMPUTING ARCHITECTURE

This type of architecture extends elastic computation, storage, and networking services through to absolute-edge/endpoints [1], [13]. It extends further than fog computing toward a real time and absolute IoT realm. A novel rule of thumb in IoT system design is to have the sensor data close to its source and avoid sharing across the network except if required unequivocally. This was previously not feasible with 1990s-era endpoint technologies. But today, with better processors, cheaper memory, and networking stacks, the

endpoint can run its own certain analytics and other applications autonomously with real-time query and NoSQL-like file-system support. As a result, it is not necessary to send the data upstream all the way to the cloud. Similar to the fog role with the cloud, when absolute-edge/mist computing is insufficient, network-edge/fog computing technology will be of assistance. Network-edge gateways can vary from very small or simple nodes to very large and powerful computing devices or even just a part of a bigger computing device [1], [13]. Essentially, the main idea is that edge-centric computing (fog and mist) devices are administered locally and not solely by a centralized third party in the cloud.

B. DECENTRALIZED CONSENSUS FOR EDGE-CENTRIC IoT

Privacy and security are two main challenges for IoT nodes while exchanging data among each other. Traditional solutions with centralized authority and message brokering are not only prone to single-point-of-failure but are also not scalable to handle hundreds of billions of transactions generated by the exponentially growing number of devices. The obvious solution is a DCS for edge-centric IoT, which is based on a trustless and immutable public ledger that uses either blockchain or blockchainless DAG data structures. Each edge-centric IoT device would administer its own role and behavior plus adhere to interaction rules [10], [12]. In short, according to [12], a blockchain or DAG-IoT combination "(a) facilitates the sharing of services and resources leading to the creation of a marketplace of services between devices, and (b) allows automation in a cryptographically verifiable manner several existing, time-consuming workflows." A blockchain or DAG resorts to a P2P network for verification and authorization of all transactions, which are conducted using crypto tokens. Henceforth, DCS for edge centric IoT allows the sharing and subsequently trading off of services/resources, consequently enabling a machine economy via device autonomy in decision-making.

III. TAXONOMY OF DECENTRALIZED CONSENSUS SYSTEMS FOR EDGE-CENTRIC IoT

The focus of this section is on examining a thematic taxonomy to categorize DCSs for edge-centric IoT. The DCSs are classified based on common features among these systems which include, data structure, scalable consensus ledger and transaction model, as illustrated in Fig. 1.

A. DATA STRUCTURE

This subsection presents the data structure of DCSs for edge-centric IoT, namely blockchain and blockchainless DAG. It describes and illustrates how each data type serves as an immutable public ledger for transactions. Albeit a blockchain can be regarded as DAG since all blockchains are subsets of trees and all trees are subsets of DAG, in this paper we categorize them as two distinct data structures.

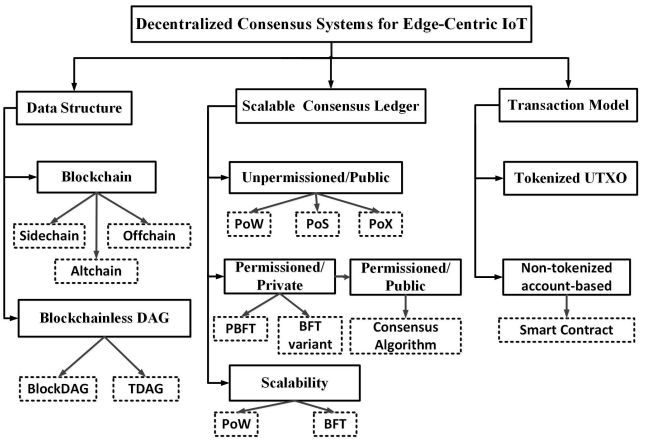


FIGURE 1. Taxonomy of DCSs for Edge-centric Internet of Things.

1) BLOCKCHAIN

The blockchain network data structure creates a group of trustless participating nodes (devices) sharing a public ledger database without middleman involvement [14]. In order to prevent the distributed environment from chaos and to lead the blockchain network toward consensus with a common global worldview, each database transaction should adhere to a predetermined rule like the Longest-Chain-Rule (LCR) in Bitcoin. Without this rule, separate copies of the blockchain will diverge into separate forks. As a result, it would cause the network to fail to establish a common acceptance truth or a unique authoritative chronology blockchain [14]. Apart from the classical Bitcoin blockchain data structure, below are the three other possible blockchain variants.

a: SIDECHAIN

Sidechain is a protocol that permits developers to connect new sidechains to the mainchain (e.g. Bitcoin). It allows to-and-from transfers of BTC (or other ledger assets) between the mainchain and various sidechains. These sidechains can have different properties from mainchain Bitcoin, in that they use different altcoins.

b: OFFCHAIN

Offchain works with off mainchain transactions. Offchain transactions are registered on a local ledger that is sometimes synchronized or broadcast to the mainchain. There are basically two types of offchain transactions, the first of which needs a third party as an intermediary and the second does not. This will be explained further in Section IV.A.2.

c: ALTCHAIN

Alternative blockchain or altchain implements separate blockchain technology to achieve distributed consensus and may use a different token as a mean of payment. However, unlike altcoin, its main entity is not 'coins' but is more for other topics or applications like smart contracts (Ethereum and Tendermint), name registration (Namecoin), file storage,

voting systems, etc. Altchains may use merge-mining to share miners with a parent Bitcoin network.

2) BLOCKCHAINLESS DAG

The aim of this data structure is to make the 'blockchain' scalable. The key to high scalability is to break away from the chain and use DAG.

a: BLOCKDAG

One possible design other than the chain proposed in [15] runs at much faster rates. This design comprises blocks in a DAG; hence it is named blockDAG. This blockDAG structure is generated via blocks referencing multiple predecessors, and more ''inclusive'' transaction rulings to accept even transactions that are embedded in blocks that are apparently in conflicts. Thus, larger blocks with long propagation time are acceptable in the network, thereby shoring up the transaction volumes and throughput [15]. A key facet of this inclusive protocol is the rewarding of the block's creator with transaction fees, although the block is not from the main chain.

b: TDAG

Instead of having a blockchain, transaction-chain or blockDAG, a transaction directed acyclic graph (TDAG) is formed with each transaction containing a list/Merkle-tree of previous transactions' hashes [16]. The main reason a transaction-chain would not scale is that with high transaction volumes, there are huge volumes of orphan-chains and low-value transactions will have difficulty getting into the chain. However, with transactions forming a DAG, this problem is potentially solvable. While large numbers of transactions may still share a common parent (or parents), providing they do not contradict each other, they may be brought back together by having the next generation of transactions ''descend'' from all of them. However, for this mechanism to function, people need incentivises to include as many childless transactions as their transactions' parents as possible and as near as possible to TDAG bottom.

B. SCALABLE CONSENSUS LEDGER

This subsection presents the scalable consensus ledger of DCSs for edge-centric IoT. A scalable consensus voting system is required for all approving nodes to select the proper sequence of succeeding transactions or blocks. This scalable ledger can be shared and corroborated on the accuracy of this transactions or blocks ordering via consensus by anyone with either unpermissioned (permissionless) or permissioned participation.

1) UNPERMISSIONED/PERMISSIONLESS

In an *unpermissioned* public shared ledger [17] like Bitcoin, anyone can use copies of the distributed ledger and maintain the ledger's integrity via trustless consensus. In essence, this prevents contributing censorship on any transaction or block to the ledger.

