Unchain Your Blockchain

Tamraparni Dasu
AT&T Labs-Research
tamr@research.att.com

Yaron Kanza
AT&T Labs-Research
kanza@research.att.com

Divesh Srivastava
AT&T Labs-Research
divesh@research.att.com

ABSTRACT

Blockchain is emerging as a preeminent decentralized ledger and receiving increasing attention from researchers, practitioners, organizations and the public. Initially, blockchain was developed to address the "double spending" problem in cryptocurrencies, but recently, many new applications of blockchain have been proposed or are being developed. Blockchain allows sharing data in a decentralized, transparent and immutable way, using a peer-to-peer network, without the need to trust any particular entity. To achieve that in public blockchain, where the peers are *a priori* unknown, efficiency and scalability are often sacrificed.

In this paper we present a novel partition of the blockchain into smaller chains, to allow association of sub-chains, wallets and transactions with real-world concepts, such as geographical areas, and by this, improve scalability and security. Our contribution is threefold. First, we discuss the utilization of a real-world hierarchical structure, such as a geospatial subdivision, to partition the ledger into a tree of connected blockchains, in order to increase scalability and provide a tradeoff between privacy and transaction latency. Second, we illustrate the use of a geospatial partitioning to support geofencing, in order to add security to cryptocurrencies and other blockchain applications. Third, we present *proof-of-location* as an alternative to proof-of-work, to cope with the large waste of energy caused by proof-of-work, which may be inflated by the partitioning.

CCS Concepts

•**Security and privacy** → **Distributed systems security;** •**Information systems** → *Spatial-temporal systems;* •**Computing methodologies** → Distributed algorithms;

Keywords

Blockchain, Proof-of-Location, PoL, decentralized ledger, hierarchical partitioning, immutable storage, cryptocurrency, Bitcoin, geofencing, localized ledger

Symposium on Foundations and Applications of Blockchain (FAB '18) March 9, 2018, Los Angeles, California, USA.

1. INTRODUCTION

Blockchain and cryptocurrencies have a growing economic influence. They are beginning to revolutionize industries [36, 37], and are considered by many to be a *game changer* in areas like finance, insurance, notary, copyright protection, distribution of digital arts, and so on. Hence, they are receiving a growing attention from researchers and practitioners.

Blockchain is a decentralized public ledger that was initially introduced as a solution to the "double spending" problem in cryptocurrencies like Bitcoin [25]. It provides an immutable storage of transactions in a chain of blocks. The chain is created in a decentralized fashion by peers, using a peer-to-peer network, without any central node to govern it or enforce rules. The chain structure provides a serialization of the stored transactions, to prevent double spending.

Recently cryptocurrencies have flourished, and in particular, the importance of Bitcoin has increased, as it becomes an acceptable method of payment to a growing number of organizations and companies. Cryptocurrencies facilitate micropayments, provide anonymity to both the payer and the payee, and lay the basis for an economy without regulation. This challenges the traditional economic order [40].

Blockchain is receiving growing attention not just as the underlying technology of cryptocurrencies, but also as a public ledger in various domains, as elaborated below.

- **Financial transactions:** Financial institutions are examining the use of blockchain as a ledger for financial transactions, to cut out the middleman to reduce costs and expedite the processing of transactions [39].

- **Digital assets:** Blockchain can be used to maintain digital assets such as stocks, bonds, land titles, etc. Stored transactions record the transfer of assets between users [10].

- **Evidence of data and documents:** The blockchain stores data and documents, either in full or merely a digest of the data (e.g., using a cryptographic hash like SHA-256). The aim is to provide an evidence of the existence of data or documents, such as contracts, patents, scientific publications, deeds, insurance policies, etc. [36].

- **Identity management:** Using blockchain for identity management is examined [2, 3]. Hashed features of a person (digest of verifiable attributes of the person) are stored with a public key or some other means to electronically sign documents or access remote ser-

vices. The aim is to protect people from identity theft and fraudulent impersonation.

- **Sharing data:** Blockchain has the potential to provide a secure infrastructure for smart cities [8, 35], and could facilitate the creation of a marketplace of social data [21] where people share their private data for public benefit.
- **Commercial use:** Blockchain-based applications are developed for tracking diamonds from the mines to the market, managing data provenance in IoT systems [5, 24], to provide transparency in product manufacturing and supply chain management [42], and support vehicle provenance [39].

While the importance of blockchains is growing rapidly, it still has drawbacks and limitations that raise concerns regarding its scalability and suitability to large-scale applications. A notable concern is that the creation and maintenance of a public blockchain cause a significant waste of energy due to excessive work by the involved peers. Leading blockchains, such as Bitcoin, are based on Proof-of-Work [4, 19], where the peers, called *miners*, need to execute a demanding computation to create a block. It was estimated that the energy consumption of maintaining Bitcoin exceeds the energy consumption of Ireland [26]. The energy consumption continues to grow as more miners join the network.

