Distributed Ledgers for Distributed Edge

Are we there yet?

Leo Eichhorn
Technical University of Munich
Germany
leo.eichhorn@tum.de

Tanya Shreedhar
IIIT Delhi
India
tanyas@iiitd.ac.in

Aleksandr Zavodovski
Uppsala University
Sweden
aleksandr.zavodovski@it.uu.se

Nitinder Mohan
Technical University of Munich
Germany
mohan@in.tum.de

ABSTRACT

Edge computing has received significant attention from both academic and industrial research circles. The paradigm aims to decentralize the existing cloud infrastructure by incorporating resources co-located alongside its client. Researchers have also proposed solutions for a fully decentralized crowdsourced compute paradigm enabled by Distributed Ledger Technologies (DLTs). This paper investigates the rationale behind DLTs over crowdsourced resource marketplaces to support the requirements of latency-critical applications targeted by edge computing. We develop a fully configurable NEtworked Blockchain emULAtor, or `NEBULA`, to scrutinize the internal performance bottlenecks of DLTs. We evaluate two blockchain categories – *proof-based* (popularly used in Bitcoin, Ethereum) and *hybrid* consensus and find that the enabling factor of DLTs – scale – is also its primary latency contributor. We show that, in reality, the latency overheads due to DLT operation far exceed the operational requirements of edge applications.

CCS CONCEPTS

• **Networks → Public Internet**; **Network measurement**; *Cloud computing*; • **Computing methodologies → Real-time simulation**; **Distributed simulation**; • **Computer systems organization → Peer-to-peer architectures**.

KEYWORDS

Distributed Ledger Technology, Blockchain, Edge Computing, Crowdsourcing, Emulation, Network Performance

ACM Reference Format:
Leo Eichhorn, Tanya Shreedhar, Aleksandr Zavodovski, and Nitinder Mohan. 2021. Distributed Ledgers for Distributed Edge: Are we there yet?. In *Interdisciplinary Workshop on (de) Centralization in the Internet (IWCI'21), December 7, 2021, Virtual Event, Germany.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3488663.3493687

1 INTRODUCTION

The growth of cloud computing in the early 2000s ushered in a new revolution for Internet-backed application services. Application providers could now utilize seemingly unlimited compute capabilities to match the demand on-the-go – thanks to the highly managed and overly provisioned datacenter silos set up by cloud operators. However, the recent growing interest in next-generation applications such as IoT, AR/VR, etc. [44] has revealed several performance gaps in the current cloud computing setup. The next-gen applications are *mission-critical* and must operate within strict latency bounds (mostly within 100 ms) [52, 58] – which cannot be satisfied by the cloud due to its remote (and numbered) deployment.

To this end, *edge computing* has emerged as a compelling solution for such applications [53, 64, 65]. The paradigm aims to *decentralize* cloud computing by utilizing resources deployed in physical proximity to the users [3]. The computing paradigm is backed by industry and academia alike, both proposing solutions for infrastructure deployments fitting their needs. While existing cloud providers, ISPs, etc. are advocating for deploying edge servers within their managed facilities [33], unmanaged approaches such as fully decentralized crowdsourced marketplaces are simultaneously gaining popularity [40]. While the majority of works in edge computing have focused on *how* to enable the paradigm through novel networking protocols [47, 54, 55, 66] or designing applications that can best utilize compute capacity near end-users [8, 18, 28], understanding *what* "edge" is has somehow lost relevance. Recent measurement studies have found that the datacenter networks and deployment has drastically expanded during the last decade, and latencies within the WAN connecting users with cloud is quite low [20, 22, 23]. As a result, there is no clear contender for a "textbook" edge infrastructure as future deployments can (potentially) benefit from unmanaged crowdsourced resources (in terms of latency) and servers in managed environments (in terms of reliability/availability). We believe that edge computing *must* embrace all possible compute opportunities since its full potential can only be reached if the infrastructure is dense and readily available.

To support such a "collective" compute fabric, researchers have turned to Distributed Ledger Technologies (DLTs) as an enabling technology [60]. Over the past decade, DLTs (and its subclass *blockchain*) have matured – enabling several commercial-grade systems through cryptocurrencies [29] and smart contracts [72]. However,

---

IWCI'21, December 7, 2021, Virtual Event, Germany

ACM ISBN 978-1-4503-9138-2/21/12…$15.00
https://doi.org/10.1145/3488663.3493687

Category	Participation Mode	Consensus Algorithms	Consensus Complexity	Consensus Finality	Fault Tolerance	Throughput (TPS)	Latency	Example Architecture
Proof-based (PoX)	Public-Permissionless	PoW, PoS, PoR	$O(1)$	Probabilistic: LCR/GHOST, confirms.	50% compute, stake value, storage space	Hundreds	Minutes	Bitcoin [56], Ethereum [81], Permacoin [51]
BFT	Public/Private-Permissioned	PBFT, RBFT	$O(n^2) - O(n^3)$	Total	33% servers	Thousands	Seconds	Hyperledger Indy [77]
Hybrid	Permissioned, Permissionless	DPoS-BFT, Tendermint	$O(n) - O(n^3)$ in delegates	Probabilistic, Total	33% delegates, stake value	Thousands	Seconds	EOS [14], Cosmos Hub [43]
DAG	Permissioned, Permissionless	PoW, parent approval	$O(1)$	Probabilistic	50% participants / comp. power	Thousands	Seconds, Minutes	IOTA [76], Spectre [70]

Table 1: Categorizing Distributed Ledger Technology (DLT) platforms

the applicability of DLTs for decentralized crowdsourced edge computing from a practical perspective is still a relatively unexplored topic. Conceptually, DLTs can easily enable several sophisticated operations desired by the crowdsourced edge. For example, smart contracts can support task scheduling [42] and resource matching [78], while users can be incentivized to participate through cryptocurrencies. As a result, the majority of research in this space treat DLTs as a "one-size-fits-all" black box that seamlessly supports the operations desired by edge computing. Furthermore, existing work in this space completely ignores (or significantly misrepresents) the additional latencies due to DLT operations within their solutions. This lapse in insight is further fuelled by the significant lack of studies investigating (or measuring) the networking overheads of DLTs at scale. We argue (and later show) that with increasing infrastructure and participant scale, DLTs act more as a bottleneck than enablers for low latency promises of edge computing.

In this paper, we investigate the suitability of DLTs (specifically, blockchains) for addressing the latency requirements of applications over decentralized edge computing. We first identify the key components in crowdsourced marketplaces that necessarily require blockchains for operation. Further, we shed light on the internals of different DLTs, particularly focusing on networking. We develop a NEtworked Blockchain emULAtor – `NEBULA` – that allows us to scrutinize the realistic operation of different DLTs in a controlled testbed. Through extensive experimentations over different parameters (e.g., network size, density, latency, block rates, etc.), we investigate the applicability of two popular blockchains – proof-based (e.g., Bitcoin, Ethereum) and hybrid consensus (e.g., EOS) – for supporting latencies of edge-enabled applications. We realize that the primary success parameter of DLTs – i.e., *scale* – is also their biggest performance bottleneck. Our aim with this work is not to deter research in this area but to provide a reality-check for future studies. We make `NEBULA` available for public use at [27].