Proof-of-work (PoW): A Sybil attack [18] is when one entity deliberately poses with many identities and is entitled to many votes, thereby biasing the system in their favor or to their aims. Bitcoin circumvents this threat via computationally ''costly'' mining such that the computing resources of one entity are insufficient to impersonate many entities [19]. To be exact, for any assembled block that qualifies as a subsequently mined block on the network, any node must search for the correct nonce (random number) in the block header. This will form the header's hash SHA-256 [20] [21] in order to achieve the amount of expected leading zeroes in the target [22]. PoW [23] is said to be created if any of the nodes is able to ''solve'' this cryptographic puzzle. On the contrary, any other node can easily verify this ''hard-earned'' answer, owing to the node's involvement in a one-way cryptographic hash function only. However, reverse engineering is not possible to be able to guess the input from the answer.

An inconsistent state or fork will appear in the network as two challenging nodes issue blocks almost concurrently. The resolution of such fork is done automatically by the PoW mechanism of the following block, which stipulates that the fork which accumulates the highest amount of work (PoW) wins. In other words, the nodes will always accept the longest chain or fork as legitimate [23] on the consensus regarding the correct order of events [24].

Proof-of-stake (PoS) is a cheaper substitute for PoW and requires much less CPU computation in mining. The PoS stipulates that the opportunities for a node to mine the next block are closely linked to the miner's balance of stakes or tokens held. PoS strategies have their own pros and cons [25], and their execution is rather complex and calls for further study. *Proof-of-X* (PoX) is reviewed comprehensively in [11].

2) PERMISSIONED

In a *permissioned private* shared ledger [17], only the owner group can use shared ledger copies. When a record is newly appended, a bounded consensus protocol examines the ledger's integrity through trusted parties. This makes shared ledger integrity maintenance much easier than the trustless consensus protocol that unpermissioned ledgers use. Since users are whitelisted, expensive consensus mechanisms like PoW are no longer required, as there is no threat of a Sybil attack [26]. This, in essence, eliminates the mining requirement which constitutes the economic incentive, thereby permitting the adoption of other possible consensus protocols. A permissioned ledger is therefore normally faster than an unpermissioned ledger due to the above efficiency. *Practical Byzantine Fault Tolerance* (PBFT) [27] is one such consensus protocol choice. It solves the Byzantine Generals Problem [28] and can be displayed in asynchronous onset such as the Internet. It is assumed that faulty(f) nodes occupy less than 30 percent of the total nodes in PBFT. In other words, PBFT needs a minimum of $3f + 1$ nodes to work.

On the other hand, in a *permissioned public* distributed shared ledger [17], anyone can use ledger copies but the ledger's integrity is only maintained by a trusted ledger owner

or validating actors. For instance, Ripple's consensus protocol [29] employs ''collectively trusted subnetworks'' called ''Unique Node Lists (UNL)'' to tackle high latency that normally comes with BFT mechanisms. To achieve consensus, a node is only required to check on its own UNL rather than the complete network. For faulty (f) nodes tolerance, it needs a minimum of $5f + 1$ nodes to work.

3) SCALABILITY

A new study revealed the presence of serious scalability challenges in the Nakamoto protocol [30], [31]. Consensus cannot be attained if honest nodes do not act in sync quickly enough, thereby not guaranteeing transaction irreversibility (or double-spending prevention). Thus, the Nakamoto protocol was forced to have a very slow block creation rate of 10 minutes per block. Moreover, block size is also limited (currently 1MiB): the propagation delay among nodes for larger block sizes will cause more undesirable forks to appear. Adopting an increase in either block size or block creation rate to boost the throughput will render Nakamoto's primary assurance obsolete, thereby un-honest nodes would only need less than 50 percent of the computational power to launch attacks on the system [31]. However, it has also been shown that in Bitcoin's special case of using selfish mining, un-honest nodes actually need only 25 percent of the computational power to attack [32].

Nonetheless, Byzantine fault-tolerant (BFT) [28] is another replicated state machine (RSM) protocol that guarantee consensus despite the presence of Byzantines or malicious nodes. Unfortunately, BFT protocols normally have problems with node-scalability [33], which is significant to the blockchain with respect to IoT applications. The majority of ''classical'' BFT consensus algorithms have $O(N^2)$ (or in some novel highly-academic algorithms $O(Npolylog(N))$ [34]) worst-case message complexity. The Nakamoto consensus (Bitcoin), on the other hand, has worst-case message complexity of $O(N)$. This explains why Bitcoin (and others alike) can scale so nicely to thousands of nodes, whereas BFT can only scale up to around a few tens of nodes. In short, owing to the different usage aims, current blockchain consensus protocols using PoW and BFT are located in two opposite corners of the scalability/performance realm [35].

C. TRANSACTION MODEL

This subsection presents the transaction models of DCSs for edge-centric IoT, whether the tokenized Unspent Transaction Outputs (UTXO) model (e.g. Bitcoin-style transactions) or account-based model (e.g. smart contracts).

1) TOKENIZED UTXO MODEL

A blockchain that adopts the UTXO model allows the transfer of assets between mutually trustless counter-parties. In this model, every transaction consumes outputs (UTXO) created from earlier transactions and generates new outputs (UTXO) for later consumption by subsequent transactions [36]. The same token or UTXO can be consumed or spent only one time, as long as every transaction validates that its input total is equal to or exceeds its output total. Overall, the UTXO model has the advantages of enhancing paradigm privacy and scalability. The former is evident when the user employs a new address for each received transaction, making it very tough for adversaries to associate actors' accounts to each other. The latter is explained as follows: if the outputs and inputs of transactions in the block are independent, all transactions' operations can be run in parallel irrespective of the execution order.

2) NON-TOKENIZED ACCOUNT-BASED MODEL

A blockchain that adopts an account-based model (e.g. smart contracts) permits multi-stage interactive processes between mutually trustless counterparties. A smart contract is a non-tokenized script kept on the blockchain with a unique address [37]. It is activated by sending it a transaction. It then runs in a stipulated pattern, automatically and autonomously, on every node in the system based on the data stored in the activating transaction. In short, every node executes a virtual machine (VM) where the blockchain system is a distributed VM. These nodes are allowed to (i) examine the code and know its results prior to making the decision to involve it in the contract, (ii) ensure the code will be executed, as it is already employed in a decentralized network, and (iii) have authenticated processes with digitally signed interactions. Since the counterparties agree with the final result of this authentication process and all possible results are accountable, the chances of discrepancy are therefore removed.

IV. STATE-OF-THE-ART AND COMPARISON OF DECENTRALIZED CONSENSUS SYSTEMS FOR IoT

The following section concisely examines state-of-the-art, compares and critically analyses DCSs topics related to edge-centric IoT paradigms based on the parameters selected from the taxonomy reported in Section III. The comparison hinges on data structure, a scalable consensus ledger and a transaction model, as presented in Table 1.

A. DATA STRUCTURE

Data structure reflects the technologies used in DCSs that serve as an immutable public ledger for transactions. Basically, data structure is divided in two main categories: blockchain and blockchainless DAG. These are further subdivided in the subcategories presented in Table 1. The sub-groups are sidechain (items L,M,N), offchain (items O,P,Q), altchain (items R,S) and subtree (item T) for the blockchain, whereas blockchainless DAG entails BlockDAG (items U,V,W,X) and TDAG (items Y,Z,AA,BB).

