bloXroute: A Scalable Trustless Blockchain Distribution Network WHITEPAPER

Uri Klarman[1,3], Soumya Basu[2,3], Aleksandar Kuzmanovic[1,3], and Emin Gün Sirer[2,3]
[1]Department of Electrical Engineering and Computer Science, Northwestern University
[2]Department of Computer Science, Cornell University
[3]bloXroute Labs Inc.

EXECUTIVE SUMMARY — Bitcoin and other cryptocurrencies provide an attractive and exciting alternative form of currency, as they provide the convenience of credit cards, the usability of cash, the security of a bank vault, and retain their values like gold, without the associated high fees and cumbersomeness. They further enable futuristic services such as paying for groceries as they are being collected or picking up a rented umbrella for a minute's walk without ever standing in line. Cryptocurrencies can enable these services without pre-payment or pre-registration and more crucially, without placing trust in the providers of these services. Such features are currently unfeasible due to payment processing fees which makes such micro-transactions uneconomical.

To be truly useful for ordinary men and women all around the world, cryptocurrencies must scale the number of transactions they can process by a factor of x$1,000$, from 3–5 transactions per second (Bitcoin and Ethereum) to thousands of transactions per second. In fact, hundreds of transactions per second will be required only to enable U.S. cars to pay for gas on a bi-monthly basis, or alternatively, to process the cups of coffee purchased at Starbucks. To allow machine-to-machine micro-transactions, and realize their potential in earnest, cryptocurrencies must scale much further.

The limiting factor of Bitcoin and other cryptocurrencies is their network. Specifically, they employ a trustless P2P network model to propagate transactions and blocks, which does not scale as the volume of transactions increases, a fact research has shown time and again. Indeed, if blocks and transactions were to be instantly propagated, immense blocks could have been mined at a rapid pace, until the limitation of designated processing units and flash storage arrays was reached.

To overcome this limitation, and to allow *all* cryptocurrencies to scale to thousands of on-chain transactions per second *today*, we propose *bloXroute*, a provably neutral transport layer which runs underneath cryptocurrencies. *bloXroute* allows to safely increase the block size and to cut down the time interval between blocks, without increasing the risk of forks, and provides real-time support for immediate transactions with zero-confirmation (0-conf). The use of *bloXroute* requires no consensus, nor a protocol change, beyond adjusting system parameters. It is compatible with any off-chain scaling solutions, complementary to the native consensus protocol used, and can be gradually deployed by any node wishing to receive blocks at a higher rate. With the networking bottleneck removed, each cryptocurrency community is free to adjust its protocol to best leverage this newfound capacity, in order to increase its real-world impact and value.

Scaling cryptocurrencies by a factor of x$1,000$ and more benefits the entire ecosystem: as an example, reducing user fees by a factor of x100, increasing the total fees collected by miners by a factor of x10, and awarding *bloXroute* a payment for the transactions it enables to maintain its sustainability. Note that the payments to *bloXroute* are utterly voluntary, yet they incentivize miners to require a significantly smaller fee. *bloXroute* is thus designed as a Win-Win-Win scenario, benefiting users, miners, and the *bloXroute* system alike, with 99.9% of the value created being captured by the users and the miners.

To support the development of this network, *bloXroute* launches its own capped-supply ERC20 token – *BLXR*, which supports *bloXroute*'s goal: promoting the success of all cryptocurrencies, rather than competing with them. To do so, *all* of the funds received by *bloXroute* are immediately directed to a newly created pooled account, the *BLXR-reserve*, which is owned by all BLXR holders. Holders receive their pro-rata share of the fees collected, denominated in the tokens and blockchains that use bloXroute. A BLXR holder can pull its proportional share of collected fees, which consists of heterogeneous cryptocurrencies. In this fashion, BLXR aligns the incentives of the entire ecosystem: bloXroute, cryptocurrencies, users, miners, and investors.

Index Terms—**Blockchain, net neutrality, peer auditing, bloXroute, BDN, broadcast, scalability, BLXR.**

I. ABSTRACT

BLOCKCHAINS are decentralized systems that forgo trusted third parties in favor of a distributed trust model through a peer-to-peer network. While such a design brings significant opportunities and disruptive potential in many industries, scalability has been a key issue preventing wider adoption. Indeed, the most prominent blockchain, Bitcoin, has a throughput 3-4 orders of magnitude smaller than Visa. The key hypothesis of our work is that it is possible to enable scalable blockchains through a global network infrastructure without placing trust in the infrastructure itself. We present *bloXroute*, the first Blockchain Distribution Network (BDN), which increases a blockchain's *on-chain* throughput by more than three orders of magnitude via an effective *broadcast* primitive, without affecting a blockchain's functionality and the balance of power among current system participants. Further, due to the fast underlying network, this throughput increase can be easily realized by tweaking the block size and block time interval. This is achieved via a *provably neutral* network design as the first-order priority for *bloXroute*. Our system is the first to combine a legacy peer-to-peer network and a novel global BDN where the peer-to-peer network is used to audit the BDN and its neutrality. *bloXroute* is protocol-, coin-, and blockchain-agnostic, capable of simultaneously supporting any number of blockchains. Additionally, we introduce

Manuscript issued on January 15, 2019
Corresponding author: U. Klarman (email: uri@bloxroute.com).