2 CHAINING THE EDGE

With *smart contracts*, DLTs have become general-purpose replicated state machines [88] and can now enable decentralized applications, such as electronic voting [49], data provenance [61], or item sharing [15]. As a subclass of DLTs, blockchains maintain the distributed ledger by batching transactions into immutable *blocks*, each block referencing its parent (Figure 1). Peers can participate in the blockchain in a variety of ways. For example, participation can be *public* or *private* (based on read access), *permissionless*, or *permissioned* (based on consensus participation) [37].

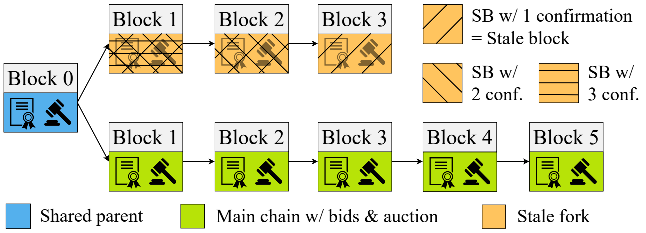


Figure 1: Stale blocks (SB) and confirmed SB in DLT.

2.1 Understanding Blockchains

Table 1 shows the four broad categories of DLTs[1] used today. *Proof-based* blockchains (PoX) are most popular as they enable cryptocurrencies like Bitcoin [56], Ethereum [81], etc., to operate in public-permissionless mode. New blocks in PoX are proposed via a distributed lottery, e.g., proof-of-work (PoW) [56], proof-of-stake (PoS) [63], proof-of-retrievability (PoR) [51] etc. The lottery winner (or *miner*) includes a zero-knowledge proof of their win within a new block, which the peers verify. Occasionally, two miners can mine a block referencing the same parent – resulting in a *fork* (see Figure 1). The participants eventually reach an agreement on transaction history through mechanisms such as longest chain, GHOST [71], etc. (green chain in Figure 1). Transactions in forks outside the main chain, a.k.a. *stale blocks* (orange chain), are disregarded and implicitly invalidated. To reduce the probability of stale forks and counter potential attacks like selfish-mining, double-spending, etc., [32, 48], participants can wait for additional *confirmations* before accepting a block. Note that despite these measures, blocks may still become stale if overtaken by a longer fork. Such blocks are referred to as *confirmed stale blocks*. On the other hand, *general stale blocks* are blocks outside of the main blockchain with at least *one* confirmation – the block itself. Such (transient) forks are a common occurrence in PoX blockchains and differ from occasional *hard forks* caused due to software updates [39]. Hard forks are not of interest to this study as they are rare and deliberate events outside of regular blockchain operations.

---

[1] We use "DLTs" and "blockchains" interchangeably for the sake of simplicity.

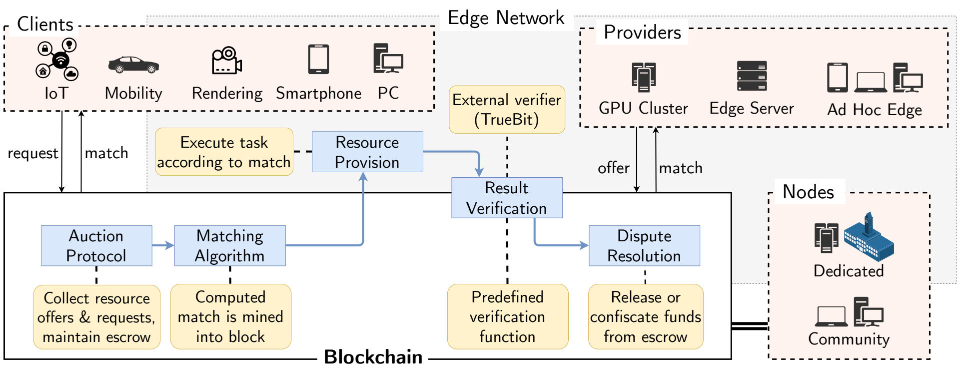


Figure 2: Devices, roles & activities in a blockchain-based edge computing marketplace.

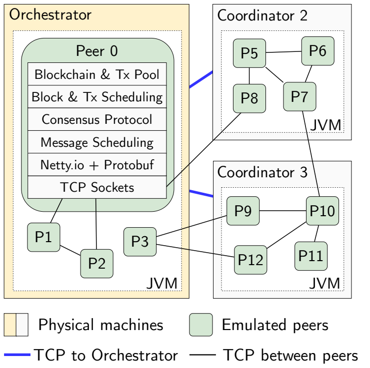


Figure 3: NEBULA emulations over three coordinators.

In contrast to PoX, blockchains utilizing traditional *byzantine fault tolerant* (BFT) protocols (e.g., Hyperledger Indy [77]) necessitate identifiable peers and permissioned access. BFT blockchains reach consensus after every block – sacrificing fault tolerance for scalability. *Hybrid or committee-based* blockchains improve upon these shortcomings by allowing peers to vote for delegates responsible for proposing blocks in fixed rounds. Delegates reach consensus using protocols like BFT (Cosmos [43]), pipelined BFT (EOS [14]), etc., with slightly relaxed finality. *DAG-based* DLTs depart from regular chain architectures, allowing blocks (SPECTRE [70]) or transactions (IOTA [76]) to reference multiple parents. To restore a consistent transaction history, the resulting partial ordering is then transformed into a total order algorithmically. However, such approaches lack maturity and cannot achieve similar performance as chain-based DLTs.

2.2 Blockchains for Resource Marketplaces

While researchers have employed blockchains for several dimensions of edge computing, its most popular use is to enable *decentralized crowdsourced marketplaces* [1, 42, 57, 78, 82, 86, 87]. Such marketplaces aim to facilitate a pervasive platform that integrates both managed and unmanaged resources at the edge. Figure 2 shows the internals of such a marketplace. *Resource providers*, including large organizations (ISPs, city admin) and individual operators, offer their hardware for an asking price. These resources are sought after by *clients/developers* to deploy their applications. The glue between the two parties is the *blockchain* that acts both as a decentralized auctioneer and as a payment portal. The blockchain can be supported by entities with dedicated resources (governments, etc.) or the community at large.

Several components are required for enabling such a marketplace to operate over blockchains, highlighted in blue. The *auction protocol* allows providers to advertise their hardware (as offers) and clients to "bid" for resources (as requests). The content of bids varies for different approaches [13, 86] but primarily includes resource valuation, costs, SLAs, etc. The *matching algorithm* is then responsible for finding optimal pairings between offers and requests, e.g., achieving maximum economic, QoS or runtime performance [73, 86]. Each match is treated as a new transaction, which is bundled with other matches and published as a block. Once bidders come to a conclusion on the best match, the *resource provisioning* allocates resources to the winning client for the agreed duration. To minimize overheads, provisioning can take place off-chain [46].