1) SIDECHAIN

A sidechain permits the developer to ''connect'' new sidechaisn to a mainchain like Bitcoin. However, there are drawbacks as mentioned in item M such as complexity, fraudulent transfers, risk of mining centralization and soft-forks. Workarounds and solutions are ongoing, most notably

TABLE 1. State-of-the-art and a comparison of DCSs based on the taxonomy.

Item	State-of-the-art DCS/year[Ref.]	Data structure	Scalable Consensus Ledger	Transaction model
A	HS-Relay/2013[38]-is a centralized, exclusive and high speed relaying network/system of peering nodes, which comprised of miners, exchanges and merchants. Its main aims are to scale up the blockchain by reducing the block propagation latency among miners as well as a fallback in time of Bitcoin main network failure.	Blockchain	Permissioned: Private	Token : bitcoin(BTC)
B	PeerCensus/2016[39] - As a certification authority with high probability in secure state, the system ensures a strong consistency in Bitcoin and others by administrating peers' identities in a P2P network. Dissimilar to Bitcoin, Discoin transaction has consensus finality whereby transactions stay committed once committed. Discoin decouples block creation and transaction confirmation, permits real-time payments.	Blockchain	Permissioned: Private/PBFT	Token: Discoin
C	The Ripple/2014[29]- overcomes high latency by employing "collectively-trusted subnetworks" inside the bigger network. In reality, the requirement of these subnetworks' "trust" is rather small and could be even smaller by the principled selection of the member-nodes. Its throughput can reach in tune of 10k of transaction-per-second(tps). Ripple emphasizes on liveness and fault tolerance but not safety property.	Blockchain	Permissioned/public: Custom-made consensus algorithm	Token: XRP
D	Stellar/2015 - FBA is robust due to its "quorum slices"(qs) where each node contribute its individual trust decision that collectively forms the system-level quorum. These qs unite the system very similar to the way individual networks's peering and transit decisions integrating the Internet. Its participants are individuals whereas Ripple's more on institutional partners. Improvement over Ripple is in safety property.	Blockchain	Permissioned:Stellar Consensus Protocol (SCP)/FBA	Token : BTC
E	Peercoin/2012[40]-Initially PoW is used in the mining of coins and later it migrate to PoS to confirm/verify(mint) new blocks. In this way, PoW addresses the issue of the classical 51 percent double spending attack via computing resources, whilst PoS tackle centralization issues via minimum stake requirement. However it lacks formal guarantees of system convergence.	Blockchain	Permissionless: Proof-of-activity(PoW/PoS-hybrid)	Token : BTC
F	Counterparty/2013-has a built-in scripting language that allows smart-contracts funtionality to be performed on Bitcoin blockchain network. In this way, real world especially M2M scenarios could be translated into scripting codes and executed automatically without any third party intermediary. Its protocol is open-sourced and thus rigorously tested. Its Picopayments is fast and cheap off-chain payment channels.	Blockchain	Permissionless: PoW	Smart contract: XCP
G	SegWit/2015[41]-is the separation (segregated) of signature (witness) from the transaction data recorded in Bitcoin's block. The design of SegWit proposal is to reduce the transaction size by about 40 percent, which effectively increase the block size likewise by generating the room in Bitcoin block in similar capacity. SegWit(BIP141) was activated on 24.8.2017 whereas SegWit2x(2MB) will be in November 2017.	Blockchain	Permissionless: PoW	Token UTXO
H	DECOR+LAMI/2016[42]-DECOR (DEterministic COnflict Resolution) refers to a framework that implements the right strategy to incentivize and align the miners' behavior to the desired outcome of the protocol. LAMI, a set of (LAtency optiMIzation) implementations that improve the latency by reducing the block propagation time among miners. The relationship between selection and reward function need further studied.	Blockchain	Permissionless: PoW	Smart contract
I	Hyperledger/2015-is an open source, under Linux foundation, distributed ledger platform designed to run smart contracts. There are two types of peers that run its protocol: validating peer and non-validating peer. Its building tool-kit is under linux foundation where IBM actively participate.	Blockchain	Permissioned: Practical BFT	Smart contract
J	Bitcoin-NG/2016[43]-can survive under intense churn and having similar Bitcoin's trust pattern. Moreover, it shows various new interesting ways of quantifying the efficiency and security aspects of Bitcoin protocol. Its two block-types structure scales very prominently. However its former key blocks are still created and confirmed gradually.	Blockchain/Alt-protocol	Permissionless: PoW	Token : BTC
K	BigChainDB/2016[44]-is a novel decentralized database system at scale, with throughput performance of 1 million writes per second, data storage capacity in tune of petabytes and excellent latency in sub-second level. More nodes mean better in thoughput and capacity but worse in latency.	Blockchain/DB	Permissioned: Private/BFT	Token : BTC
L	Pegged sidechains/2014[45]-allows the interchange of the Bitcoin BTC with other different ledger assets across various blockchains. Riding on existing Bitcoin BTC, a user is therefore can access to any novel cryptocurrency systems via their ownned assets without the worry on lack of liquidity and market volatility. It divides Bitcoin into smaller networks with sidechain without PoW needed.	Blockchain/Sidechain	Permissioned: Private/BFT	Token : BTC
M	RootStock RSK/2016[46]-is an open source smart contract framework which has a two-way peg to Bitcoin and uses merge-mining to secure blockchain and incentivize miners. It scales up to 100 tps using DECOR + GHOST (Nimblecoin) protocol for 10sec/block.	Blockchain/Sidechain	Permissionless: PoW	Smart contract using 2-way peg security
N	Mimblewimble/2016-employs extremely different ways to construct Bitcoin-like transaction by noninteractive merging, cut-through of confidential transactions and no requirement of full history blockchain verification. It reduces Bitcoin history of 15Gb to not even 1 MB, enhanced privacy but not for micropayment channel yet.	Blockchain/Sidechain	Permissionless: Compressing PoW	UTXO token
O	Duplex Micropayment Channel(DMC)/2015[47]-provides a deterministic end-to-end security and facilitates instant Payment Service Providers (PSP) network. An enduring channel can be generated to allow near infinite number of transfers to be executed locally between two users, with no undesired loading to main Bitcoin network. It needs to handle conflict resolution between tx issuers adn tx approvers.	Blockchain/Off-chain	Permissioless:Peer-to-peer trustless payment channel	Token : BTC
P	Lightning Network (LN)/2015[48]-transaction values are transferred over a network of payment channels. With this, Bitcoin is now able to scale not only to billions of users but also down to the satoshi level of micro or nano payments, coupled with almost instant or real-time transactions. LN is still not widespread deployed due to its routing issue of tx in the second layer(offchain) albeit SegWit activation helps its adoption.	Blockchain/Off-chain	Permissioless:Peer-to-peer trustless payment channel	Low fees and secured by smart contract; token : BTC
Q	Tumblebit/2016[49]-is a novel off-chain unidirectional and untraceable payment channel which is interoperable with existing Bitcoin blockchain. It permits users to conduct near instant, anonymous, off-chain micropayments via a trustless intermediary - the Tumbler. Similar to LN in its adoption status.	Blockchain/Off-chain	Permissionless via a trustless intermediary - the Tumbler	Better privacy and settlement;Token:BTC
R	Ethereum/2014[50][51]-opens possibilities in resources that comes with each individual executional codes and state, but able to fulfill interaction via a "message passing framework". Its main objective to end-developer is to provide a trustful, end-to-end, fully integrated software development platform which is greatly needed to be explored and exploited, via object messaging compute framework. Using GHOST variant, focus more on A than C in CAP theorem. Fit for Dapp but not for IoT micro txs. However,its new Raiden Network[52] is for fast payment offchain channel.	Blockchain/Altchain	Permissionless: PoW/PoS(Casper)	Smart contract; token : Ether (ETH)
S	Tendermint/2014[53]-assumes a partial synchrony algorithm, derived from the Dwork, Lynch, Stockmeyer(DLS) protocol [54], with attack-resilient up to 30 percent of Byzantine users. Similar to Ethereum but focus more on C than A in CAP theorem. Scale upto 10,000 tps. Not for public IoT but more for private/hybrid enterprise application.	Blockchain/Altchain	Permissioned: Proof-of-validation/PoS/PBFT-similar	Smart contract
T	GHOST/2013[55]-is a novel policy in protocol whereby the selection of mainchain is based on the the heaviest sub-tree instead of the classical longest chain. It scales upto 214tps and prevent safety attack due to improved blockrate but its liveness can be compromised.	Blockchain/sub-tree	Permissionless: PoW	Token : BTC
U	Inclusive BlockDag/2015[15]-Using more "forgiving" transaction acceptance rule, it permits to include transactions even embedded in conflicting blocks. Henceforth bigger size blocks, which are slower in propagation, are allowed, and the amount of transactions can be shored up, thereby enable higher rate of operation as compared to those in Bitcoin. It scales significantly and solve liveness issues in GHOST. But still need total order thus facing security-scalability trade-off.	BlockDAG	Permissionless: PoW	Token : BTC
V	Spectre/2016[56]- is secured against attacker with upto 50 percent of the computational power (bounded only by the bandwidth constraints and network congestion). Its excellent block generation rates means transactions approval only in order of seconds (bounded mainly via the network's "round-trip-time"). It improves the throughput and latency significantly.	BlockDAG	Permissionless: PoW	Token agnostic
W	Braidcoin/2015[57]-creating blockDAG that tie up asynchronously into 'braid'. It can attain the optimal block rate of 11.6 seconds in a real-life asynchronous network condition. A Bloom Filter could be used to quickly verify two non-conflicting blocks.	BlockDAG	Permissionless: PoW	Token : BTC
X	Jute/2016[58]-The blockDAG is coupled with a comprehensive sorting algorithm that aligns blocks in an exact order, securing Jute from the risk of reorganization or reordering unless it is in the case of 51 percent attack. Block time is set to 6 seconds and block size to 50kb.	BlockDAG	Permissionless: PoW	Token : BTC
Y	DagCoin/2015[59]-The degree of confirmation security is gauged by the accumulated weightage or total PoW spend in referencing the transaction. Similar to the Tangle but no tip approval strategy.	TDAG	Permissionless: PoW	Token
Z	The Tangle/2015-16[60]-employing the concept of zero transaction fee, which makes autonomous M2M micro- or nano-payments possible in the realm of emerging IoT. After PoW, the Tangle will then conduct a probabilistic analysis using a Markov Chain Monte Carlo (MCMC) as the tip selection algorithm for choosing which transactions to approve. Fit for IoT since feeless micropayments. Focus on P than C in CAP theorem. Uses eventual C instead. Its Flash Network[61] is for second layer fast payment channel.	TDAG	Permissionless: PoW	40% Token: IOTA; 60% account based
AA	Byteball/2016[62]-Without the need of miners to find a new block, every new transaction gets referenced, implicitly confirmed(at-least partially) almost immediately by peers. These direct and indirect references build-up the confirmation or security like a snowball and thus the name byteball. With tx-fee, thus it is more for Bitcoin replacement rather than for IoT.	TDAG	Permissioned	Token: Byte; Smart contract, debts...
AB	Blockchain-free crypto/2016[63]-addressing blockchain issues: "mining pool" oligopolies or centralization and incompressible high latency in block validation. As it needs tx-fee and miners, thus less fit for IoT.	TDAG	Permissionless: PoW	Token, smart contract, etc