Another concern is the low rate of transactions. In Bitcoin, a block is created approximately every 10 minutes, and the size of a block is fixed (1 MB in Bitcoin, 2MB in SegWit2x, and 8MB in Bitcoin Cash), and the rate of adding transactions to the blockchain is around 7 transactions per second.[1] Such a limitation exists in other blockchains as well, e.g., it is estimated that in Ethereum the transaction rate is about 10–30 transactions per second.[2] This is several orders of magnitude smaller than the transaction rate that modern financial institutions are able to process (e.g., more than 30,000 transactions per second in VISA). Changing the block-creation rate or the size of a block is difficult because a blockchain is decentralized, without any entity that can force a change or enforce new rules. In addition, rapid block-creation may result in frequent forks, which would make the blockchain less stable and more vulnerable to attacks.

Anonymity in cryptocurrencies like Bitcion provides some advantages but also creates risks. A money transfer from an owner of coins to a payee requires merely a signature using the private key of the payer. If the private key of a coin owner is revealed or stolen, the coin can be stolen. A lost private key is like lost money. Thus, cryptocurrencies are susceptible to theft and money loss.

In this paper, we envision a partitioning of blockchain into a hierarchy of sub-chains, reflecting a real-world subdivision, to increase scalability and security. We illustrate a geospatial partitioning and explain how localization and location certificates [20, 31] can be used to reliably establish association with sub-chains. The levels of the hierarchy provide a tradeoff between privacy and confirmation time of transactions. To prevent inflated energy consumption when replacing a single blockchain by many sub-chains, we introduce a novel *proof-of location* (PoL) approach that mitigates the energy consumption problem.

[1] `https://blockchain.info/charts/transactions-per-second`

[2] `https://etherchain.org/charts/tps`

2. BACKGROUND

We start by providing some background. We mainly refer to cryptocurrencies, to simplify the discussion, but the methods we suggest can be applied to other domains as well.

2.1 Blockchain

Blockchain is a decentralized ledger that stores transactions in a chain of blocks. In cryptocurrencies, a transaction can be a reward to the creator of a block, or a transfer of coins from the owner to a payee. Each transaction includes the public key of the payee. Transactions form a chain of coin transfers. To transfer money, the owner of the coins signs the transfer using the private key that matches the public key in the transaction that granted her/him the coins. Given coins and the transaction t that granted them, only someone who possesses the private key that matches the public key in the transaction t can spend the coins, i.e., transfer them on. In many blockchains, user identities are not revealed, to provide anonymity, hence, money transfer is between *wallets*, where a user may have many wallets.

We denote by $t = (x \to y, m)$ a transaction that transfers m coins from wallet x to wallet y. We denote by $t = (\to y, m)$ a transaction that grants m coins to y as a reward.

To prevent double spending, the transactions are added to the blockchain and are publicly visible. The chain defines a serialization of the transactions, so that if two transactions transfer the same coins (double spending), after the insertion of one of the transactions into the blockchain, the other transaction is considered invalid, and should not be added to the blockchain. The blockchain, thus, represents a consensus of the peers on what are valid transactions.

The transactions are organized into blocks, which are created and added to the blockchain by members of a peer-to-peer network. In Bitcoin, these peers are called *miners*. The first block in the chain is the *genesis block*. Every other block contains a hash of the previous block in the chain, e.g., using SHA-256. This means that a change in one of the blocks would either result in an incorrect chain or would require changing the hash values in all the following blocks.

A blockchain is maintained in a decentralized manner. It is immutable, where changes of past blocks are practically impossible. To achieve that and to prevent forks, where a separation of the chain cannot be resolved, blockchains like Bitcoin rely on *proof-of-work* (PoW)—a computation that is hard and time consuming, e.g., a cryptographic riddle. In Bitcoin, each block includes a *nonce* such that the hash of the block (with the nonce) has at least k leading zeros. Computing the nonce is hard, hence it is a PoW. The value k is determined such that the overall computation by all the peers (miners) would require approximately 10 minutes for computing a block. In a case of a conflict, or a fork, miners are expected to add blocks to the longest branch. This causes short branches to be abandoned and prevents forks. A block that contains invalid transactions, e.g., double spending, will be ignored by the majority of the peers, and eventually will not be part of the chain.

An attacker that tries to change a block in the blockchain needs to create an alternative branch and compete with all the other miners, in an attempt to make the alternative branch the longest one. The chances of succeeding are slim, due to the hardness of block creation. This provides immutability, stability and reliability. A comprehensive survey of blockchain technologies is provided in [1].

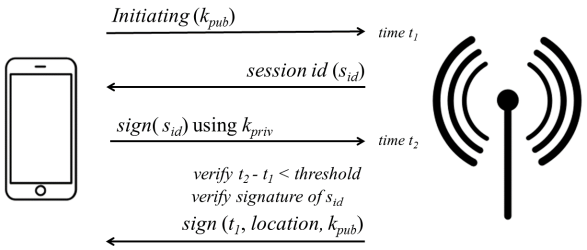

Figure 1: Issuing a location certificate for a requester (left) by a corroborator (right).