Since such a marketplace can include selfish actors, *verification* of final results to SLA in the original bid becomes necessary [78]. Operations like log verification [9], correctness checking [11], etc., can be carried out by incentivized external entities off-chain [75]. In case the verification indicates a dispute, the aggrieved party can initiate the *resolution* process over smart contracts, which may result in monetary or reputational punishments. While all components in Figure 2 are integral for enabling crowdsourced edge marketplaces over blockchains, only auctioning and matching operations are critically dependent on blockchains. Therefore, the wait time for gathering bids and matching them will strongly depend on underlying blockchain performance.

2.3 Blockchains as Plug-and-Play?

The majority of research encouraging an alliance between *edge* and *blockchains* has focused on "edge" as it provides more room for novel research solutions [38, 73]. Most of these approaches consider existing blockchain technologies as plug-and-play "blackboxes" [42, 86, 87]. The impact of potential overheads within blockchains on overall performance is either not considered or usually dismissed as out-of-scope.

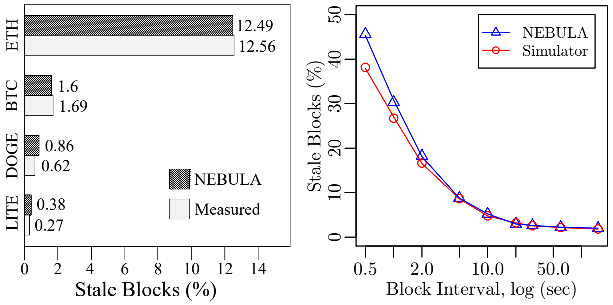
We argue that using DLTs to support edge computing is analogous to finding the right tool to hammer a nail. The utilities and trade-offs of each DLT variant differ significantly and *must* be taken into consideration when proposing solutions reliant on the technology [37]. Considering a DLT is a distributed networked system, its notable performance metrics are end-to-end latency and throughput, *in other words*, "time taken to mine and confirm a new block to the chain". Parameters, such as consensus mechanism and block sizes, can significantly affect these metrics at scale. However, similar to traditional networks, optimizing for both high throughput, and low latency is difficult (and almost contradictory) in DLTs. For example, Rainblock [59] plugs a performance bottleneck in Ethereum, allowing the technology to achieve ≈20K transactions per second. However, the solution packs 480× transactions in each block which increases the block size from 40-60 KB to 24 MB – resulting in a significant jump in propagation latency. On the other hand, next-generation applications care for both throughput and latency, with the latter being stricter of the two (most apps must operate under 100 ms) [52]. While many simulators and models evaluating different protocols for blockchains have been proposed in the past [5, 32, 68], the potential bottlenecks due to networking have remained largely unexplored in the literature. We attempt to bridge this gap by developing a configurable blockchain emulator called `NEBULA` , which allows us to investigate different blockchain technologies at scale.

3 BLOCKCHAINS UNDER MICROSCOPE

Figure 3 shows the internals of our `Java`-based NEtworked Blockchain emULAtor – `NEBULA`. We model blockchains as a network of virtualized peers that communicate via simple message-passing of `Protobuf` [2] over TCP. Each peer runs on a separate thread, allowing us to emulate n nodes on m coordinator machines ($n >> m$). One *coordinator* generates a network topology based on parameters such as the number of nodes, bandwidth, density and latency of the links, etc. Additionally, we also allow replicating networks from synthetic (e.g., random, scale-free [4]) and realistic (e.g., CAIDA AS [36], user-cloud connections [26]) datasets. Nodes in the graph are split between coordinator machines depending on their computing capabilities. The underlying network is modeled as TCP connections with application-level delays. The nodes are then instructed to establish interconnections to mimic the specified topology and collaborate as a blockchain. `NEBULA` is available publicly at [27].

In this work, we evaluate proof-based (PoX) and hybrid blockchain approaches (DPoS-BFT). While PoX is most popular, DPoS-BFT is the most promising performance-wise [7, 12]. For PoX, emulated peers create blocks randomly and concurrently at a configurable rate. For DPoS-BFT, `NEBULA` selects a subset of BPs that generate blocks at a constant rate in a round-robin fashion. The consensus protocol is *pipelined* BFT, where a block is published only if $^2/_3$ BPs verify the block. Consensus protocols in `NEBULA` are implemented as `Protobuf` message definitions allowing for easy extensions, e.g., Tendermint [17], with minimal overhead. The emulator also allows fine-tuning several other performance-affecting parameters, e.g., transaction rate, size, and fee, block verification time, number of confirmations, etc., to name a few.

Correctness. We emulate four popular PoX blockchains, specifically Bitcoin [24], Ethereum [5], Dogecoin, and Litecoin [32], with similar block rates and propagation delays as their real-world counterparts. We generate up to 10K blocks over multiple iterations and compare the total stale blocks to previous real-world measurements [5, 24, 32]. As noted in §2.1, generated stale blocks is a factor of consensus protocol and network latencies, and thus is an accurate representation of blockchain behavior. Figure 4a showcases that it is possible to emulate real-world blockchains in controlled settings through `NEBULA` . The slight deviation in accuracy is likely due to our lack of knowledge about other internal parameters of these blockchains. To address this, we compare `NEBULA` to a blockchain simulator [32]. Figure 4b shows that in similar networks with equal parameter settings, our emulations closely follow the simulations with a minimal error of 8.7%. Compared to the simulator, `NEBULA` is more fine-grained as it exposes additional configuration parameters and opens up the future potential for hardware-in-the-loop experiments.



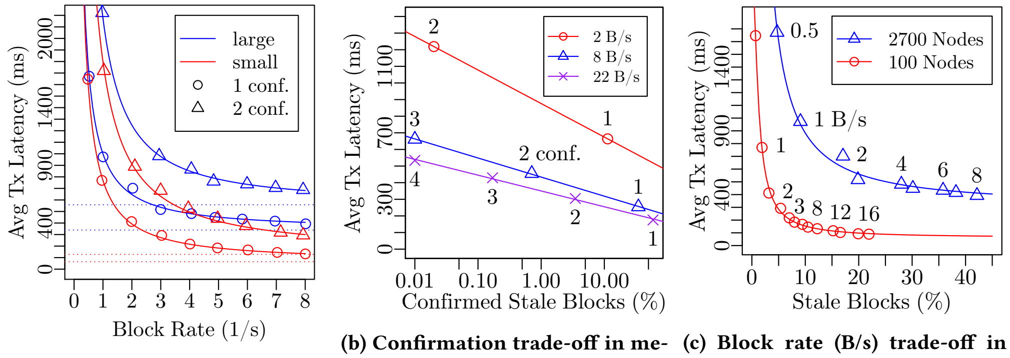
(a) Cryptocurrencies. **(b) Blockchain simulator.**

Figure 4: `NEBULA` correctness to (a) Ethereum, Bitcoin, Dogecoin & Litecoin, and (b) blockchain simulator.