the merge-mining techniques in items M and N. As each sidechain represents a fraction of the overall network hash-rate, it is vulnerable to the infamous 51 percent attack where the weakest sidechain can be compromised easily. As a result, an attacker can not only double-spend, but they can steal from the Bitcoin 'bank' that supports the sidechain altcoin, and put the sidechain dangerously operating at a very minimum reserve. One solution is to use merge-mining to ensure all sidechains are mined with the same hashrate at the same time, thereby generating successful PoW for both blockchains together. However, these efforts require full-node operation, which is expensive; hence, there is incentive to employ a mining-pool to split the cost. This also presents the risk of centralization. Core developers remain skeptic of this

sidechain proposal despite continuing efforts to resolve these limitations.

2) OFFCHAIN

An offchain functions on off-mainchain transactions, which are registered on a local ledger that is sometimes synchronized or broadcasts to the mainchain. An example where a third party is requires as an intermediary is Coinbase, which offers instant transfers between customers but with the obvious intermediary risks like customers assets losses or freezing beyond the customers' control. The no-intermediary type is Table 1 lists for items O, P and Q, which refer to payment channels where signed transactions on the mainchain are followed by series of instant offchain payments until there is a final settlement or disagreement to process on the mainchain. This addresses the issue of loss-of-coins but active interaction and connectivity are necessary for payment transfers. It is obvious that the latter type is more suitable for DCSs for IoT, which require instant, trustless and secured micropayments. However, there are concerns with payment channels, especially for item P, that come with three security assumptions. First, the main blockchain functions well by confirming the transaction speedily, otherwise one would fail to retrieve funds from the channel, or 'fight back' any cheating encountered. Second, the channel node must safely secure the data; otherwise if the channel in the present state is exposed or stolen prior to a payment confirmation, then the adversary can revoke all transactions and steal the money from the broadcasted channel's transactions. Finally there must be no significant software bugs, else adversaries can identify the weakness in all edge cases such as not broadcasting the exact transaction at the exact time. In such case, adversaries can steal money from the channel.

3) ALTCHAIN AND SUBTREE

The alternative blockchain, or altchain, implements a separate blockchain technology to achieve distributed consensus. It may use a different token as payment means or emphasize on other than 'coins'. In Table 1, two altchains (items R and S) are identified in relation with DCSs for IoT applications. The difference between items R and S is that Ethereum uses a sub-tree variant from the original subtree concept GHOST (Greedy Heaviest Observed Sub-Tree) in item T. The sub-tree concept differs from the conventional Bitcoin LCR in countering the malicious double-spend attack when the block rate increases to scale up the blockchain. Increasing the block rate will raise the number of forks, thus inviting potential double-spend attacks. This GHOST protocol [55] demonstrates that fairness and mining power utilization can also be enhanced by including forks outside the mainchain.

4) BLOCKDAG

To facilitate operation at much higher rates, a different blockchain design is proposed. The proposed design employs blocks configured in a directed acyclic graph (DAG) pattern; hence the name blockDAG [15] (Table 1 items U, V, W and X). The blockDAG is formed by having blocks referencing multiple predecessors, as well as including blocks that have conflicting transactions in items U, V and X, but not in item W. Henceforth, bigger blocks with slower propagation time are acceptable in the system. This will subsequently shore up the transaction volume significantly.

This new protocol [15] using blockDAG offers even larger throughput in addressing the pressure of scaling up, as well as offers an incentive scheme that is fairer to weaker miners; hence, security is more robust as well. Items U, V, W and X have several things in common. However, among the four items, Jute, Inclusive BlockDAG and Spectre allow conflicting transactions, whereas Braidcoin does not. As a result Simple-Payment-Verification (SPV) is preserved in Braidcoin but not in Jute, where the SPV needs to be adjusted to ignore any blocks with less than a certain number of confirmations. Another difference is that to achieve consistency, item U reorders DAGs by linearlization, Jute uses a linked list, but Braidcoin employs the cohort method where the block is decided solely from the graph structure during a slower network growth rate than that of the bead publication rate.