2.2 Location Certificate

Geospatial partition is natural in many blockchain applications. It is based on reliably mapping transactions to their location and time, and providing a certificate of that. The *location certificate* is a digital proof that a device was at a particular place at a specific time. GPS cannot be used for that because GPS can be spoofed [38].

One way to produce location certificates is based on the existence of a trusted *localized corroborators* that could provide the certificate [20]. A localized corroborator is a server that has a known location, and that can only be accessed from a short range. It can be a server that is directly, physically, connected to stationary devices. For mobile devices it can be a cellular tower, a wireless access point (Wi-Fi, Bluetooth, ZigBee), an optical access point (based on infrared light), etc. A device can only be connected to the localized corroborator if it is near the corroborator—a few meters in a case of Bluetooth, ZigBee or infrared sensor; dozens of meters for Wi-Fi; and a few kilometers for a cellular tower. Higher accuracy can be achieved by taking the signal strength into account [30, 44, 45]. The trustworthiness of certificates can be strengthen by adding cryptographically signed geotags to IP packets [11]. We assume that the corroborator has a unique pair of a private key and a public key, as part of a public-key cryptosystem.

For our purposes, we consider issuing a location certificate for a device that holds a specific private key—the private key remains concealed and only the public key is revealed to the corroborator or to a verifier. For a given pair (k_{priv}, k_{pub}) of private and public keys, the certificate attests that a device containing the private key k_{priv} was near the corroborator at the time of the issuing.

The protocol involves the following steps.

1. The requester sends an initiation message to the server, including the public key k_{pub}.

2. The corroborator sends a random session id s_{id} to the requester.

3. The requester sends back the session id s_{id} signed using the private key k_{priv}.

4. The corroborator checks the time that elapses between sending s_{id} and getting it back (signed) and verifies the authenticity of the signature using k_{pub}. When the time difference is a few milliseconds (less than a threshold of say 5 milliseconds), the corroborator issues a certificate consisting of the time, location and requester public key k_{pub}, signed by the private key of the corroborator.

The requester cannot create a certificate without the corroborator because a valid certificate requires the signature of the corroborator. The session id can only be signed after the beginning of the session, because it is unknown before the session starts. Therefore, after the session initiation, a device that can sign the session id with k_{priv} must be near the corroborator, to provide a response in a latency that is smaller than the threshold. The certificate can include a precise location or a general one, e.g., a city, a county, a state, to increase privacy.

A *certified transaction* is a pair (t, C) of a transaction $t = (x \to y, m)$ and a location certificate C, where the public key of y is used to create the certificate. As explained, the certificate is created by a device that at the certified time is near the corroborator and contains the private key of y.

3. BLOCKCHAIN PARTITIONING

We present now our partitioning approach. In public blockchains like Bitcoin and Ethereum, the transaction rates are low. One of the reasons for the low transaction rate is the serialization of all the transactions, even those that are not conflicting. Had there been a partition of the transactions into groups so that transactions from different groups could never conflict, non-conflicting transactions could have been processed in parallel, and blocks of non-conflicting transactions could have been generated in parallel. This can be achieved by creating a partition of the blockchain into a hierarchy of blockchains (sub-chains) and associating transactions with different nodes of the hierarchy. Each sub-chain is managed independently, so blocks of different sub-chains can be created and added to the appropriate chain in parallel.

The study of parallel creation of blocks led to the development of the BlockDAG data structure, where a new block can extend several previous blocks, not just one, and the "heaviest" tree is selected in a greedy fashion, e.g., using the GHOST protocol [34]. The SPECTRE protocol [33] utilizes BlockDAG for a virtual vote on the order of the blocks, to achieve high throughput and fast confirmation time. Two other notable attempts to cope with the low transaction rates in public blockchains are Bitcoin-NG [14] and Algorand [17]. Bitcoin-NG speeds up block creation by electing a leader for a specified epoch, and allowing the leader to create a large number of blocks till the next leader is elected. Algorand employs a sophisticated method of randomly selecting a small group of users (who are replaced when their identity is revealed) and executing a Byzantine Agreement protocol by the chosen users, to prevent forks altogether. Our approach is orthogonal to these systems. First, in a hierarchy of linked sub-chains, any blockchain implementation can be used, including Bitcoin, Bitcoin-NG, Algorand, and others. The hierarchical structure may even link different types of blockchain. Second, scalability is achieved by adding new sub-blockchains to the hierarchy without changing the technology or performing a hard fork.

Different hierarchies can be used. Geospatial hierarchy is a natural one, e.g., a partition into neighborhoods, cities, counties, states and countries. Such a partition is suitable, for example, when using blockchains to record real-estate transactions. Another partition example is a partition into business units of a large global company, e.g., teams, depart-

ments, divisions, sub-organizations, etc. Such a partition can be applied when a company ledger is used for recording processes, data sharing, code transfer, etc.