3.1 Dissecting Blockchain Performance

We now attempt to empirically understand the possibility of employing blockchains for edge computing, especially for the model use-case of decentralized resource marketplaces (§2.2). Among all components, potential bottlenecks in blockchain's performance directly impact *auction* and *matching* stages as new auctions can only be entertained once existing ones are dealt with. In such an architecture, resource bids are sent to the auction protocol in the blockchain as traditional transactions that are added to a pool of unmatched bids. The matching is completed during block creation, and the block is finalized in accordance with the underlying consensus protocol (e.g., PoW, DPoS, etc.). The final block includes a pairwise match between all included transactions. The block – thus, the auction – is accepted by the clients once additional blocks extend it and *confirm* it as per the protocol. We use confirmation latency as our primary performance metric, i.e., the time elapsed between receiving resource bids and confirming the block that contains the match of those bids. Additionally, we measure the percentage of stale blocks and investigate their impact on consistency, efficiency, and fault-tolerance of auctions executed on the chain.

Setup Configuration. We conducted our experiments on a compute cluster of 15 Linux-based VMs, totaling ≈60 CPU cores. All VMs are interconnected by 1 Gbps Ethernet. We emulate three distinct blockchain network sizes for our experiments: *small*, *medium*, and *large*. Small and medium are scale-free networks of 100 and 500 nodes, respectively, with topology similar to Ethereum [80]. The small network follows Gaussian latency distribution with a mean latency of 10 ms (representative of a *smart city* [69]). Configurations of the medium and large network are inspired from user-to-cloud latencies and traversal measurements in [21]. The average end-to-end propagation delay in the medium network is set to 63 ms, similar to user to data center latencies in the study. Our large network is a topology of 2700 ASes residing between end-users and the cloud. Considering these ASes are the best contenders for potential edge server deployments, the network topology reflects an edge infrastructure spanning multiple organizations. Due to



(a) PoX at 1–2 confirmations.

(b) Confirmation trade-off in medium network.

(c) Block rate (B/s) trade-off in small and large networks.

Figure 5: Latencies and stale block trade-offs in PoX at different network sizes.

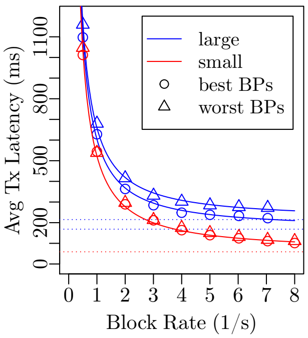


Figure 6: DPoS-BFT latency for 21 best and worst BPs.

the comparatively high block rates of our experiments, individual blocks are smaller and contain only a few transactions – allowing us to focus on the accuracy of our latency measurements without exhausting bandwidth limits. We conduct multiple iterations for each experiment, generating up to 30K blocks per iteration.

Proof-based Blockchains (PoX). We first evaluate the suitability of PoX blockchains for supporting edge marketplace tasks. Figure 5a shows how an increase in the frequency of blockchain-backed auctions (translating to increasing block rate) affects overall transaction latency. The dashed horizontal lines are the limits of the fitted Weibull functions. At first glance, it appears that PoX blockchains can effectively support high-frequency auctions at the edge, as the latency decreases to an asymptotic limit with increasing block rate. Naturally, as the block generation rate rises, the waiting time for transactions to be included in the next block approaches zero, and the remaining latency is primarily due to network propagation. However, higher rates lead to higher concurrency in block creations and, thus, increased stale block percentages (see Figure 5c). More stale blocks in the chain not only imply wastage of computational resources but also a possibility of invalid auctions – as the same bid can be included in multiple concurrent blocks.

The number of confirmed stale blocks in PoX can be reduced by waiting for additional confirmations (see §2.1). As expected, Figure 5b shows that stale block percentages decrease exponentially if more confirmations are required before accepting the block. However, the additional wait time for confirmations also results in an increase in overall latency. Note that the confirmations do not affect *general stale blocks* in PoX (denoted by stale blocks with one confirmation in Figures 5b and 5c). Moreover, the trade-off between latency and stale blocks increases with network size (see Figure 5c). We find that for larger networks (>2K nodes), even with the most lenient settings, PoX blockchains add ≈ 500 ms overhead to edge marketplace operation (see Figure 5a) – far exceeding the 60-100 ms desired threshold [52].

Hybrid Blockchains (DPoS-BFT). By design, delegated proof-of-stake with BFT (DPoS-BFT) avoids waiting for additional confirmations by centralizing consensus to a subset of block producers (BP). BPs are elected by the network through stake-based voting [79]. Figure 6 shows the transaction latency of DPoS-BFT at different block rates and network sizes. Here, we restrict the consensus to 21 BPs (default in EOS [14]) and show the results when BPs are the best or the worst connected nodes in the network.

Similar to PoX, transaction latency in DPoS-BFT is primarily influenced by network sizes (see Figure 6). While optimal BP selection can reduce latencies in large networks, it has a negligible effect in smaller networks. Our result, however, indicates that in small networks, DPoS-BFTs can support < 100 ms latencies at high block rates – showing some promise for edge-based applications. However, while DPoS-BFT ensures that only one BP proposes blocks at any given time, stale blocks can still occur in the chain. The primary contributor is network propagation delay, as BP responsibility changes between nodes at the end of each round. Therefore, preventive mechanisms, like increasing the time allocated for the last block per BP, are still needed in DPoS-BFT – which will result in additional latencies. Furthermore, as noted in Table 1, the fault-tolerance of DPoS-BFT is lower than PoX, and the blockchain security can be attacked with 33% BP control (which is quite possible in relatively small networks). Nevertheless, in terms of performance, a local DPoS-BFT blockchain seems to be most promising for our use-case.

4 DISCUSSION

Our results show that it is largely unclear if blockchains can be proponents for crowdsourced edge computing. While the performance differs for different consensus technologies, achieving the coveted 60-100 ms end-to-end latency [52] is usually not possible for blockchains spanning large networks. Additionally, one must sacrifice between latency (confirmations) or processing cycles when dealing with stale blocks. Due to stale forks, expensive roll-back mechanisms [16] would still be required in edge marketplaces to avoid monetary losses. Here, two things must be noted to appreciate our results. *First*, while we discuss our results in context to edge computing requirements, our experiment design is fairly generic, and our results are also relevant for understanding blockchain networking bottlenecks. *Second*, the transaction latency in §3.1 does not include the processing overhead of the auction/matching protocol itself, which is also known to be significant [30]. We do not empirically explore alternate blockchains like DAGs in this work since recent measurements report their confirmation latencies to last several minutes [35] – making them unsuitable for edge.

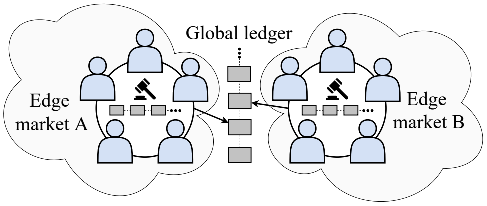


Figure 7: Sharded blockchains for edge computing.