5) TDAG

TDAG is a DAG structure without blocks but composed of transactions instead. Each transaction contains a list/Merkle-tree of previous transactions' hashes. There are two advantages TDAG designs over blockchain or blockDAG designs, especially in light of the ever-increasing data growth. First is the speed. Any new transaction will have atleast partial confirmations from peers almost instantly once released into the network, meaning no more long waits for miners to secure a new block. Secondly, TDAG outperforms blockchain or blockDAG in terms of scalability. When the rate of new transactions or tips is high, the TDAG structure becomes wide. There are no classical problems with block size limits that prevent long propagation delays due to larger block, and fork orphaning due to higher block rate. Among the four items Y, Z, AA and BB, Dagcoin was published the earliest, followed by IOTA Tangle that appeared on the mainnet in July 2016 and finally Byteball in September 2016. All three are similar in using the innovative DAG structure with no roles separation between transaction issuer and transaction approver, meaning there are no miners. Dagcoin is currently similar to Tangle but has no tip approval strategy such that its TDAG structure keeps widening into no apparent mainchain. Tangle and Byteball have other similarities besides the DAG structure. First, both Byteball and IOTA tokens are not inflationary, and a huge number are forecasted in IOTA on account of future the countless IoT devices. Second, both are quantum secured, where Tangle employs hash-based Winternitz signatures but Byteball uses the NTRU algorithm. Quantum-resistant cryptography is important to PoW efficiency because based on the quantum Grovers algorithm($\Theta(\sqrt{N})$), Tangle for example is much more secure in quantum computing (Table 2).

TABLE 2. Resistance to security threat by quantum computer (adapting from [60]).

	Classical computer operations	Quantum computer operations	Quantum computer better efficiency by
Blockchain: To issue a block, need $N = 2^{68}$ nonces to find hash	$\Theta(N)$	$\Theta(\sqrt{N})$, i.e. $\sqrt{2^{68}} = 2^{34}$	17 billion times!
Tangle TDAG: To issue a transaction, need $N = 3^8$ nonces	$\Theta(N)$	$\Theta(\sqrt{N}) = 10(\sqrt{N})$	only in the order of $3^4 = 81$.

Third, both Byteball and Tangle are scalable but Tangle is forkable for minute IoT devices. Thus, there is chance of orphaning (abandoned forks) in Tangle but not in Byteball. Both also have a number of technical differences due to their respective strengths in their application areas. That is, Tangle still employs PoW, whereas Byteball uses mainchain to attain total ordering of transactions. Tangle has no transaction fees and is suitable for conducting micropayments in IoT applications, but Byteball has a fee of one Byte per Byte data. The transaction finality in Tangle is still probabilistic as it is based on the Markov Chain Monte Carlo (MCMC) strategy (currently using checkpoint/coordinator for control), whereas Byteball is deterministic. IOTA is a single token specifically for IoT machine-to-machine (M2M) micropayment transactions, whereas Byteball has more assets like a smart contract besides its Byte token. All consensus ledgers in Tangle are permissionless/public but some Byteball assets are permissioned/private. These explain the unique feature of the Byteball integrated private asset that permits conducting anonymous transactions, which is very suitable for Fiat-currency replacement similar to Bitcoin in the long run. In contrast, Tangle is more suitable for edge-centric IoT applications. Moreover, Tangle file sizes are kept small for two reasons. First is the snapshotting feature whereby the network can easily snapshot the Tangle if it becomes too huge. Next is the Lightclient feature whereby any tiny IoT device can easily sign a transaction while another processing station does the hashing and later appends the transaction to the Tangle. Hence, the tiny IoT device does not require in storing the entire Tangle.

B. SCALABLE CONSENSUS LEDGER

Decentralized, anonymous and public accessibility are the main features of the permissionless consensus ledger. In contrast, a permissioned ledger provides immutability, granular permission granting capability, automatic processing and data-syncing, plus superb privacy and security. In short, it all depends on the aspiration of either decentralized connectivity in a permissionless consensus ledger or enterprise efficiency in a permissioned consensus ledger. The former is apparently applicable in use-cases, such as industry disruption, disintermediation and social platforms, whereas the latter suits several enterprise use-cases. For all the pros and cons, the ultimate trade-offs to consider are trustlessness and scalability. In PoW blockchain design, the greater the trustlessness, the more computational power is needed. Moreover, to ensure privacy, a transaction fee is needed to reward the miner who solves the cryptographic evidence. Using sharding in Tendermint and future Ethereum, private permissioned and public permissionless ledgers can co-exist to attain the benefits of both for optimal performance.

1) PERMISSIONLESS PoW: BLOCKCHAIN vs TDAG

Blockchainless DAG design especially TDAG is inherently superior in scalability over blockchain design. Thus, the ultimate trade-off left to consider is trustlessness. However, IoT application automatic device accessibility warrants a PoW permissionless scalable ledger design such as the IOTA Tangle. One of the key differences between the permissionless PoW blockchain and PoW TDAG design is that TDAG has no miners. This is because the transaction issuer is also the transaction approver. It is therefore no more consensus roles decoupling but the users themselves self-regulate. The advantages of having no miners involved are mainly three fold. First, there is no centralization control (computational and political) by a few major pool operators which would allow them to abuse the policies on filtering, censoring and postponing specific transactions. Second, there is no participant discrimination, whereas a clear separation of roles into miners and issuers would cause discrimination and consequently conflict. In such case, all the participants would end up spending resources to resolve the conflict. Third, there is no transaction fee to pay miners, which is required in blockchain PoW to achieve consensus by paying miners' expenses and tackling spam attacks. Therefore, the no transaction fee feature is paramount to the IoT ecosystem.

2) SCALABILITY

To achieve decentralization, blockchain solutions like Bitcoin/Ethereum have limitations on the maximum transaction rate. However, the high transaction rate is among the basic requirements and the backbone for IoT applications.

Duplex micropayment channels enlighten a myriad of issues. Among them is that they allow excellent scalability for Bitcoin-based transactions. Bitcoin transactions are employed to set up micropayment channels as well as handle conflict resolution between issuers and approvers. Existing transfers of bitcoins (BTC) from payers to payee are now managed at a higher level via the Payment Service Provider (PSP) network. Using hashed timelock contracts, the security of these payments is maintained and transfers between hops only happen if the recipient-to-be receives the payment. These transfers are much more suitable for real-time applications (only bounded by network speed) and secured against reverse payments, which is very different from the Bitcoin approach that needs a long confirmation time (10 mins). Thus, using the PSP network in duplex micropayment channels, Bitcoin can now carry out real-time and genuine micropayments with minimal fees and excellent scalability [47].

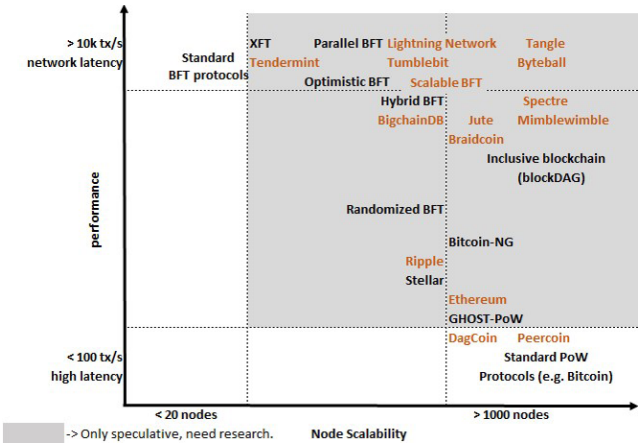


FIGURE 2. Performance vs node scalability (Adapting from [35]).