We elaborate on geospatial partition. Our underlying assumption is that most transactions are local, e.g., cash exchange is often between people who are geographically near, and this may also be true in a cryptocurrency that aims to replace cash. Other usages of geospatial partition are real estate transactions, supply chains, management of data in smart cities, and so on. The hierarchy provides a tradeoff between privacy and efficiency, where local transactions are more efficient and non-local ones are more private.

A localized blockchain is defined with respect to a given area A, e.g., the area of the USA. Localization is with respect to a *hierarchical partition* of A, and each wallet is associated with a sub-area in A.

EXAMPLE 1. *In a hierarchical partition of the USA, the country is partitioned into states, states are partitioned into counties, and counties are partition into cities and towns. A transaction within a city is registered merely in the city. A transfer of coins from a city in one county to a city in another, within the same state, is registered in the relevant cities, counties and the state. A transfer across states is recorded in all the levels of the hierarchy.*

The partitioning of the blockchain makes local transactions faster and cheaper than non-local ones, because a local transaction is notarized for a local area and "competes" with less transactions. When moving higher in the hierarchy, each transaction may need to compete with transactions from a wider area—this will increase privacy, but also expected to increase the transaction delay (i.e., lengthen the wait time till the transaction is recorded in the blockchain).

The hierarchy is the result of a recursive partitioning of A. Formally, let $\mathcal{A}$ be a set of subareas of A. The hierarchical partition $H = (T, \alpha)$ of A comprises a tree $T = (V, E, v_{root})$ and a function $\alpha : V \to \mathcal{A}$, where V, E, and v_{root} are the vertexes, edges and root of T, respectively. The function α maps each vertex v to a subarea in $\mathcal{A}$, such that for each node v that is not a leaf it must hold that: *(1)* $\alpha(v) = \cup_{u \in children(v)} \alpha(u)$, and *(2)* $\alpha(u_1) \cap \alpha(u_2) = \emptyset \quad \forall u_1 \neq u_2 \in children(v)$. That is, the areas associated with the children of a vertex v are a partition of the area associated with v.

A wallet is localized by associating it to a node of H. Let W be the set of all wallets, then $\lambda : W \to V$ is a function that maps wallets to nodes of H. A wallet $w \in W$ is associated with the area $\alpha(\lambda(w))$. A transaction $t = (x \to y, m)$ is *local* if $\lambda(x) = \lambda(y)$ is a leaf of H. Otherwise, the *LCA* of t is the least-common ancestor $lca(x, y)$ in T. The area of t is $\alpha(lca(x, y))$.

Certification. A *certification requirement* allows only processing of certified transactions (t, C). When including a certified transaction in a blockchain, it is required to verify that the certificate C is valid and includes the public key of the receiving wallet y. The location in C should be inside the area of the receiving wallet, i.e., in $\alpha(\lambda(y))$.

We consider three types of transfers. A *lateral transfer* between wallets in the same node. An *ascending transfer* from a wallet in a node v to a wallet in the parent of v. A *descending transfer* from a wallet in a node v to a wallet in a child of v. The blocks of each node of H are managed separately from the blocks of the other nodes, with a distinct chain for each node. To increase the efficiency, blocks associated with different nodes can be created in parallel.

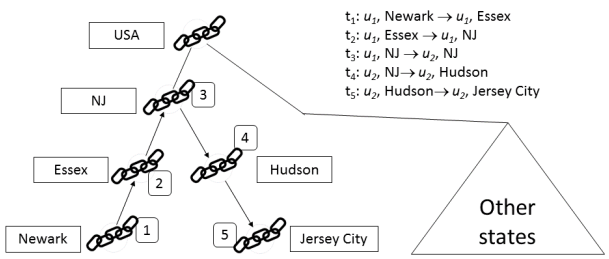

Figure 2: Hierarchical partitioning and a transfer.

To prevent double spending, each transaction $t = (x \to y, m)$ must be added to the blockchain of the node associated with x, to get accepted. The transaction $t' = (y \to z, n)$ that follows t is added to the blockchain of the node associated with y. A local transaction that is related to node v is added to $blockchain(v)$, as a lateral transfer. A non-local transaction from x to y is translated to a sequence of transfers along the shortest path from x to y in T.

For example, a transfer of coins within Chicago is local and requires a single lateral transfer. A transfer from a wallet of user u_1 in Newark, NJ to a wallet of u_2 in Jersey City, NJ requires the five transfers depicted in Fig. 2.

The geographic partition can be done in different ways depending on how people use money. Several transfers are needed for non-local transactions, but blocks of different chains are created in parallel. For anonymity, users can choose the level at which they execute transactions—a higher level provides a more obfuscated exposure of the user location. There is, however, a tradeoff between privacy and the time that elapses till a transaction is added to the blockchain.