Despite its popularity in literature [42, 86], propagating and confirming local edge transactions through a global blockchain network within stringent latency bounds is an uphill battle. While recent developments such as ZK- or Optimistic-rollups [34] attempt to improve scalability, adding consensus from a global network to the critical path of an edge transaction is a massive operational overhead. In small networks, DLTs show some promise for decentralized computing (Figures 5a and 6). However, restricting the size of the network is tough in reality. One possible way is through fully-permissioned DLTs, e.g., HotStuff [85], which restricts participation to only a few pre-approved nodes (likely to be under the same management – city administration, industry, etc.). While several factors favor such a deployment in the future, the primary benefits of DLTs (trust, privacy, openness) are no longer present in such gated participation policies.

Another solution would be to collocate DLTs alongside edge networks. Figure 7 shows a fragmented (a.k.a. *sharded*) blockchain which localizes data and task operation to different edge markets distributed across geography. Still, each sharded blockchain cannot be permissionless in nature, as access needs to be restricted within geography. Localized blockchain markets are lazily synchronized by a global *interledger* [31, 67]. Applications can operate in local markets and are administered globally via the interledger. Such a blockchain can concurrently support highly frequent localized tasks and occasional global operations with minimal latency overhead. Interestingly, the approach of blockchain sharding describes a hybrid centralization solution. While general membership is retained to smaller regions using a "centralized" third party (i.e., city admin, interledger), some decentralization in control/accountability is retained by individual members who need not necessarily trust each other. Still, significant research effort is required to understand the trade-offs in such a blockchain architecture, especially for tasks over multiple shards and potential double-spends [67].

Practical relevance of blockchains & crowdsourcing. Irrespective of the technical challenges in supporting edge marketplaces over DLTs, the relevance of both these domains, in reality, remains to be seen. In the past 5-6 years, the interest and knowledge about blockchains in the general public have increased significantly as cryptocurrencies like Bitcoin and Ethereum, are now accepted as legitimate tenders [10, 45]. Crowds are more open to participating as miners of new DLTs to invest early as coin holders. Even without reward mechanisms, crowdsourcing as a concept is now more accepted, as evident from the 1200% surge in Folding@Home contributors in 2020 [19]. On the other hand, cloud providers, such as Amazon and Google, are dominating the computing market and are investing significantly to expand their network reach globally [21]. Many have also established specialized servers within ISP facilities – allowing them to reduce latencies while retaining network control [50]. Interestingly, many cloud players also have a significant presence in the edge as manufacturers and operators of smartphones, smart devices, etc. By incorporating "user-owned-but-manufacturer-operated" hardware with existing cloud network, these organizations have the unique opportunity to build a hybrid crowdsourced infrastructure without involving blockchains. On the other hand, efforts to dethrone the stronghold of cloud providers and usher in the era of decentralized computing are strongly backed by independent investors. For example, "Internet computer", a blockchain-based non-proprietary compute fabric from Dfinity raised $195M [74] and was launched earlier this year [25].

Security. Our results show that blockchains can only support latencies for edge tasks in smaller networks (typically implying lower node counts), which directly inhibits its own paradigm of success – *strength-in-numbers*. By restricting the network size, inherent benefits of DLTs, e.g., security, privacy and trust, start to crumble and avenues for novel attack vectors open up while existing security loopholes are further magnified [6]. For example, 51% attacks over PoX are now realistically achievable [32], and DPoS blockchains, which are already vulnerable to malicious cartels influencing its voting procedure [83], are prone to complete takeovers. Permissioned blockchains under a central authority might overcome several such security challenges, but they come at the cost of trust and privacy.

Other Applications. Despite potential drawbacks in latency, other edge computing use cases not reliant on network delay should be mentioned [84]. Here, decentralized storage systems [62], weather sensors [41], or certain smart home/city applications [52] (i.e., bus timetables, smart parking meters) have been proposed in the past. While these applications are located outside of edge latency feasibility zones [52], they could still benefit from higher bandwidth and localized traffic as promised by edge computing. Due to the decentralized and distributed nature of edge in general, building a self-sufficient ecosystem on top of a secure and trustless DLT platform is undoubtedly enticing.

5 CONCLUSION

In this paper, we investigated the rationale behind crowdsourced edge computing backed by distributed ledgers. We identified popular approaches in crowdsourced marketplace research that rely on blockchain for operation. We designed the generic, scalable and configurable blockchain emulator `NEBULA` to shed light on the internal overheads of proof-based and hybrid blockchain technologies. We highlight several research directions which might make DLTs fit for the edge, albeit with potential limitations, such as sharded interledgers. *In conclusion*, we believe that there are several challenges to be tackled for enabling distributed edge over blockchains. Future research in this direction *must not* blindly offload performance-critical functionality to be handled by DLTs.

ACKNOWLEDGMENTS

This work was supported by the Swedish Foundation for Strategic Research with grant number GMT-14-0032 (Future Factories in the Cloud) and EU Celtic project Piccolo (C2019/2-2).

REFERENCES

[1] 2021. *iExec - The First Decentralized Marketplace for Cloud Resources.* https://iex.ec/.

[2] 2021. *Protocol Buffers.* https://developers.google.com/protocol-buffers.

[3] Nasir Abbas, Yan Zhang, Amir Taherkordi, and Tor Skeie. 2018. Mobile Edge Computing: A Survey. *IEEE Internet of Things Journal* 5, 1 (Feb 2018), 450–465. https://doi.org/10.1109/JIOT.2017.2750180

[4] Réka Albert and Albert-László Barabási. 2002. Statistical mechanics of complex networks. *Reviews of Modern Physics* 74, 1 (Jan 2002), 47–97. https://doi.org/10.1103/revmodphys.74.47

[5] Maher Alharby and Aad van Moorsel. 2020. BlockSim: An Extensible Simulation Tool for Blockchain Systems. *Frontiers in Blockchain* 3 (Jun 2020). https://doi.org/10.3389/fbloc.2020.00028

[6] Muneeb Ali, Jude Nelson, Ryan Shea, and Michael J. Freedman. 2016. Blockstack: A Global Naming and Storage System Secured by Blockchains. In *Proceedings of the 2016 USENIX Conference on Usenix Annual Technical Conference* (Denver, CO, USA) *(USENIX ATC '16)*. USENIX Association, USA, 181–194.

[7] Salem Alqahtani and Murat Demirbas. 2021. Bottlenecks in Blockchain Consensus Protocols. arXiv:2103.04234 https://arxiv.org/abs/2103.04234

[8] Ganesh Ananthanarayanan, Paramvir Bahl, Peter Bodík, Krishna Chintalapudi, Matthai Philipose, Lenin Ravindranath, and Sudipta Sinha. 2017. Real-time video analytics: The killer app for edge computing. *computer* 50, 10 (2017), 58–67.

[9] Atakan Aral, Rafael Brundo Uriarte, Anthony Simonet-Boulogne, and Ivona Brandic. 2020. Reliability Management for Blockchain-Based Decentralized Multi-Cloud. In *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*. https://doi.org/10.1109/CCGrid49817.2020.00-91

[10] Tomaso Aste. 2019. Cryptocurrency market structure: connecting emotions and economics. *Digital Finance* 1, 1 (2019), 5–21.