The above examples are some great proposals on the list to improve Bitcoin/Ethereum beyond current capabilities. For instance, projects like Tendermint [53], Lightning Network [48], Ethereum's Raiden Network [52] and IOTA's Flash Network [61] exhibit as second layer solutions, but none of the current proposals make them a great choice as the backbone of the underlying IoT layer. This is where TDAG (e.g. IOTA Tangle) comes in, which is lightweight (due to the Lightclient and snapshotting features), highly scalable and has no fees. Fig 2 presents the overall positioning of the respective solutions regarding the fitness for application in edge-centric IoT. Generally, the PoW-based blockchain provides excellent node scalability but limited performance(transaction throughput and latency), whilst the BFT-based blockchain delivers the opposite. Moreover, the blockchainless concept in PoW-based blockDAG and TDAG offers superior performance especially in TDAG solutions such as IOTA Tangle and Byteball. In particular, IOTA Tangle manages to be highly scalable as it forgoes immediate global consensus (but will rather achieve it at some later stage) and permits node generation of transactions to proceed despite the truth of their ''current view'' of the network still being unknown. One thing in common between Lightning Network and IOTA Tangle in achieving excellent transactional scallability is the off-chain or off-Tangle features. Both are able to circumvent the limitation faced while on-chain or on-Tangle.

C. TRANSACTION MODEL

In the consensus system, it is the transactional logic that determines the shared computational state. Bitcoin and its derived altcoins in the list were designed to basically transfer balances of digital assets or values in cryptocurrencies from a payer to the payee while their transaction logic or model is within a token system. However the programming language Script used in this token system is not Turing complete and there is no sense of state other than either True or False only. In contrast, the transactional logic or model can be made arbitrarily complex with the Turing complete languages executable with a virtual machine (VM) sitting on top of the shared replicated ledger. This VM permits the generation of smart contracts which are multiparty agreements coupled with cryptographic authentication, self-executing and self-enforcing capabilities. For IoT application with millions of tiny devices, the token system is most appropriate for the transfer of resources and values, for example IOTA in Tangle. For more complex IoT applications, for instance the IBM and Samsung collaboration on the IoT project ADEPT employs smart-contract based systems in Ethereum. However there is a great synergy between blockchain technology like Ethereum and a blockchainless technology in TDAG like IOTA Tangle. Ethereum can run in a type of central control station, while Tangle runs on all the tiny devices and interconnects them with each other and with the Ethereum blockchain. This reflects maximum interoperability in IoT system is attainable by creating an environment where TDAG (e.g. Tangle) becomes the backbone of the IoT system, while different blockchains (mostly smart contracts) handle other application aspects. Apart from monetary transactions (which can be nano-transactions), Tangle for example also allows devices to send messages, making it a perfect solution for IoT devices to communicate with each other and with the outside world. There is big difference compared to the TCP/IP protocol, in that M2M communication needs to maintain message authenticity and integrity. By messaging on a distributed ledger it is possible to prove the authenticity and integrity of a message besides the inherent tamper-proof feature.

V. DISCUSSION ON RESEARCH ISSUES IN DECENTRALIZED CONSENSUS FOR EDGE-CENTRIC IoT

This section thoroughly discusses recent issues in the decentralized consensus for edge-centric IoT research domain. These are recent centralization risk concerns due to the economies of scale in PoW, the offchain scalabilty solution dilemma in the Lightning network and the consistency concern in IOTA Tangle that may need a check-point for control.

A. CENTRALIZATION RISK DUE TO ECONOMIES OF SCALE IN PoW

The original intention of Nakamoto's design was mining on computer CPUs. However, the graphic card GPU was later employed owing to its higher hashing power. Subsequently, GPUs were overtaken by the dedicated ASICs. Currently all professional Bitcoin mining is conducted on ASICs, typically in data centres with proper ventilation and accessibility to cheap or subsidized electricity. Economies of scale in PoW have therefore caused mining power to fall to fewer individuals than originally intended. Similarly, Nakamoto also failed to foresee the imminent mining pools where collaborative miners share block rewards relative to their mining power contribution. However, it is unfortunate that power is now being concentrated to the pool owner and specialized hardware thereby constituting the infamous centralization risk where a few entities dominate most of the hashing power. Moreover, it is acclaimed that not less than 50 percent of

mining hardware is now situated in China. Nonetheless, this centralization risk is mitigated by the argument that such attack will not be in line with miners' long-term economic interests. This is because the risk will cause Bitcoin's integrity and hence the exchange rate to fall sharply, compromising miner' well-being in terms of hardware investment and their coins' values. As such, a 51 percent attack risk would render a bad risk-reward ratio for miners as the community could rationally accept the last honest block and reject the dishonest chain.

B. DILEMMA WITH BLOCKCHAIN SOLUTION ADOPTION

While the Bitcoin development community is generally enthusiastic about SegWit, it is evident that some miners are not interested in its adoption. The reason is that miners incur in transaction fees losses through the Lightning Network offchain payment process rather than via the traditional miners' on-chain transactiosn. As mentioned earlier, it is only by coupling the capability of decentralization using PoW with the scripting capabilities that the blockchain potential can be greatly unleashed. It can be used to achieve sophisticated transactions and contracts albeit the instruction set is rather limited. This feature is further enhanced by a Turing complete scripting language digital ledger that is capable of running smart contract in Ethereum and Tendermint. Currently a new proposal Aeternity channels [64] claims to have all the Lightning Network features as well as the smart contract capabilities. Whereas Ethereum transactions are executed one at a time sequentially, Aeternity transactions can be done in parallel so it is more scalable. Likewise, the Lightning network only concentrates on executing transfers, whereas the Aeternity state channel can process functional smart contracts. In short, Ethereum is to BitCoin as Aeternity is to the Lighting network plus Bitcoin.

C. IOTA TANGLE MAY NEED CHECK-POINT FOR CONTROL

All decentralized consensus designs have a certain centralization flaw when it comes to mining (as shown in section V.A), but this can be prevented by allowing the transaction issuers to conduct PoW instead of miners, as with the Iota Tangle solution. However, it may seems to have flaws in its probabilistic security model that may need a certain centralized check-point to co-ordinate and overcome a tragedy of the commons. The reason is that IOTA Tangle could not assert Nakamoto's ingenious security model which is bounded by LCR, but instead to be unbounded in security risk where mathematical modelling cannot cover all in game theoretical analysis. Tangle has excellent nodes scalability via partition-tolerance. But the cost of this universality - semantics of the consistency cannot be unambiguously provably obtained. Thus, Tangle makes a tradeoff in the CAP theorem [65], that is, it only guarantees eventual consistency but not strong consistency. Since Tangle can interact with any blockchain technology from Bitcoin to Ethereum, there is the possibility of establishing a collborative communication network of multiple blockchains, which leads to the opportunity to develop even more robust and flexible decentralized solutions.