Non-geographic partitions could be applied as well. In a large corporation, for instance, a partition based on the divisions and subdivisions of the company could be used to manage company transactions, as in the geospatial partition.

Geofencing. Partitioning of blockchains can be used to strengthen security. We describe geofencing as an example.

A private key of a wallet can be stolen, which may lead to the loss of the coins. By *geofencing wallets*, coins can be more secure. In geofencing, a wallet is associated with an area, as explained in Section 3. For executing a transaction, the payee needs to provide a location certificate for a place within the area of the payer's wallet, at the time of the transaction. If, for example, Alice associates her wallet with her neighborhood, a malicious attacker from a different country, say Mallory, would be limited in her ability to spend the money. Even if Mallory would steal the private key of Alice, to create a certificate and transfer the coins she would need to have a device in Alice's neighborhood with the private key of the receiving wallet. If Mallory would use as a proxy a device in Alice's neighborhood, to create a certificate on her behalf, she would need to surrender her private key to the proxy. Hence, the taken money could be spent by the proxy. This would make cryptocurrencies more secure. The stronger security would also make it safer to create backups for a lost key. Note that Alice could transfer money from her local wallet to a wallet associated with her state, if she wants to use the money when traveling within the state.

Geofencing can be done by requiring a certificate from the payee, the payer or from both, to restrict, at the time of the transaction, the location of the payer, the payee or of both. Note that geofencing strengthens the security provided by the private keys, it does not replace private keys. There is a tradeoff between security and privacy here—smaller area provides more security but less privacy, and vice versa.

Geofencing can be applied to various applications of blockchain, e.g., in a blockchain that supports a supply chain, transactions of item transfer could be limited to the warehouses, i.e., they could only be recorded at the warehouses, to provide strict control over transfers and their registration.

4. PROOF-OF-LOCATION

Blockchains that are based on Proof of Work (PoW) are wasteful, that is, consume an excessive amount of energy. A partition of the blockchain could increase the amount of energy that is required to sustain the system. In this section we show how location certificates can be used to establish Proof-of-Location as a non-wasteful alternative to PoW, to achieve consensus in a public blockchain.

4.1 Proof of Work

Over the years, PoW has been proven to be a successful and reliable consensus mechanism for a public (permissionless) blockchain like Bitcoin, and capable of preventing a Sybil Attack [13]. Its main limitation, however, is the immense energy consumption that is required to maintain the system. Miners who create a block are rewarded for that by receiving transaction fees or a block-creation incentive. They compete to create blocks, and thus, many miners spend significant computation power on finding a suitable nonce, for each block. Furthermore, if miners would collude, they could issue a 51% attack or in some cases, even a 25% attack [15]. This is a real threat because Bitcoin miners are already organized into large groups and share their computational resources to create blocks [16].

4.2 Alternatives to PoW

Several methods were proposed as an alternative to PoW. One of them is *proof-of-stake* (PoS) [6, 7, 22], where the voting power is given to "stake holders" of the system, i.e., to those who have coins. The creator of a block needs to provide a cryptographic proof of existence of a certain amount of coins in its possession, and these coins are locked till some conditions are met. This approach was criticized as non-resilient to forks, since, unlike in PoW, the expected gain from working on more than one branch is often higher than the cost of doing so. Furthermore, in this method peers with many coins could delay the creation of new blocks (when they are selected to create the next block) and could use that for extortion, or in an attempt to attack the system for an external gain [23].

In *proof-of-disk-space* the creators of blocks need to waste disk space to create a block [27, 29]. Like PoW, it is a wasteful approach. A consensus protocol to cope with the case where an unknown number of peers could be offline was suggested in [28].

Several solutions were designed for private (permissioned) blockchains, see an analysis in [12]. Practical Byzantine Fault Tolerance (PBFT) [9, 41] was proposed as a method to reach consensus by voting, but it requires knowing the number of peers, so it is unsuitable for a public blockchain in which joining the peer-to-peer network is open to the public. *Proof of authority*[3] was developed for private blockchains, with trusted entities as authorities. It relies on establishing trust in the peer-to-peer network, e.g., see [43].

4.3 Implementing PoL

We introduce now *proof-of-location* (PoL)—a novel alternative to PoW. It aims to avoid waste when creating a block, and yet keep the process decentralized and independent of knowledge about the reputation of peers, or their number. It is based on the ability to create a location certificate to provide a location proof [20, 31, 32], for a particular place, to create the next block.

Block creation. The blockchain is created such that a location ℓ is selected in each step, in an unpredictable way, and the next block is the one that was created by the peer with the PoL closest to the selected location. If two location certificates have the same distance from the selected location, the one with the smallest time stamp is selected.