[11] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. 2014. Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture. In *23rd USENIX Security Symposium (USENIX Security 14)*. USENIX Association, San Diego, CA, 781–796. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/ben-sasson

[12] Sofiane Benahmed, Ivan Pidikseev, Rasheed Hussain, JooYoung Lee, S.M. Ahsan Kazmi, Alma Oracevic, and Fatima Hussain. 2019. A Comparative Analysis of Distributed Ledger Technologies for Smart Contract Development. In *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. 1–6. https://doi.org/10.1109/PIMRC.2019.8904256

[13] Erik-Oliver Blass and Florian Kerschbaum. 2018. Strain: A Secure Auction for Blockchains. In *Computer Security*, Javier Lopez, Jianying Zhou, and Miguel Soriano (Eds.). Springer International Publishing, Cham, 87–110.

[14] block.one. 2018. *EOS.IO Technical White Paper v2.* https://github.com/EOSIO/Documentation.

[15] Andreas Bogner, Mathieu Chanson, and Arne Meeuw. 2016. A Decentralised Sharing App Running a Smart Contract on the Ethereum Blockchain. In *Proceedings of the 6th International Conference on the Internet of Things* (Stuttgart, Germany) *(IoT'16)*. Association for Computing Machinery, New York, NY, USA, 177–178. https://doi.org/10.1145/2991561.2998465

[16] Vincenzo Botta, Daniele Friolo, Daniele Venturi, and Ivan Visconti. 2019. Shielded Computations in Smart Contracts Overcoming Forks. Cryptology ePrint Archive, Report 2019/891. https://eprint.iacr.org/2019/891.

[17] Ethan Buchman, Jae Kwon, and Zarko Milosevic. 2018. The latest gossip on BFT consensus. *CoRR* abs/1807.04938. http://arxiv.org/abs/1807.04938

[18] Youdong Chen, Qiangguo Feng, and Weisong Shi. 2018. An industrial robot system based on edge computing: An early experience. In *{USENIX} Workshop on Hot Topics in Edge Computing (HotEdge 18)*.

[19] CoinTelegraph. 2020. FoldingHome Surpasses 400,000 Users Amid Crypto Contribution. (2020). https://cointelegraph.com/news/foldinghome-surpasses-400-000-users-amid-crypto-contribution.

[20] Lorenzo Corneo, Maximilian Eder, Nitinder Mohan, Aleksandr Zavodovski, Suzan Bayhan, Walter Wong, Per Gunningberg, Jussi Kangasharju, and Jörg Ott. 2021. Surrounded by the Clouds: A Comprehensive Cloud Reachability Study. In *Proceedings of The Web Conference 2021 (WWW '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3442381.3449854

[21] Lorenzo Corneo, Maximilian Eder, Nitinder Mohan, Aleksandr Zavodovski, Suzan Bayhan, Walter Wong, Per Gunningberg, Jussi Kangasharju, and Jörg Ott. 2021. Surrounded by the Clouds: A Comprehensive Cloud Reachability Study. https://doi.org/10.1145/3442381.3449854

[22] Lorenzo Corneo, Nitinder Mohan, Aleksandr Zavodovski, Walter Wong, Christian Rohner, Per Gunningberg, and Jussi Kangasharju. 2021. (How Much) Can Edge Computing Change Network Latency?. In *2021 IFIP Networking Conference (IFIP Networking)*. IEEE, 1–9.

[23] The Khang Dang, Nitinder Mohan, Lorenzo Corneo, Aleksandr Zavodovski, Jörg Ott, and Jussi Kangasharju. 2021. Cloudy with a Chance of Short RTTs: Analyzing Cloud Connectivity in the Internet. In *Proceedings of Internet Measurement Conference (IMC '21)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3487552.3487854

[24] C. Decker and R. Wattenhofer. 2013. Information propagation in the Bitcoin network. In *IEEE P2P 2013 Proceedings*. 1–10. https://doi.org/10.1109/P2P.2013.6688704

[25] Dfinity. 2021. Internet Computer Genesis. (2021). https://dfinity.org/mercury.

[26] Maximilian Eder, Lorenzo Corneo, Nitinder Mohan, Aleksandr Zavodovski, Suzan Bayhan, Walter Wong, Per Gunningberg, Jussi Kangasharju, and Jörg Ott. 2021. Surrounded by the Clouds. https://doi.org/10.14459/2020mp1593899 https://mediatum.ub.tum.de/1593899.

[27] Leo Eichhorn, Tanya Shreedhar, Aleksandr Zavodovski, and Nitinder Mohan. 2021. NEBULA: NEtworked Blockchain emULAtor. (2021). https://github.com/LeoEichhorn/NEBULA.

[28] Mohammed S Elbamby, Cristina Perfecto, Mehdi Bennis, and Klaus Doppler. 2018. Toward low-latency and ultra-reliable virtual reality. *IEEE Network* 32, 2 (2018), 78–84.

[29] Md Sadek Ferdous, Mohammad Jabed Morshed Chowdhury, and Mohammad A. Hoque. 2021. A survey of consensus algorithms in public blockchain systems for crypto-currencies. *Journal of Network and Computer Applications* 182 (2021), 103035. https://doi.org/10.1016/j.jnca.2021.103035

[30] Guoju Gao, Mingjun Xiao, Jie Wu, He Huang, Shengqi Wang, and Guoliang Chen. 2019. Auction-based VM allocation for deadline-sensitive tasks in distributed edge cloud. *IEEE Transactions on Services Computing* (2019).

[31] Alberto Garoffolo, Dmytro Kaidalov, and Roman Oliynykov. 2020. Zendoo: a zk-SNARK Verifiable Cross-Chain Transfer Protocol Enabling Decoupled and Decentralized Sidechains. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. 1257–1262. https://doi.org/10.1109/ICDCS47774.2020.00161

[32] Arthur Gervais, Ghassan O. Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. 2016. On the Security and Performance of Proof of Work Blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (Vienna, Austria) *(CCS '16)*. Association for Computing Machinery, New York, NY, USA, 3–16. https://doi.org/10.1145/2976749.2978341

[33] Fabio Giust, Xavier Costa-Perez, and Alex Reznik. 2017. Multi-access edge computing: An overview of ETSI MEC ISG. *IEEE 5G Tech Focus* 1, 4 (2017), 4.

[34] Alex Gluchowski. 2020. Optimistic vs. ZK Rollup: deep dive.

[35] Fengyang Guo, Xun Xiao, Artur Hecker, and Schahram Dustdar. 2020. Characterizing IOTA Tangle with Empirical Data. In *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*. 1–6. https://doi.org/10.1109/GLOBECOM42002.2020.9322220

[36] Bradley Huffaker, Marina Fomenkov, et al. 2012. *Internet topology data comparison.* Technical Report. Cooperative Association for Internet Data Analysis (CAIDA).