BLXR, an investment vehicle which allows both investing in *bloXroute* and directing *bloXroute*'s revenues back to the crypto ecosystem.

II. Introduction

The blockchain and cryptocurrency revolution, initiated by Bitcoin in 2008 [1], is thriving on the Internet; the market capitalization of Bitcoin, Ethereum, and other prominent cryptocurrencies has crossed 500 billion USD. The key feature of blockchains is the lack of a central trusted authority, instead relying on a global peer-to-peer (P2P) network[1] to validate and certify all transactions. Given the distributed and decentralized nature of blockchains, it is believed that such systems have a disrupting potential in many other areas beyond finance, including healthcare, retail, government, insurance, *etc*.

The major problem for blockchains is *scalability*, which is fundamentally hindered by the distributed system design and limitations of the underlying P2P network model, as we elaborate in depth in Section III. In particular, the blockchain system throughput is measured in terms of the number of transactions per second (TPS) a system can support. Currently, Bitcoin had reached its capacity with an average throughput of 2.94 TPS. For comparison, Visa's centralized system processes an average of 2,000 TPS, its daily peak is 4,000 TPS, and it has the capacity to process up to 56,000 TPS. Without scalability, cryptocurrencies and blockchains are simply incapable of providing the functionality they promise, not only in finance but also in other areas such as commerce, healthcare, and IoT.

In this paper, we add a protocol-agnostic networking solution that solves the blockchain scalability problem without changing the existing blockchain model and leaving the current system design intact. We embrace a *blockchain distribution network* (BDN) to enable blockchain scaling *without* compromising the decentralization of control over transactions in a blockchain. The key challenges are to design such a BDN to be *neutral* (we explain the notion of network neutrality in this specific context below) and *auditable* by the global Peer Network, while retaining the existing blockchain's functionality, properties, and the balance of power among current system participants.

We propose *bloXroute*, a scalable, neutral, and auditable BDN. To achieve scalability, *bloXroute* utilizes (i) system-wide caching that enables faster propagation and Gigabyte-size blocks, and (ii) cut-through routing that enables swift and efficient transmission of blocks through the network. In essence, *bloXroute* implements and provides an efficient *broadcast* primitive to the blockchain nodes, via a network of Gateways, making them operate as if they are on the same Local Area Network, while in reality they might be residing at opposite parts of the globe. As a result, *bloXroute* increases the throughput of the associated *bloXroute*-supported blockchains by more than 3 orders of magnitude relative to the state-of-the-art P2P blockchain systems, closing the gap between P2P blockchains and traditional payment systems such as Visa.

With thousands of transactions per second, *bloXroute* can enable blockchains to support and automate very mundane tasks. For example, if a blockchain records one transaction every time a car fills its gas tank, just supporting the US would require 400-500 transactions per second. If every vending machine supported just four purchases a day through a blockchain, that blockchain would need to support 1000 transactions per second. If every vote in the 2016 US presidential election was recorded on a blockchain and was cast over 24 hours, that blockchain would need to support at least 1500 transactions per second. These applications were thought to be years away, but *bloXroute* allows current blockchains to scale to this level by simply adjusting its parameters.

We define *bloXroute*'s neutrality as follows: *bloXroute* propagates blocks in the exact same manner for every user of the system. In particular, *bloXroute* propagates blocks without knowledge of the transactions they contain, their number, and the "wallets" or addresses involved. Miners are free to include arbitrary transactions in a block. Furthermore, *bloXroute* cannot infer the above characteristics even when colluding with other nodes, or by analyzing blocks' timing and size. *bloXroute* cannot favor specific nodes by providing them blocks ahead of others, and cannot prevent any node from joining the system and utilizing it. In short, *bloXroute* can only propagate all blocks to all its Gateways fairly.

To achieve neutrality and enable its auditing, *bloXroute* supports encrypted blocks, which prevent it from stopping the block propagation based on its content or any other feature. A block's encryption key is only revealed after the block has been propagated through the network. To ensure *bloXroute* is not discriminating against individual nodes, Gateways do not propagate blocks directly to *bloXroute*, but relay them via peers in the P2P network to obscure a block's origin from *bloXroute*. To prevent *bloXroute* from blocking or stalling blocks arriving from a particular set of nodes, nodes can actively audit *bloXroute*'s service and performance by sending *test-blocks* to *bloXroute*. Lastly, *bloXroute* incorporates peer-controlled measures to sustain blockchain operations even in the event of a complete system failure. The *bloXroute* system as a whole is protocol-agnostic, capable of providing its scaling services to numerous cryptocurrencies and blockchains simultaneously.

Our main contributions are the following:

- We present *bloXroute*, the first BDN that utilizes a global network infrastructure to scale blockchains *without* affecting the decentralized control over transactions in a blockchain.
- We define network neutrality for the first time in the context of blockchains; we introduce the design principles of such a neutral BDN, and outline its fairness and counter-discriminatory properties.
- Our BDN is protocol- and network-agnostic, allowing improvements in the underlying infrastructure to help the cryptocurrency community as a whole rather than a select few.