The selection of a location ℓ can be done in different ways. One way is as follows. Consider the geographical area in which the block creators (peers) are active, e.g., USA. Let G be a grid that covers this area. Let $c_1, \ldots, c_m$ be the cells of G. Let B be the last block in the blockchain, so far, and $h(B)$ the hash of B. The selected location is the center of the cell number $h(B) \bmod m$, i.e., $c_{h(B) \bmod m}$ of G. This yields a cell whose coordinates cannot be computed without knowing B. Note that for a hash function h whose digest has a size of 256 bits, even if the remainder of the division $2^{256}/m$ is non-zero, the difference between $\left\lfloor \frac{2^{256}}{m} \right\rfloor$ and $\left\lfloor \frac{2^{256}}{m} \right\rfloor + 1$ is negligible, so if h is uniform then the selection of cells can, practically, be regarded as uniform.

To control the hardness of block creation, so that an attacker could not create an alternative branch fast, we suggest that the distance of the certificate from ℓ would be limited by an adaptable inflating bound. One option to do so, is as follows. Let t_{prev} be the creation time of the last block. The *inflating distance limit* is $d(t) = \delta \cdot minutes(t - t_{prev})^k$, for given k and δ. A location certificate with location and time (l_p, t_p) satisfies the distance limit if $distance(l_p, \ell) < d(t_p)$. For $k = 3$ and $\delta = 100$ meters, in the first minute (time difference < 1), the certificate should be for a location that is less than 100 meters from ℓ. In the second minute (time difference < 2), the certificate should be for a location that is less than 800 meters from ℓ. The distance limit (in meters) as a function of the time difference (in minutes) evolves as follows: $(2, 800), \ldots, (4, 6400), \ldots, (8, 512, 000), \ldots, (10, 100, 000), \ldots$. With these parameters, the distance limit is 100 kilometers after 10 minutes, and covers the area of the USA after about half an hour. (These parameters can be changed to control the block creation rate, and guarantee that blocks will be created within a reasonable time.)

An attacker that would try to change a block and then create the longest branch, by competing with the other miners, would need to produce location certificates faster than the other miners. However, without a machine and a corroborator near any arbitrary location ℓ, the attacker would need to wait, e.g., if its nearest machine to ℓ is 100 kilometers, it would need to wait 10 minutes, and at that time the other

[3] https://github.com/paritytech/parity/wiki/Proof-of-Authority-Chains

miners would add blocks to the main chain. Note that with machines that cover an area of 10 km^2, about 1,000,000 machines would be needed to cover the area of the USA.

An advantage of the proposed method is that, unlike in Bitcoin, if the locations of the peers (miners) are arbitrary, a group of miners that collude do not have an advantage over a group that do not collude. This would make the system less vulnerable to colluding peers. Furthermore, for an attacker it will be hard to create blocks fast, even with a large computation power, because the computation power would not help arriving at ℓ or getting close to ℓ faster.

Fork Prevention. When two or more branches are constructed in parallel without being abandoned, forks occur. Forks cause the blockchain to be less reliable, and reduce consistency. To cope with that, the rule of thumb is that the miners would continue the longest branch so far. But there is also a need to discourage the miners from extending other branches. In PoW, the computation of a nonce is demanding, so miners have an incentive to invest their computation power only on the branch with the highest chance of success (the longest one). This can be achieved in PoL if there would be a cost to each certificate, e.g., where miners would pay to the corroborators for each creation of a location certificate. (Note that in PoW miners pay for block creation in their electricity bills.) A payment would encourage miners to only "invest" in a branch with a high chance of success. The payment can be adaptive, e.g., including ℓ in the certificate and making the fee proportional to the distance between the corroborator and ℓ, to discourage miners that are geographically far from ℓ from creating a block.

Effect on Miners. In PoL, the miners create location certificates and reveal their location. This, however, does not affect users, i.e., there is no disclosure of the locations of the payers or the payees whose transactions are added to a block. It is an open question, however, whether revealing the location of miners is much different from revealing their IP addresses, as being done anyway in the peer-to-peer network. (Miners can hide their IP address, e.g., by using onion routing [18], but this would slow them down in the "race" to create a block. Such a tradeoff between privacy and effectiveness can be made also in PoL, where a miner may decide only to create location certificates by a mobile device when she/he is far from her/his home or office.)

Decentralized System. In PoL, the system remains decentralized, because location certificates are not produced by a single entity. The certificate may be produced by different companies and organizations using network access points, e.g., modifying all the cell towers to serve as corroborators. A company that would not provide reliable certificates, the blocks with its certificates would not be accepted by the majority of the miners, and hence, users will stop acquiring certificates from it. Hence, the incentive of certificate providers to be honest is similar to that of miners in a public blockchain like Bitcoin.

Sybil Attack. To create a certificate there is a need to be near the corroborator. Therefore, forging many identities that are located in a single place does not increase the ability to create a block if PoL is used. Also, having more machines or stronger machines in proximity to a single corroborator does not give an advantage. A miner could try to deploy many machines in many remote places. This, however, would require investment in equipment and would incur maintenance costs, and unlike Bitcoin mining farms could not be in a single location.