[37] Niclas Kannengießer, Sebastian Lins, Tobias Dehling, and Ali Sunyaev. 2020. Trade-Offs between Distributed Ledger Technology Characteristics. *ACM Comput. Surv.* 53, 2, Article 42 (May 2020), 37 pages. https://doi.org/10.1145/3379463

[38] A. Kiani and N. Ansari. 2017. Toward Hierarchical Mobile Edge Computing: An Auction-Based Profit Maximization Approach. *IEEE Internet of Things Journal* 4, 6 (Dec 2017), 2082–2091. https://doi.org/10.1109/JIOT.2017.2750030

[39] Lucianna Kiffer, Dave Levin, and Alan Mislove. 2017. Stick a fork in it: Analyzing the Ethereum network partition. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*. 94–100.

[40] Stéphane Kündig, Constantinos Marios Angelopoulos, Sanmukh R. Kuppannagari, José Rolim, and Viktor K. Prasanna. 2020. Crowdsourced Edge: A Novel Networking Paradigm for the Collaborative Community. In *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. 474–481. https://doi.org/10.1109/DCOSS49796.2020.00080

[41] Tanesh Kumar, An Braeken, Vidhya Ramani, Ijaz Ahmad, Erkki Harjula, and Mika Ylianttila. 2019. SEC-BlockEdge: Security Threats in Blockchain-Edge based Industrial IoT Networks. In *2019 11th International Workshop on Resilient Networks Design and Modeling (RNDM)*. 1–7. https://doi.org/10.1109/RNDM48015.2019.8949107

[42] T. Kumar, E. Harjula, M. Ejaz, A. Manzoor, P. Porambage, I. Ahmad, M. Liyanage, A. Braeken, and M. Ylianttila. 2020. BlockEdge: Blockchain-Edge Framework for Industrial IoT Networks. *IEEE Access* 8 (2020), 154166–154185. https://doi.org/10.1109/ACCESS.2020.3017891

[43] J. Kwon and E. Buchman. 2021. Cosmos Whitepaper. 1 (2021). https://v1.cosmos.network/resources/whitepaper.

[44] Maria A. Lema, Andrés Laya, Toktam Mahmoodi, Maria Cuevas, Joachim Sachs, Jan Markendahl, and Mischa Dohler. 2017. Business Case and Technology Analysis for 5G Low Latency Applications. *IEEE Access* 5 (2017), 5917–5935. https://doi.org/10.1109/ACCESS.2017.2685687

[45] Jiaqi Liang, Linjing Li, and Daniel Zeng. 2018. Evolutionary dynamics of cryptocurrency transaction networks: An empirical study. *PloS one* 13, 8 (2018), e0202202.

[46] Anil Madhavapeddy, Thomas Leonard, Magnus Skjegstad, Thomas Gazagnaire, David Sheets, Dave Scott, Richard Mortier, Amir Chaudhry, Balraj Singh, Jon Ludlam, Jon Crowcroft, and Ian Leslie. 2015. Jitsu: Just-In-Time Summoning of Unikernels. (05 2015).

[47] Yuyi Mao, Changsheng You, Jun Zhang, Kaibin Huang, and Khaled B Letaief. 2017. A survey on mobile edge computing: The communication perspective. *IEEE*

Communications Surveys & Tutorials 19, 4 (2017), 2322–2358.

[48] Diego Marmsoler and Leo Eichhorn. 2020. On the impact of architecture design decisions on the quality of blockchain-based applications. *The Knowledge Engineering Review* 35 (2020), e24. https://doi.org/10.1017/S0269888920000193

[49] Patrick McCorry, S. F. Shahandashti, and F. Hao. 2017. A Smart Contract for Boardroom Voting with Maximum Voter Privacy. In *IACR Cryptol. ePrint Arch.*

[50] Microsoft. 2021. Azure Stack Edge. (2021). https://azure.microsoft.com/en-us/products/azure-stack/edge/.

[51] A. Miller, A. Juels, E. Shi, B. Parno, and J. Katz. 2014. Permacoin: Repurposing Bitcoin Work for Data Preservation. In *2014 IEEE Symposium on Security and Privacy*. 475–490. https://doi.org/10.1109/SP.2014.37

[52] Nitinder Mohan, Lorenzo Corneo, Aleksandr Zavodovski, Suzan Bayhan, Walter Wong, and Jussi Kangasharju. 2020. Pruning Edge Research with Latency Shears. In *Proceedings of the 19th ACM Workshop on Hot Topics in Networks* (Virtual Event, USA) *(HotNets '20)*. Association for Computing Machinery, New York, NY, USA, 182–189. https://doi.org/10.1145/3422604.3425943

[53] Nitinder Mohan and Jussi Kangasharju. 2016. Edge-Fog cloud: A distributed cloud for Internet of Things computations. In *2016 Cloudification of the Internet of Things (CIoT)*. IEEE, 1–6.

[54] Nitinder Mohan, Tanya Shreedhar, Aleksandr Zavodavoski, Otto Waltari, Jussi Kangasharju, and Sanjit K Kaul. 2018. Redesigning mptcp for edge clouds. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 675–677.

[55] Nitinder Mohan, Tanya Shreedhar, Aleksandr Zavodovski, Jussi Kangasharju, and Sanjit K Kaul. 2019. Is two greater than one?: Analyzing Multipath TCP over Dual-LTE in the Wild. *arXiv preprint arXiv:1909.02601* (2019).

[56] Satoshi Nakamoto. 2019. *Bitcoin: A peer-to-peer electronic cash system.* Technical Report. Manubot.

[57] Matteo Nardini, Sven Helmer, Nabil El Ioini, and Claus Pahl. 2020. A Blockchain-based Decentralized Electronic Marketplace for Computing Resources. *SN Computer Science* 1 (08 2020). https://doi.org/10.1007/s42979-020-00243-7

[58] Quoc-Viet Pham, Fang Fang, Vu Nguyen Ha, Md. Jalil Piran, Mai Le, Long Bao Le, Won-Joo Hwang, and Zhiguo Ding. 2020. A Survey of Multi-Access Edge Computing in 5G and Beyond: Fundamentals, Technology Integration, and State-of-the-Art. *IEEE Access* 8 (2020), 116974–117017. https://doi.org/10.1109/ACCESS.2020.3001277

[59] Soujanya Ponnapalli, Aashaka Shah, Amy Tai, Souvik Banerjee, Vijay Chidambaram, Dahlia Malkhi, and Michael Wei. 2020. Rainblock: Faster Transaction Processing in Public Blockchains. arXiv:1909.11590 [cs.DC]

[60] Jorge Peña Queralta and Tomi Westerlund. 2020. Blockchain for Mobile Edge Computing: Consensus Mechanisms and Scalability. *CoRR* abs/2006.07578 (2020). arXiv:2006.07578 https://arxiv.org/abs/2006.07578

[61] Aravind Ramachandran and Murat Kantarcioglu. 2017. Using Blockchain and smart contracts for secure data provenance management. arXiv:1709.10000 http://arxiv.org/abs/1709.10000

[62] Yongjun Ren, Yan Leng, Yaping Cheng, and Jin Wang. 2019. Secure data storage based on blockchain and coding in edge computing. *Math. Biosci. Eng* 16, 4 (2019), 1874–1892.