BLOXROUTE TOKEN (BLXR) is an ERC20 token that supports *bloXroute*'s goal: promote the success of all cryptocurrencies. To this end, BLXR does *not* compete with other cryptocurrencies. Instead, by passing *all* revenues received

[1]We interchangeably use the terms "P2P network" and "Peer Network."

by *bloXroute* (expected at the order of billions of USD, see Section VIII-A) to BLXR holders, the success of BLXR becomes tied to *bloXroute*'s success, and to the success of all other cryptocurrencies. BLXR thus aligns the incentives of the entire ecosystem: *bloXroute*, cryptocurrencies, users, miners, and investors.

III. Blockchain Scalability Problem

Here, we introduce the blockchain scalability problem. Readers that are not familiar with the blockchain technology and terminology are encouraged to first read Appendix A, which provides the necessary background.

A. Throughput Scalability

In Bitcoin, as in other cryptocurrencies and blockchain systems, system throughput is measured by the number of transactions per second (TPS) it supports. Since the Bitcoin network produces one 1 MB block about once every 10 minutes with an average transaction size of 544 bytes [2]. The system handles an average of 1764 transactions per 10 minutes, or 2.94 TPS. In comparison, Visa performs 2000 TPS on average, with an average daily peak of 4000 TPS, and can support up to $56,000$ TPS. Moreover, cryptocurrencies aim to enable machine-to-machine low fee transactions of very small sums (mircro-payments), which are expected to require a considerably higher throughput than Visa, MasterCard, and PayPal combined [3].

The system throughput directly depends on 2 parameters: the block size (B), *i.e.*, the number of bytes which can contain transactions in each block, and the inter-block time interval (t_B), *i.e.*, the average time required for the system to mine a new block. As noted above, in Bitcoin $B = 1$ MB and $t_B = 600$ seconds, which allows 2.94 TPS. To improve Bitcoin's throughput, it is possible to increase B to include more transactions, and to reduce t_B, so that blocks are mined at a higher rate. However, these parameters cannot be arbitrarily changed, as we detail below.

B. Scalability Constraints

In Bitcoin, it has been shown that a modern processor (CPU) can support thousands of transactions per second, while disk I/O can support hundreds of thousands of transactions per second [4]. In contrast, the capacity of the P2P propagation model is orders of magnitude more restricted, and is insufficient for wide real-world adoption. We conduct a detailed analysis of the propagation model, provided in Appendix B. The key insight from our analysis is that increasing the block size (B) by a factor of X also increases the time required for a block to propagate by the same factor X. This effect was also empirically found in previous studies [4], [5].

Below we explain how a long propagation time causes blockchains to unravel, and why *no* blockchain can scale significantly based on the existing P2P propagation model. We further explain how reducing the inter-block time interval (t_B) causes the exact same effect as increasing the block size (B), *i.e.*, they both cause blockchains to unravel if used for significant scaling.

C. Block Propagation Time

1) Security and Usability

The most crucial effect of the block propagation time is the possibility for transactions to be undone, *i.e.*, removed from the blockchain. A transaction can be undone if a fork occurs and the block which contains it gets orphaned. Broadly speaking, a fork occurs when a miner mines a new block on top of a previous block, rather then on top of the most recent block. Since the blockchain incentive system incentivizes mining on top of the most recent block, forks occur because miners have not yet received the most recent block. The block propagation time, *i.e.*, the time required for a new block to propagate throughout the system, therefore defines the opportunity window in which forks may occur. The longer the propagation time, the higher the probability for a fork to occur.

Consider Bitcoin's mining, which follows the exponential distribution with a mean of 600 seconds ($t_B = 600$), mining a new block every 10 minutes on average. Further consider the time required for a block to reach 90% of the network ($t_{90^{th}}$) to be the block propagation time. The probability for a fork to occur therefore approximates [6]:

$$P(fork|t_B = 600) = 1 - e^{\frac{-t_{90^{th}}}{600}}$$

Based on the above, the probability for a fork to occur is $P(fork) = 1.915\%$ for a propagation time of $t_{90^{th}} = 11.6$ seconds, which was the average propagation time observed in March, 2017 [7]. Due to the non-negligible probability for a fork to occur, it is considered best practice to wait for several blocks, *e.g.*, 6, to be mined on top of a transaction before deeming it secure, and wait longer times for larger transactions. For such a transaction to be undone, a fork must not be resolved for 6 consecutive blocks, which has a probability of: $P(6\ blocks\ fork) = P(fork)^6 \approx 10^{-10}$.

An attempt to increase the block size (B) by a factor of 10, which would increase system capacity to $\sim$ 30 TPS, would increase the propagation time to $t_{90^{th}} = 116$ seconds. This in turn would increase the probability for a fork to occur to $P(fork) = 17.58\%$, which is unacceptable for real-world usability. More importantly, it would increase the probability for a fork to remain unresolved for 6 blocks by a factor of 600,000, and *users will have to wait for* 14 *blocks to be mined* to maintain the same level of confidence,

Scaling the system to $\sim$ 300 TPS, which is *at least one order of magnitude too small for wide real-world adoption*, would keep the blockchain at a continuous state of fork.