VI. CONCLUSIONS AND FUTURE WORKS

In this survey, we studied how edge-centric IoT evolved from cloud-centric IoT as well as the need for a decentralized structure to counter centralized structure security problems. We also evaluated the broad fields of Blockchain and DAG with their features and associated concepts. We specifically scrutinized the foundation of the Nakamoto protocol, and explored the challenges with blockchain scaling against the security tradeoff. We realized that TDAG technology (especially IOTA Tangle) may be an answer in addressing the tremendous, ever-growing scale of edge-centric IoT and the absolute need for very low latency micro-payments in the M2M P2P decentralised infrastructure. By doing so, we delivered a holistic technical perspective on distributed decentralized consensus viv-a-vis edge-centric IoT. We also pointed out future research opportunities, where both blockchain and blockchainless DAG solutions can work cohesively to deliver a complete and comprehensive edge-centric IoT solution.

REFERENCES

[1] A. Corsaro, "Cloudy, foggy and misty Internet of Things," in *Proc. 7th ACM/SPEC Int. Conf. Perform. Eng.*, 2016, p. 261.
[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generat. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.
[3] H. Madsen, G. Albeanu, B. Burtschy, and F. Popentiu-Vladicescu, "Reliability in the utility computing era: Towards reliable fog computing," in *Proc. 20th Int. Conf. Syst., Signals Image Process. (IWSSIP)*, Jul. 2013, pp. 43–46.
[4] P. Brody and V. Pureswaran, "Device democracy: Saving the future of the Internet of Things," IBM, New York, NY, USA, Tech. Rep., Sep. 2014. [Online]. Available: http://www-935.ibm.com/services/us/gbs/thoughtleadership/internetofthings/, doi: 10.1109/ACCESS.2016.2566339.
[5] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A survey on Internet of Things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, Jan. 2014.
[6] C. Zhu, V. C. M. Leung, L. Shu, and E. C.-H. Ngai, "Green Internet of Things for smart world," *IEEE Access*, vol. 3, pp. 2151–2162, Nov. 2015.
[7] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput.*, 2012, pp. 13–16.
[8] I. Stojmenovic, "Fog computing: A cloud to the ground support for smart things and machine-to-machine networks," in *Proc. Austral. Telecommun. Netw. Appl. Conf. (ATNAC)*, Nov. 2014, pp. 117–122.
[9] R. L. Reid, "Tomorrowworld," *Civil Eng. Mag. Arch.*, vol. 85, no. 6, pp. 54–59, 2015.
[10] F. Glaser and L. Bezzenberger, "Beyond cryptocurrencies—A taxonomy of decentralized consensus systems," in *Proc. 23rd Eur. Conf. Inf. Syst. (ECIS)*, Münster, Germany, 2015, pp. 1–18.
[11] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2015.
[12] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
[13] P. Garcia Lopez *et al.*, "Edge-centric computing: Vision and challenges," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 5, pp. 37–42, 2015.
[14] G. Greenspan, "Avoiding the pointless blockchain project," Tech. Rep., 2015. [Online]. Available: http://www.multichain.com/blog/2015/11/avoidingpointless-blockchain-project/, doi: 10.1109/ACCESS.2016.2566339.
[15] Y. Lewenberg, Y. Sompolinsky, and A. Zohar, "Inclusive block chain protocols," in *Financial Cryptography and Data Security*. Berlin, Germany: Springer, 2015, pp. 528–547.