An attacker may try to apply *cryptojacking*, i.e., use machines of other users to create location certificates, somewhat like unauthorized use of machines for Bitcoin mining. But in such a case, to create the certificate, the attacker would need to expose the private key of the wallet that would receive the incentive fee (this key is necessary to create the certificate). Any hijacked machine would then have the private key that would allow it to spend the new coins.

To increase security, there should be many corroborators distributed over a large area. More importantly, each corroborator should have a different private key—if the security of a corroborator will be breached, using its key for creating fake certificates would be limited to a single location.

5. CONCLUSION AND DISCUSSION

Blockchain has the potential to revolutionize data sharing among organizations and individuals, by providing a decentralized, transparent and tamper-proof storage of transactions. It is the underlying technology of many cryptocurrencies, and is adapted for other uses. However, currently blockchains are not scalable (they have a low transaction rate), and public blockchains are wasteful (require a high usage of electricity to support PoW), and insecure (provide no protection from theft of a private key). In this paper, we present a novel approach of partitioning the blockchain into a tree of sub-chains based on a real-world hierarchy, like a geographical or an organizational partition, where transactions of different sub-chains do no conflict with one another. Such a partition provides a tradeoff between efficiency and privacy—high levels provide more privacy than low levels but a longer expected wait till the transaction is added to a block, and vice versa. Scalability can be achieved by partitioning leaf nodes in which the transaction rate is too high. Creating an optimal hierarchy and adapting the hierarchy to changes are challenging research directions.

An important advantage of the hierarchical partitioning is that there is no need to develop a new technology or perform hard forks to cope with scalability issues. The recent debate about how to increase the block size of Bitcoin illustrates how difficult it is to make changes in public blockchains.

We explain how a geographic partitioning combined with location certificates can be used to increase security by applying geofencing. With the growing popularity of cryptocurrencies and their usage in applications that do not require privacy, strengthening security by restricting usage of coins to specified locations could proliferate utilization of cryptocurrencies. How to further increase security of cryptocurrencies at the expense of privacy, but without completely revealing user identities, is an open question.

The partition of the blockchain may inflate the excessive energy consumption cause by PoW. Thus, we suggest a novel non-wasteful proof-of-location (PoL) method, to achieve consensus for block creation. In PoL, unlike PoW or PoS, having a strong computation power or many coins does not increase the chances of creating the next block. This has the potential of providing higher stability than that of PoW or PoS, however, further research is required to prove that.

Note that our vision of using partitions to create subchains can be generalized from hierarchies to a network of blockchains, e.g., by connecting existing blockchains. We defer a detailed discussion in the interest of space.

6. REFERENCES

[1] D. T. T. Anh, M. Zhang, B. C. Ooi, and G. Chen. Untangling blockchain: A data processing view of blockchain systems. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[2] D. Augot, H. Chabanne, T. Chenevier, W. George, and L. Lambert. A user-centric system for verified identities on the bitcoin blockchain. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, pages 390–407. Springer, 2017.

[3] D. Augot, H. Chabanne, O. Clémot, and W. George. Transforming face-to-face identity proofing into anonymous digital identity using the bitcoin blockchain. *arXiv preprint arXiv:1710.02951*, 2017.

[4] A. Back. Hashcash–a denial of service counter-measure, 2002.

[5] N. Baracaldo, L. A. D. Bathen, R. O. Ozugha, R. Engel, S. Tata, and H. Ludwig. Securing data provenance in internet of things (IoT) systems. In *International Conf. on Service-Oriented Computing*, pages 92–98, 2016.

[6] I. Bentov, A. Gabizon, and A. Mizrahi. Cryptocurrencies without proof of work. In *International Conference on Financial Cryptography and Data Security*, pages 142–157. Springer, 2016.

[7] I. Bentov, C. Lee, A. Mizrahi, and M. Rosenfeld. Proof of activity: Extending bitcoin's proof of work via proof of stake. *ACM SIGMETRICS Performance Evaluation Review*, 42(3):34–37, 2014.

[8] K. Biswas and V. Muthukkumarasamy. Securing smart cities using blockchain technology. In *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*, pages 1392–1393. IEEE, 2016.

[9] M. Castro and B. Liskov. Practical byzantine fault tolerance and proactive recovery. *ACM Trans. Comput. Syst.*, 20(4):398–461, 2002.

[10] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman. Blockchain technology: Beyond bitcoin. *Applied Innovation*, 2:6–10, 2016.

[11] T. Dasu, Y. Kanza, and D. Srivastava. Geotagging IP packets for location-aware software-defined networking in the presence of virtual network functions. In *Proc. of the 25th ACM SIGSPATIAL International Conf. on Advances in Geographic Information Systems*. ACM, 2017.