2) Decentralization

Block propagation time also affects the ability of nodes to participate in the Bitcoin network, as nodes must be capable of receiving blocks at a higher rate than they are produced. Failing to achieve this, nodes cannot track the balances stored in the blockchain, and thus they cannot determine the validity of transactions and blocks, and are in effect excluded from the Bitcoin Network. To allow 90% of nodes to remain in the network, the propagation time to 90% of the network must be smaller than the inter-block time interval:

$$t_{90^{th}} < t_B$$

The profitability of miners, and thus the underlying decentralization of the blockchain, is also affected by the block propagation time. Once a new block is mined, miners which have not yet received the new block become considerably less profitable, as any block they mine will cause a fork and are likely to be orphaned. The probability for a block to be orphaned depends at the rate at which it is propagated to the network. Thus, it is in the miners' interest to receive blocks as soon as possible, and to have their own blocks propagate as fast as possible. Large mining operations, known as *mining farms*, invest large sums in mining hardware and infrastructure. This results in their blocks being mined by a larger fraction of the network much more quickly than a smaller mining operation. Further, large mining farms also coordinate and construct ad-hoc networks between themselves, resulting in even more centralization pressure. To compete, small mining operations must invest proportionally larger sums to achieve the same networking performance, and are thus less profitable. Since the security of Bitcoin and other blockchain systems depend on the decentralization of mining, such a centralizing force has an adverse effect on their security.

3) Scaling Using Shorter Time interval

Scaling the throughput of a blockchain system can also be achieved by reducing the inter-block time interval (t_B), *i.e.*, the average time between blocks. However, the probability for forks to occur depends on the ratio between the propagation time ($t_{90^{th}}$) and the inter-block time interval (t_B), as is evident from the equations above. Therefore, reducing t_B by a factor X would have the *exact* same effect on the probability for forks as increasing the block size (B) by the same factor X. Thus, scaling via a shorter inter-block time interval (t_B) would have the same effects on the system security, usability, and decentralization. We provide additional details in APPENDIX B.

IV. RELATED WORK

A. Centralized Propagation Systems

While *bloXroute* is the first propagation system that addresses the blockchain scaling problem without a centralized trusted intermediary, block propagation systems do exist in Bitcoin. In particular, in order to minimize the negative effects of long block propagation times, as well as to put smaller miners on equal terms with larger mining farms, centralized Bitcoin relay networks were deployed.

1) Bitcoin Fast Relay Network / FIBRE

The first relay network to be deployed, the Bitcoin Fast Relay Network (BFRN) relays blocks using multiple gateways around the globe to reduce block propagation time for miners. BFRN focuses on utilizing low-latency connections and block compression to reduce the block propagation time. BFRN was later replaced by FIBRE, which uses a similar architecture while utilizing fiber-optic wires and forward error correction (FEC) to further reduce latency and packet error rate, aiming to minimize the number of RTTs required to propagate a block.

2) Falcon

The Falcon Network was deployed (by two members of our *bloXroute Labs* team) after BFRN and before FIBRE and aims to reduce block propagation time by using cut-through routing [8], where relay-nodes relay the first bytes of an inbound block as soon as they arrive rather than wait for the entire block to arrive.

While both FIBRE and Falcon have greatly reduced orphan rates in the Bitcoin network, Bitcoin cannot rely on these services to achieve higher throughput for multiple reasons. First and foremost, centralized systems place the control over which transactions are included in the blockchain, and which miners may participate, in the hands of their operators. Indirectly, they place this control in the hands of law enforcement and rule makers where they reside. The administrators may reject blocks which contain transactions among unauthorized parties, or blocks mined by unauthorized miners, according to their own policies, business interests, or legal requirements. *bloXroute* addresses this issue by being inherently ignorant of blocks' content, origins, and their receivers, and by making itself auditable by the global Peer Network it serves.

Second, these networks are operated on a volunteer basis by small groups, and are dependent on their goodwill and funding, which is a precarious foundation for Bitcoin's stability and scalability. Indeed, a notice of BFRN's shutdown was announced without any ready replacement in place. In contrast, *bloXroute* is designed as a sustainable operation which allows cryptocurrencies to safely utilize it for their scalability needs.

B. Off-Chain Scaling Solutions

An alternative approach, using *off-chain transactions*, aim to reduce some of the redundancy on the main blockchain. Generally speaking, an off-chain scaling solution will open up a payment channel between two parties, have the parties exchange funds while keeping track of intermediate balances, and then post a settlement transaction on the blockchain. These solutions include the Lightning Network [9], TeeChan [10], and more.

These solutions are promising and are complementary to *bloXroute*'s proposition. If the underlying blockchain can support 1000 times the number of transactions as before thanks to *bloXroute*, and if off-chain transactions increase the throughput by another factor of 1000, then that blockchain's throughput has increased by 6 orders of magnitude.

C. On-Chain Scaling Solutions

On-chain scaling solutions usually involve modifying the consensus protocol in some way to achieve higher throughput. One such approach, known as "sharding", splits the blockchain into several smaller "shards", which are maintained and interleaved in a fashion that aims to keep blockchain's original security properties while only requiring a full node to track one shard instead of the full blockchain.

Other approaches, such as Bitcoin-NG [11], suggest to replace blocks by a stream of transactions, or forgo them altogether, while still other systems aim for nodes to place trust in specific nodes, and to assure their honest behavior through the ability to replace them. There are also newer consensus protocols based on proof of stake, such as Casper [12] or Algorand [13]. We point interested readers to a survey of

consensus protocols for tolerating Byzantine faults, that are used in the state-of-the art blockchain systems [14].