[63] Fahad Saleh. 2020. Blockchain without Waste: Proof-of-Stake. *The Review of Financial Studies* 34, 3 (07 2020), 1156–1190. https://doi.org/10.1093/rfs/hhaa075 arXiv:https://academic.oup.com/rfs/article-pdf/34/3/1156/36264599/hhaa075_supplementary_data.pdf

[64] Mahadev Satyanarayanan. 2017. The emergence of edge computing. *Computer* 50, 1 (2017), 30–39.

[65] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. 2016. Edge computing: Vision and challenges. *IEEE internet of things journal* 3, 5 (2016), 637–646.

[66] Tanya Shreedhar, Sanjit K Kaul, and Roy D Yates. 2019. An age control transport protocol for delivering fresh updates in the Internet-of-Things. In *2019 IEEE 20th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 1–7.

[67] Vasilios A. Siris, Pekka Nikander, Spyros Voulgaris, Nikos Fotiou, Dmitrij Lagutin, and George C. Polyzos. 2019. Interledger Approaches. *IEEE Access* 7 (2019), 89948–89966. https://doi.org/10.1109/ACCESS.2019.2926880

[68] S. Smetanin, A. Ometov, N. Kannengießer, B. Sturm, M. Komarov, and A. Sunyaev. 2020. Modeling of Distributed Ledgers: Challenges and Future Perspectives. In *2020 IEEE 22nd Conference on Business Informatics (CBI)*, Vol. 1. 162–171. https://doi.org/10.1109/CBI49978.2020.00025

[69] David Soldani, Y. Jay Guo, Bernard Barani, Preben Mogensen, Chih-Lin I, and Sajal K. Das. 2018. 5G for Ultra-Reliable Low-Latency Communications. *IEEE Network* 32, 2 (March 2018), 6–7. https://doi.org/10.1109/MNET.2018.8329617

[70] Yonatan Sompolinsky, Yoad Lewenberg, and Aviv Zohar. 2016. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. *IACR Cryptol. ePrint Arch.* 2016 (2016), 1159.

[71] Yonatan Sompolinsky and Aviv Zohar. 2015. Secure High-Rate Transaction Processing in Bitcoin. In *Financial Cryptography and Data Security*, Rainer Böhme and Tatsuaki Okamoto (Eds.). Springer Berlin Heidelberg, 507–527.

[72] Nick Szabo. 1996. Smart contracts: building blocks for digital markets. *EXTROPY: The Journal of Transhumanist Thought,(16)* 18, 2 (1996).

[73] A. G. Tasiopoulos, O. Ascigil, I. Psaras, and G. Pavlou. 2018. Edge-MAP: Auction Markets for Edge Resource Provisioning. In *2018 IEEE 19th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. 14–22. https://doi.org/10.1109/WoWMoM.2018.8449792

[74] TechCrunch. 2018. DFINITY raises $102M for a decentralised 'Internet Computer'. (2018). https://techcrunch.com/2018/08/29/dfinity/.

[75] Jason Teutsch and Christian Reitwießner. 2019. A scalable verification solution for blockchains. *CoRR* abs/1908.04756 (2019). arXiv:1908.04756 http://arxiv.org/abs/1908.04756

[76] The IOTA Foundation. 2020. The Coordicide. 1 (2020). https://files.iota.org/papers/20200120_Coordicide_WP.pdf.

[77] The Linux Foundation. 2017. Hyperledger Architecture. (2017). https://www.hyperledger.org/.

[78] Mu Wang, Changqiao Xu, Xingyan Chen, Lujie Zhong, Zhonghui Wu, and Dapeng Oliver Wu. 2021. BC-Mobile Device Cloud: A Blockchain-Based Decentralized Truthful Framework for Mobile Device Cloud. *IEEE Transactions on Industrial Informatics* 17, 2 (2021), 1208–1219. https://doi.org/10.1109/TII.2020.2983209

[79] Qi Wang, Ming Xu, Xiangxue Li, and Haifeng Qian. 2020. Revisiting the Fairness and Randomness of Delegated Proof of Stake Consensus Algorithm. In *2020 IEEE Intl Conf on Parallel Distributed Processing with Applications, Big Data Cloud Computing, Sustainable Computing Communications, Social Computing Networking (ISPA/BDCloud/SocialCom/SustainCom)*. 305–312. https://doi.org/10.1109/ISPA-BDCloud-SocialCom-SustainCom51426.2020.00064

[80] Taotao Wang, Chonghe Zhao, Qing Yang, Shengli Zhang, and Soung Chang Liew. 2021. Ethna: Analyzing the Underlying Peer-to-Peer Network of the Ethereum Blockchain. arXiv:2010.01373 [cs.NI]

[81] Gavin Wood et al. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper* 151, 2014 (2014), 1–32.

[82] K. Wright, M. Martinez, U. Chadha, and B. Krishnamachari. 2018. Smart-Edge: A Smart Contract for Edge Computing. In *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 1685–1690. https://doi.org/10.1109/Cybermatics_2018.2018.00281

[83] Brent Xu, Dhruv Luthra, Zak Cole, and Nate Blakely. 2018. EOS: An Architectural, Performance, and Economic Analysis. (2018).

[84] Ruizhe Yang, F. Richard Yu, Pengbo Si, Zhaoxin Yang, and Yanhua Zhang. 2019. Integrated Blockchain and Edge Computing Systems: A Survey, Some Research Issues and Challenges. *IEEE Communications Surveys Tutorials* 21, 2 (2019), 1508–1532. https://doi.org/10.1109/COMST.2019.2894727

[85] Maofan Yin, Dahlia Malkhi, Michael K. Reiter, Guy Golan Gueta, and Ittai Abraham. 2019. HotStuff: BFT Consensus with Linearity and Responsiveness. In *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing* (Toronto ON, Canada) *(PODC '19)*. Association for Computing Machinery, New York, NY, USA, 347–356. https://doi.org/10.1145/3293611.3331591

[86] A. Zavodovski, S. Bayhan, N. Mohan, P. Zhou, W. Wong, and J. Kangasharju. 2019. DeCloud: Truthful Decentralized Double Auction for Edge Clouds. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. 2157–2167. https://doi.org/10.1109/ICDCS.2019.00212

[87] Aleksandr Zavodovski, Nitinder Mohan, Walter Wong, and Jussi Kangasharju. 2019. Open Infrastructure for Edge: A Distributed Ledger Outlook. In *2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 19)*. USENIX Association, Renton, WA. https://www.usenix.org/conference/hotedge19/presentation/zavodovski

[88] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Weili Chen, Xiangping Chen, Jian Weng, and Muhammad Imran. 2020. An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems* 105 (2020), 475–491. https://doi.org/10.1016/j.future.2019.12.019