[12] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan. BLOCKBENCH: a framework for analyzing private blockchains. In *Proc. of the ACM International Conf. on Management of Data*, pages 1085–1100, 2017.

[13] J. R. Douceur. The sybil attack. In *International Workshop on Peer-to-Peer Systems*, pages 251–260. Springer, 2002.

[14] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse. Bitcoin-NG: A scalable blockchain protocol. In *NSDI*, pages 45–59, 2016.

[15] I. Eyal and E. G. Sirer. Majority is not enough: Bitcoin mining is vulnerable. In *International Conf. on Financial Cryptography and Data Security*, pages 436–454, 2014.

[16] A. Gervais, G. Karame, S. Capkun, and V. Capkun. Is bitcoin a decentralized currency? *IEEE security & privacy*, 12(3):54–60, 2014.

[17] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 51–68. ACM, 2017.

[18] D. Goldschlag, M. Reed, and P. Syverson. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999.

[19] M. Jakobsson and A. Juels. Proofs of work and bread pudding protocols. In *Secure Information Networks*, pages 258–272. Springer, 1999.

[20] Y. Kanza. Location corroborations by mobile devices without traces. In *Proc. of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2016.

[21] Y. Kanza and H. Samet. An online marketplace for geosocial data. In *Proc. of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2015.

[22] A. Kiayias, A. Russell, B. David, and R. Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual International Cryptology Conference*, pages 357–388. Springer, 2017.

[23] J. A. Kroll, I. C. Davey, and E. W. Felten. The economics of bitcoin mining, or bitcoin in the presence of adversaries. In *Proceedings of WEIS*, 2013.

[24] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu. Blockchain based data integrity service framework for IoT data. In *Web Services (ICWS), 2017 IEEE International Conf. on*, pages 468–475, 2017.

[25] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[26] K. J. O'Dwyer and D. Malone. Bitcoin mining and its energy footprint. In *25th IET Irish Signals & Systems Conference*, Limerick, Ireland, 2014. IET.

[27] S. Park, K. Pietrzak, J. Alwen, G. Fuchsbauer, and P. Gazi. Spacecoin: A cryptocurrency based on proofs of space. Technical report, IACR Cryptology ePrint Archive, 2015.

[28] R. Pass and E. Shi. The sleepy model of consensus. In *International Conf. on the Theory and Application of Cryptology and Information Security*, pages 380–409, 2017.

[29] L. Ren and S. Devadas. Proof of space from stacked expanders. In *Theory of Cryptography Conference*, pages 262–285. Springer, 2016.

[30] S. Saha, K. Chaudhuri, D. Sanghi, and P. Bhagwat. Location determination of a mobile device using IEEE 802.11 b access point signals. In *Wireless Communications and Networking*, volume 3, pages 1987–1992. IEEE, 2003.

[31] S. Saroiu and A. Wolman. Enabling new mobile applications with location proofs. In *Proc. of the 10th Workshop on Mobile Computing Systems and Applications*. ACM, 2009.

[32] N. Sastry, U. Shankar, and D. Wagner. Secure

verification of location claims. In *Proceedings of the 2nd ACM workshop on Wireless security*, pages 1–10. ACM, 2003.

[33] Y. Sompolinsky, Y. Lewenberg, and A. Zohar. Spectre: A fast and scalable cryptocurrency protocol. *IACR Cryptology ePrint Archive*, 2016:1159, 2016.

[34] Y. Sompolinsky and A. Zohar. Secure high-rate transaction processing in bitcoin. In *International Conference on Financial Cryptography and Data Security*, pages 507–527. Springer, 2015.

[35] J. Sun, J. Yan, and K. Z. Zhang. Blockchain-based sharing services: What blockchain technology can contribute to smart cities. *Financial Innovation*, 2(1):26, 2016.

[36] M. Swan. *Blockchain: Blueprint for a new economy.* " O'Reilly Media, Inc.", Sebastopol, CA, USA, 2015.

[37] D. Tapscott and A. Tapscott. *Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World*. Penguin Random House, New York, NY, USA, 2016.

[38] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun. On the requirements for successful gps spoofing attacks. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 75–86. ACM, 2011.

[39] S. Underwood. Blockchain beyond bitcoin. *Commun. ACM*, 59(11):15–17, Oct. 2016.

[40] P. Vigna and M. J. Casey. *The age of cryptocurrency: How bitcoin and digital money are challenging the global economic order*. St. Martin's Press, New York, NY, 2015.

[41] M. Vukolić. The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *International Workshop on Open Problems in Network Security*, pages 112–125. Springer, 2015.

[42] N. Vyas. Disruptive technologies enabling supply chain evolution. *Supply Chain Management Review*, 2016.

[43] L. Xiong and L. Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.

[44] K. Yedavalli and B. Krishnamachari. Sequence-based localization in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 7(1), 2008.

[45] K. K. Yedavalli. *Location Determination using IEEE 802.11 b*. PhD thesis, University of Colorado, 2002.