While the above approaches show potential, their robustness, security, usability, and adoption rates in practice remains to be seen. However, *all* on-chain scaling solutions will perform strictly better with a faster network layer and this is where *bloXroute* improves their performance. Indeed, in every distributed consensus protocol, every honest node must reach the same decision. Thus, regardless of the consensus protocol, every honest peer must obtain information about each transaction in the system. *bloXroute* focuses on this particular problem, which is fundamentally a broadcast problem, since every valid piece of information (transaction/block) must be propagated to every honest peer in the system. *bloXroute* is thus complementary to a native consensus protocol used, and it is capable of boosting up the performance, often dramatically, for *any* blockchain.

V. BLOXROUTE: SYSTEM VISION AND GOALS

bloXroute's goal is to enable cryptocurrencies and blockchain systems to scale to thousands of *on-chain* transactions per second. Moreover, it aims to provide said scalability to numerous cryptocurrencies and blockchains simultaneously, utilizing a global infrastructure to support distributed blockchain systems in a provably neutral fashion. Here, we outline the system's trust model and the components it utilizes to achieve scalability, to prevent discrimination, and to enable new features for the blockchains it serves.

A. Trust Model

bloXroute's trust model is based on two observations. First, we observe that long block propagation times will not allow trustless P2P blockchains, *e.g.*, Bitcoin, to scale to thousands of on-chain transactions per second. Second, we observe that small centralized systems scale very well by placing trust in a small subset of participants, and passing them the control over the transactions included in the blockchains, *e.g.*, Ripple [15], EOS [16], BitShares [17], Steem [18]. However, such centralization defeats the single most notable aspect of cryptocurrencies: the distribution and decentralization of control over transactions. Providing control over a blockchain's transactions to a limited number of participants allows said participants to collude, censor, and discriminate between users, nodes, and miners. A limited participant set also reduces the number of nodes a malicious actor has to compromise to control the system.

bloXroute gets around this tradeoff by reversing the direction of trust in centralized systems. While centralized systems place trust in a subset of nodes to enable scalability, *bloXroute* enables scalability by using a small set of servers which place trust in the entire network instead. The system utilizes a Blockchain Distribution Network (BDN) to enable scaling, yet nodes need not place any trust in the BDN. Instead, the BDN blindly serves the nodes, without knowledge of the blocks it propagates, their origin, or their destination. Moreover, its behavior is constantly audited by the nodes it serves, and it is incapable of discriminating against individual nodes, blocks, and transactions. While such a design places the BDN at a disadvantage compared to the nodes it serves, its robustness allows it to withstand dishonest and malicious behaviors.

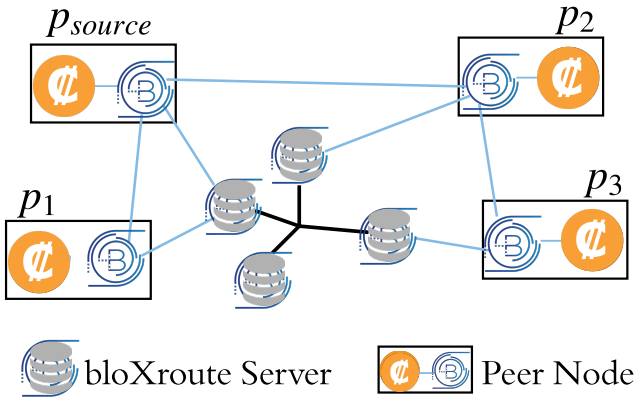

Fig. 1. The components of the *bloXroute* system: the *bloXroute* BDN, and the Peer Network nodes utilizing it. Each Peer Network node runs a Gateway process as an intermediary between its blockchain application and the *bloXroute* BDN.

B. System Components

The *bloXroute* system consists of two types of operational networks, as shown in Figure 1:

- *bloXroute* is a high-capacity, low-latency, global BDN network, optimized to quickly propagate transactions and blocks for multiple blockchain systems.
- *Peer Networks* are P2P networks of nodes which utilize *bloXroute* to propagate transactions and blocks, while carefully auditing its behavior. Each Peer Network consists of all the nodes using a specific protocol. For example, all the Bitcoin nodes utilizing *bloXroute* form a single Peer Network, while all the Ethereum nodes utilizing *bloXroute* form a different Peer Network.

bloXroute propagates blocks on behalf of the Peer Networks' nodes. However, contrary to relay networks, *bloXroute* propagates blocks without knowledge of the transactions they contain, their number, their sums, the "wallets" or addresses involved, the miner to produce each block, nor the actual origin of the node that creates a block. Furthermore, *bloXroute* cannot infer the above characteristics even when colluding with other nodes of the Peer Network, or by analyzing blocks' timing and size. *bloXroute* cannot favor specific nodes by providing them blocks ahead of others, and cannot prevent any node from joining the system and utilizing it. *bloXroute* can only propagate all blocks to all its Gateways fairly.

The *bloXroute* system as a whole is protocol-agnostic, providing its scaling services to numerous cryptocurrencies and blockchains simultaneously. The system operates at the transport layer of the OSI model (Layer 4), interacting with both the application layer and the networking layer, and provides service to whichever blockchain protocol is running at the application layer.

C. bloXroute-Supported Features

Here, we summarize the main features that a *bloXroute*-supported blockchain can attain. As a global network, *bloXroute* is capable of dramatically increasing a blockchain performance in the following ways.