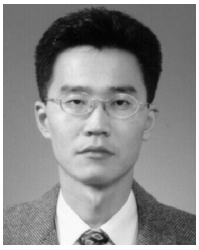
[16] TomHolden. (2016). *Proposal: Transaction-Directed Acyclic Graphs*. [Online]. Available: https://bitcointalk.org/index.php?topic=1504649.0
[17] M. Walport, "Distributed ledger technology: Beyond blockchain," U.K. Government Office Sci., Tech. Rep., 2016. [Online]. Available: https://www.gov.uk/government/publications/distributed-ledger-technology-blackett-review, doi: 10.1109/ACCESS.2016.2566339.
[18] J. R. Douceur, "The Sybil attack," in *Proc. 1st Int. Workshop Peer-to-Peer Syst.*, 2002, pp. 251–260.
[19] A. Juels and J. G. Brainard, "Client puzzles: A cryptographic countermeasure against connection depletion attacks," in *Proc. NDSS*, vol. 99. 1999, pp. 151–165.
[20] *Announcing the Secure Hash Standard*. Accessed: Jun. 6, 2017. [Online]. Available: http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf
[21] Hashcash. *Hashcash—Bitcoin Wiki*. Accessed: Jun. 6, 2017. [Online]. Available: https://en.bitcoin.it/wiki/
[22] A. M. Antonopoulos, *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*, 1st ed. Newton, MA, USA: O'Reilly Media, Inc., 2014.
[23] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Tech. Rep., 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf, doi: 10.1109/SP.2015.14.
[24] F. B. Schneider, "Implementing fault-tolerant services using the state machine approach: A tutorial," *ACM Comput. Surv. (CSUR)*, vol. 22, no. 4, pp. 299–319, 1990.
[25] V. Buterin. (Jul. 2014). *On Stake*. [Online]. Available: https://blog.ethereum.org/2014/07/05/stake
[26] T. Swanson, "Consensus-as-a-service: A brief report on the emergence of permissioned, distributed ledger systems," Tech. Rep., 2015. [Online]. Available: http://www.ofnumbers.com/2015/04/06/consensus-as-a-service-a-brief-report-on-the-emergence-ofpermissioned-distributed-ledger-systems/, doi:10.1109/ACCESS.2016.2566339.
[27] M. Castro and B. Liskov "Practical Byzantine fault tolerance," in *Proc. OSDI*, vol. 99. 1999, pp. 173–186.
[28] L. Lamport, R. Shostak, and M. Pease, "The Byzantine generals problem," *ACM Trans. Program. Lang. Syst.*, vol. 4, no. 3, pp. 382–401, Jul. 1982. [Online]. Available: http://doi.acm.org/10.1145/357172.357176
[29] D. Schwartz, N. Youngs, and A. Britto, "The ripple protocol consensus algorithm," Ripple Labs Inc., San Francisco, CA, USA, White Paper, 2014, p. 5. [Online]. Available: https://ripple.com/files/ripple_consensus_whitepaper.pdf
[30] C. Decker and R. Wattenhofer, "Information propagation in the Bitcoin network," in *Proc. IEEE 13th Int. Conf. Peer-to-Peer Comput. (P2P)*, Sep. 2013, pp. 1–10.
[31] Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in Bitcoin," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, 2015, pp. 507–527.
[32] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.*, 2014, pp. 436–454.
[33] E. A. Brewer, "Towards robust distributed systems," in *Proc. PODC*, 2000, p. 7.
[34] V. King, S. Lonargan, J. Saia, and A. Trehan, "Load balanced scalable byzantine agreement through quorum building, with full information," in *Proc. Int. Conf. Distrib. Comput. Netw.*, 2011, pp. 203–214.
[35] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication," in *Proc. Int. Workshop Open Problems Netw. Secur.*, 2015, pp. 112–125.
[36] A. M. Antonopoulos, *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*. Newton, MA, USA: O'Reilly Media, Inc., 2014.
[37] N. Szabo, "Smart contracts," to be published. [Online]. Available: http://szabo.best.vwh.net/smart.contracts.html, doi: 10.1109/ACCESS.2016.2566339.
[38] M. Corallo, "High-speed bitcoin relay network," Tech. Rep., Nov. 2013, doi: 10.1109/SP.2015.14.
[39] C. Decker, J. Seidel, and R. Wattenhofer, "Bitcoin meets strong consistency," in *Proc. 17th Int. Conf. Distrib. Comput. Netw.*, 2016, p. 13.
[40] S. King and S. Nadal, "PPCoin: Peer-to-peer crypto-currency with proof-of-stake," to be published, doi: 10.1109/SP.2015.14.
[41] P. Wuille, "Segregated witness and its impact on scalability," in *Proc. SF Bitcoin Devs Seminar*, 2015, doi: 10.1109/SP.2015.14.
[42] P. Camacho and S. D. Lerner, "Decor+LAMI: A scalable blockchain protocol," Tech. Rep., 2016. [Online]. Available: https://scalingbitcoin.org/papers/DECOR-LAMI.pdf
[43] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-NG: A scalable blockchain protocol," in *Proc. 13th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*, 2016, pp. 45–59.
[44] T. McConaghy *et al.*, "BigchainDB: A scalable blockchain database," Tech. Rep., 2016. [Online]. Available: https://www.bigchaindb.com/whitepaper/bigchaindbwhitepaper.pdf, doi: 10.1109/CTS.2016.0082.
[45] A. Back, G. Maxwell, M. Corallo, M. Friedenbach, and L. Dashjr, "Enabling blockchain innovations with pegged sidechains," Tech. Rep., 2014. [Online]. Available: http://www.opensciencereview.com/papers/123/enablingblockchain-innovations-with-pegged-sidechains
[46] S. Demian-Lerner. (2016). *Sidechains, Drivechains, and RSK 2-Way Peg Design*. [Online]. Available: http://www.rootstock.io/blog/sidechains-drivechains-and-rsk-2-way-peg-design
[47] C. Decker and R. Wattenhofer, "A fast and scalable payment network with bitcoin duplex micropayment channels," in *Stabilization, Safety, and Security of Distributed Systems*. Cham, Switzerland: Springer, 2015, pp. 3–18.
[48] J. Poon and T. Dryja, "The Bitcoin lightning network: Scalable off-chain instant payments," Tech. Rep., 2015. [Online]. Available: https://lightning.network/ doi: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0073.
[49] E. Heilman, L. Alshenibr, F. Baldimtsi, A. Scafuro, and S. Goldberg, "TumbleBit: An untrusted Bitcoin-compatible anonymous payment hub," NDSS, Cryptol. ePrint Arch., Tech. Rep. 2016/575, 2016.
[50] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project Yellow Paper, Tech. Rep., 2014. [Online]. Available: http://gavwood.com/paper.pdf
[51] V. Buterin, "A next-generation smart contract and decentralized application platform," White Paper, 2014. [Online]. Available: https://www.ethereum.org/pdfs/EthereumWhitePaper.pdf
[52] V. Buterin. (2017). *Raiden Network Specification*. [Online]. Available: https://raiden-network.readthedocs.io/en/stable/spec.html
[53] J. Kwon. (2014). *Tendermint: Consensus Without Mining*. [Online]. Available: http://tendermint. com/docs/tendermintfgv04.pdf
[54] C. Dwork, N. Lynch, and L. Stockmeyer, "Consensus in the presence of partial synchrony," *J. ACM*, vol. 35, no. 2, pp. 288–323, 1988.
[55] Y. Sompolinsky and A. Zohar, "Accelerating bitcoin's transaction processing. Fast money grows on trees, not chains," in *Proc. IACR Cryptol. ePrint Arch.*, 2013, p. 881.
[56] Y. Sompolinsky, Y. Lewenberg, and A. Zohar, "SPECTRE: A fast and scalable cryptocurrency protocol," in *Proc. IACR Cryptol. ePrint Arch.*, 2016, p. 1159.
[57] B. McElrath. (2015). *Braidcoin*. [Online]. Available: https://github.com/mcelrath/braidcoin
[58] D. Vorick. (2016). *Jute*. [Online]. Available: https://github.com/Taek42/jute
[59] S. D. Lerner, "DagCoin: A cryptocurrency without blocks," Tech. Rep., 2015. [Online]. Available: https://bitslog.files.wordpress.com/2015/09/dagcoin-v41.pdf
[60] P. Serguei. (2016). *The Tangle*. [Online]. Available: https://www.iotatoken.com/IOTA_Whitepaper.pdf
[61] P. Handy. (2017). *Flash Network*. [Online]. Available: https://blog.iota.org/instant-feeless-flash-channels-88572d9a4385
[62] A. Churyumov, "Byteball: A decentralized system for storage and transfer of value," 2016, to be published. [Online]. Available: https://byteball.org/Byteball.pdf
[63] X. Boyen, C. Carr, and T. Haines, "Blockchain-free cryptocurrencies. A rational framework for truly decentralised fast transactions," *IACR Cryptol. ePrint Arch.*, vol. 2016, p. 871, 2016.
[64] Z. Hess, Y. Malahov, and J. Pettersson, "AEternity blockchain," to be published. [Online]. Available: https://aeternity.com/aeternity-blockchain-whitepaper.pdf
[65] S. Simon, "Brewer's cap theorem," pp. 1–6. [Online]. Available: https://doi.org/10.1109/IntelCIS.2015.7397224

KIMCHAI YEOW received the B.Eng. degree in computer engineering from NUS, Singapore; the master's degree in IT from UNITAR; and the DBA degree from UUM, Malaysia. He is currently pursuing the Ph.D. degree in computer science with the Unversity of Malaya. He is a member of the Center for Mobile Cloud Computing Research, Malaysia. His research interests include, edge-centric computing and IoT, new generation cryptocurrency, meshnet, and blockchain and blockchainless DAG. He has 24 years of research and development working experience in RF wireless, TV, telco, and computer related products in Singapore.

ABDULLAH GANI (M'00) is currently a Professor with the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia. He has authored over 100 academic papers in conferences and journals. He is currently involving in mobile cloud computing with High Impact Research Grant. He is a Fellow of Academic Sciences Malaysia. He is currently the Director of C4MCCR, University Malaya, which focuses on high impact research.

RAJA WASIM AHMAD received the Ph.D. degree in computer science from C4MCCR, University of Malaya. He is currently an Assistant Professor with the COMSATS Institute of Information Technology, Pakistan. His research interests include mobile application energy profiling, energy efficient computational offloading, cloud resource allocation, VM migration, and IoTs.

JOEL J. P. C. RODRIGUES (S'01–M'06–SM'06) is currently a Professor with the National Institute of Telecommunications, Brazil, and a Senior Researcher with IT, Portugal. He has been a Professor with UBI, Portugal, and a Visiting Professor with UNIFOR. He is a Leader of the Internet of Things research group (CNPq), the President of the Scientific Council, IEEE ComSoc Distinguished Lecturer (2018–2019), Member of the IEEE ComSoc Board of Governors as Director for Conference Development, ParkUrbis-Covilhã Science and Technology Park, the Past-Chair of the IEEE ComSoc TCs on eHealth and on Communications Software, a Steering Committee Member of the IEEE Life Sciences Technical Community. He is an Editor-In-Chief of three international journals and editorial board member of several journals. He has authored or co-authored over 500 papers in refereed international journals and conferences, three books, and two patents.



KWANGMAN KO is currently a Professor with the School of Computer and Information Engineering, Sangji University, South Korea. His current research focuses on the retargetable tool suite (low power/energy optimized compiler, simulator, and debugger) for the embedded systems, virtual machines and the energy-oriented architecture description language. He is a committee member of the various Korea information and multimedia societies.