- *Scalability*. A blockchain that utilizes the *bloXroute* network will be capable of performing thousands of *on-chain* transactions-per-second, via effective system-wide caching and cut-through routing, today. In the near future, additional orders-of-magnitude improvements are attainable thru the use of more sophisticated data structures and designated networking hardware.
- *Confirmation Times*. The *bloXroute* system significantly shortens confirmation times for a *bloXroute*-supported blockchain transactions, *i.e.*, at the order of tens of milliseconds.

VI. BLOXROUTE: PROVABLE NEUTRALITY

Here, we outline the mechanisms and policies that make *bloXroute* a *provably neutral* network. *bloXroute* can only propagate all blocks to all its Gateways fairly, and it is incapable of discrimination due to the auditing performed by the Peer Network nodes.

A. Counter-Discrimination Mechanisms

1) Encrypted Blocks

To prevent *bloXroute* from stopping the propagation of any block based on its content, *i.e.*, based on the wallets, addresses and sums of a block's transactions, its timestamp, its coinbase transaction, or any other attribute, blocks are propagated after being encrypted. *bloXroute*'s encryption also alters the block size, hiding the number of transaction and their total size. A block's encryption key (k_1) is only revealed after the block had been propagated, and is propagated directly over the Peer Network. k_1's minuscule size, only several bytes, allows it to quickly propagate directly over the Peer Network, and *bloXroute* is powerless to stop it.

2) Indirect Relay

In order to ensure *bloXroute* is not preventing individual nodes from propagating their blocks, nodes do not propagate blocks directly to *bloXroute*. Instead, a node wishing to propagate a block will first propagate it to a peer on the Peer Network, which will relay it to *bloXroute*, obscuring the block's origin from *bloXroute*.

In addition to indirectly relaying blocks to *bloXroute*, nodes may request their peers to relay to them incoming blocks arriving from *bloXroute*. This ensures that *bloXroute* cannot discriminate against nodes through late delivery of blocks since nodes are not required to directly interact with *bloXroute* in order to benefit from its service. The short delay (0.5 RTT) such nodes experience overlaps with the time required for nodes to receive k_1, nulling any negative effect.

3) Test-Blocks

While *bloXroute* is oblivious to which node originated each block, it may attempt to block or stall blocks arriving from some subset of nodes, affecting all the blocks they relay. In order to detect and prevent such behavior, nodes must be capable of continuously monitoring *bloXroute*'s service. Such monitoring is achieved by allowing nodes to send encrypted invalid blocks, *test-blocks*, directly to *bloXroute*, and measuring the time required for peers to report the arrival of the test-blocks. *bloXroute* is unable to employ discriminatory policies over valid blocks alone, and to faithfully propagate test-blocks, since the two are indistinguishable until their keys are published.

4) Sustainability through Peer Auditing

bloXroute provides a provably neutral block dissemination service by allowing the Peer Network nodes, via Gateways, to continuously monitor its behavior using test-blocks, and through its willingness to propagate un-validated encrypted data. This puts *bloXroute* at a disadvantage in comparison to its users, and opens the door to resource-wasting malicious behavior, which *bloXroute* is provisioned to withstand.

bloXroute's robustness and service incurs costs, namely, the delivery of large traffic volumes to a large number of nodes. For example, assuming a network of $10,000$ full nodes, *i.e.*, similar in size to today's Bitcoin network, each of which creating four 1 MB test-blocks per day, *bloXroute* would deliver 100 TB per day, or 9.26 Gbps. At the same time, assuming transactions are created at a rate of $3,000$ TPS, their delivery would require additional 132 Gbps. Note that the bandwidth required to support test-blocks increases exponentially as the number of full nodes increases, while the bandwidth required to support higher TPS does not. The cost of supporting a reliable, low-latency, global infrastructure which immediately delivers these large traffic volumes is non-negligible.

To assure *bloXroute*'s sustainability, transactions may include a minuscule, optional and voluntary payment to *bloXroute*, which provides greater incentives for miners to include them. The validation of such payments is done *by the Peer Network nodes*; the nodes validate that as blockchains process ever increasing volumes of transactions per second, and as the cost of supporting them increases, a fraction of the capacity *bloXroute* creates is dedicated for transactions which contain such payments. Thus, transactions can opt-in to this additional capacity by including a payment to *bloXroute*, which will reduce the overall fee they must carry. Since this additional capacity has lower demand and excess capacity, which will always outweigh the payment to *bloXroute*. For each transaction, *bloXroute*'s optional payment is 10% of the miner fee. Note that *bloXroute*'s payment is dwarfed by the costs saved and by the capacity enabled, which translate to orders of magnitude less fees per transaction, order of magnitude more fees collected by miners, while maintaining *bloXroute*'s profitability and sustainability.

5) Partial Disclosure of Peers

The key attribute of the *bloXroute* system is the ability of Peer Network nodes to audit the behavior of *bloXroute*. To that end, nodes relay test-blocks to *bloXroute*, and validate their peers quickly receive them. However, *bloXroute* and/or colluders might attempt to relay a node's blocks only to its immediate peers, and not to the entire Peer Network, causing the node to falsely believe its blocks are relayed to the entire

network. To prevent such a behavior, Peer Network nodes do not reveal all the nodes they are aware of. Instead, nodes conceal half the nodes they are aware of, including half of their immediate peers. Thus, if an adversary were to analyze a nodes' known peers, it will be unable to determine which nodes are its immediate peers and which are not.

B. Countering the Different Discrimination Forms

Here, we analyze the different forms of discrimination a malicious BDN might attempt to employ, and describe how the counter-discrimination mechanisms described in Section VI-A prevent *bloXroute* from employing them. We assume no colluding between *bloXroute* and Peer Network nodes, and defer the analysis of colluding to Section VI-C.

1) Content-based Discrimination

One form of discrimination is to stall or prevent the propagation of blocks based on their content, *i.e*, based on the wallets, addresses and sums of a block's transactions, its timestamp, its coinbase transaction, or any other attribute. To prevent *bloXroute* from such discrimination, blocks are encrypted prior to their propagation. Furthermore, *bloXroute*'s encryption is padded to hide the block size. Each block's unique encryption key (k_1) is only revealed after the block is propagated, and k_1's small size allows it to quickly propagate over the Peer Network. Thus, *bloXroute* is powerless to stop k_1's propagation.

2) Individual Node Discrimination

A different form of discrimination is to prevent individual nodes from propagating their blocks, based on nodes IP address, their operators' identity, node implementation, or any other node attribute. To ensure *bloXroute* cannot discriminate in this fashion, Peer Network nodes relay their blocks indirectly. Rather than transmitting a block directly via *bloXroute*, nodes propagate blocks to *bloXroute* through their peers, preventing *bloXroute* from knowing the origin of the blocks it receives.

3) Large Scale Node Discrimination

Another form of discrimination is to stall or drop blocks arriving from a large subset of nodes, affecting all the blocks they relay. However, *bloXroute* is providing service to some nodes, relaying blocks through these nodes will negate the discrimination. The Peer Network can detect this and decide the best venue for broadcasting their blocks by sending test-blocks directly through *bloXroute*.

4) Transaction Discrimination

To prevent *bloXroute* from rejecting transactions, the Peer Network can propagate them without relying on *bloXroute* at all. Since blocks are encrypted when relayed through *bloXroute*, miners are free to include any transaction in the block without interference from *bloXroute*. Simply put, *bloXroute* cannot reject specific transactions, nor can *bloXroute* avoid relaying blocks which contain specific transactions. *bloXroute* can only propagate all blocks to all its Gateways fairly.

5) Discriminating Block Delivery

There are several forms of block delivery discrimination which a BDN network might employ. First, it can discriminate *in favor* of some nodes, delivering them blocks ahead of other nodes. Second, it can discriminate against individual nodes by postponing block delivery, or not delivering them blocks at all. Third, it can cease to deliver blocks to majority of nodes, and only serve a small subset of nodes. Lastly, it can cease to deliver blocks completely, either maliciously or as a result of a system failure.

To prevent *bloXroute* from discriminating in favor of individual nodes, a block's encryption key (k_1) is only propagated after the node which originated it (p_{source}) learns of the block's propagation from its peers. Thus, any node $p_{privileged}$ to receive the block ahead of time will be forced to wait until it receives k_1 from its peers, which will only commence once the block is delivered to p_{source}'s arbitrary peers, thus placing $p_{privileged}$ on par with its peers.

To protect themselves from late block delivery, nodes compare between their own test-blocks propagation time and those of their peers, which will indicate whether or not they are being discriminated against. If a node identifies such discrimination, nodes will request block delivery from their peers rather than relying on *bloXroute*. This will place the discriminated nodes on par with their peers, as the short delay they suffer (0.5 RTT) overlaps with the time required for k_1 to propagate.

If *bloXroute* ceases to deliver blocks completely, whether maliciously or due to a large scale system failure, the Peer Network will replace it with an alternative BDN. Nodes can deploy their own alternative BDNs by running bloXroute's code on their own network of machines, incurring only low costs by limiting the test-blocks rate they allow. During this time, *bloXroute* can be replaced permanently, if the need arises. If the discrimination is due to a system failure rather than malicious behavior, the peers will return to using *bloXroute* once the failure is resolved.

Note that the existence of an alternative to *bloXroute* is sufficient to deter any malicious behavior on its part, preventing the need to make use of the alternative BDNs.

C. Preventing Colluding and Malicious Behavior

While we have shown that *bloXroute* cannot engage in malicious behavior unilaterally, we now consider the scenario where it colludes with some fraction of the Peer Network. Note that *bloXroute*'s design cannot solve collusion that is inherent in the underlying cryptocurrency, *e.g.*, a 51% attack, nor are we aiming to minimize its effectiveness. Rather, *bloXroute*'s design goal is not to exacerbate existing attack vectors, and not enable new attack vectors.

1) Colluding to Prevent Block Propagation

bloXroute and its colluding nodes might attempt to prevent the propagation of a block, either based on its content or its origin. However, as outlined above, blocks are transmitted to *bloXroute* indirectly, and are encrypted prior to their propagation. Thus, once the block is relayed to an honest peer, the honest peer will relay it to *bloXroute*, which will be unable to distinguish it from a test-block. The only fashion in which *bloXroute* can prevent the block propagation is to drop *all* test-blocks, which will cause all nodes to abandon it, and would fail to stop the block propagation.

As an example, assume a node p_{source} wishes to propagate block b. p_{source} first relays b'^1, a version of b encrypted using the key k_1, to p_1. Further assume p_1 to be actively colluding with *bloXroute*. To prevent or delay b's propagation, colluders will refrain from propagating b'^1 to the entire Peer Network, and possibly share all knowledge of p_{source}, b, k_1, and b'^1 among themselves. Once p_{source} relays the block to p_1, it starts relaying b'^i, and k_i to additional peers $\{p_i \mid i > 1\}$, until it learns of a block that was successfully relayed. For the attack to succeed, *bloXroute* must avoid or delay the propagation of blocks $\left\{b'^i \mid i > 1\right\}$, as they arrive from their respective nodes. Failing to do so, the attack will only delay b's propagation by the time required for p_{source} to relay the block to its first non-colluding peer, identically to the same attack vector in the P2P trustless model.

Since (i) p_{source} selects its peers at its own discretion, (ii) each peer p_i relays a version of b which is encrypted with a different key, and (iii) the encryption obscures both b's content and size, it is impossible for *bloXroute* to distinguish between incoming test-blocks and encrypted versions of b. Thus, to affect b's propagation, *bloXroute* must refuse all incoming blocks arriving from non-colluding nodes, possibly until a different block is provided by the colluding nodes. Such behavior is immediately visible to all the nodes of the Peer Network, as they will fail to get reports of arrival for their own test-blocks.

We note that a necessary condition for the attack to be launched is for p_1, the first peer to which p_{source} relays the block, is colluding with *bloXroute*. We further note that deliberate failure to relay blocks is already an existing attack vector in the existing P2P trustless model, with the same probability of success. The result in both models, delaying the propagation of a b until it is sent to p_i, the first non-colluding node, is identical.

We further note that while it is also possible for *bloXroute* to reject only a portion of blocks arriving from non-colluders, rather than all of them, such an attack is even less effective. For example, rejecting 50% of blocks arriving from non-colluders will only delay b's propagation by half the time required to relay a block through a peer, on average, while clearly visible to at least 50% of the nodes. Increasing the percentage of rejected nodes increases its visibility even further, while decreasing its visibility reduces its effect.

Lastly, it is worth noting that honest nodes of the Peer Network can determine whether or not their test-blocks are being relayed or not. If a node's test-blocks are being relayed, *i.e.*, in absence of ongoing node discrimination, nodes can directly relay their blocks to *bloXroute*, and obscure the block's validity even from its peers.

2) Colluding to Prevent Block Delivery

In addition to colluding to prevent block propagation, nodes might attempt to collude with *bloXroute* in order to prevent or delay block delivery to a subset of the Peer Network nodes. When such an attack is launched against a small number of nodes, it is easily discovered by the targeted nodes, since their honest immediate peers, which are unknown to the colluders, will notify them of the blocks and encryption keys they receive. To protect themselves, discriminated nodes can request any of their immediate peers to relay to them all incoming blocks. Such a request does not place the discriminated nodes at the selected peer's mercy, as it continues to receive blocks from its other peers and from *bloXroute*. Such a request also doesn't place a heavy burden on the node's peer, as they control the number of nodes they relay traffic to. The presence of colluding nodes does not affect the attack's effectiveness, which is identical to such an attack in a system using the P2P trustless model.

VII. Onboarding Process

A cryptocurrency that wishes to take advantage of *bloXroute* can do so in the following steps. The first nodes and miners of a specific cryptocurrency who wish to utilize *bloXroute* are required to do nothing more than simply running *bloXroute*'s Gateway process in parallel to their blockchain application. For onboarding purposes, *bloXroute* will run a sufficient number of BDN nodes around the world, so that users can propagate blocks and receive transactions faster than any other peer. As more nodes use *bloXroute*, cryptocurrencies will see considerably fewer forks and stronger security guarantees that result from *bloXroute*'s superior block distribution. For example, Ethereum will see fewer uncle blocks, and will achieve higher throughput.

As a second step, cryptocurrencies can adjust their protocol to capture more of the capacity increase provided by *bloXroute*, *e.g.*, increasing the block size and reducing the inter-block time interval. *bloXroute* requires no further changes to the protocol, and will allow cryptocurrencies to use the network for free as long as they produce no more than 100 transactions per second (TPS). As a reference, Bitcoin supports only 3 TPS. Thus, *bloXroute* allows to increase capacity by a factor of 33 today, without requiring any fees and any protocol change beyond adjusting the block size and inter-block time interval.

Cryptocurrencies require no protocol change beyond adjusting the block size and inter-block time interval to fully utilize *bloXroute*'s capacity. Once the 100 TPS threshold is reached, there becomes an increasing incentive for users to make the minuscule payments to *bloXroute* to reduce their fees. This in turn would eventually cause users to demand the implementation of making such payments easily from their wallets and nodes. Note that *the protocol itself does not change*; the validity requirements remain the same, as is the structure of blocks and transactions, and all the messages among nodes.

VIII. Bloxroute Token (BLXR): Token Dynamics, and Revenues

Bloxroute Token (BLXR) is an Ethereum ERC20 token that supports *bloXroute*'s goal: promote the success of all cryptocurrencies. By passing *all* funds received by *bloXroute* to BLXR token holders, the success of BLXR becomes tied to *bloXroute*'s success, and to the success of all other cryptocurrencies. BLXR thus aligns the incentives of the entire ecosystem: *bloXroute*, cryptocurrencies, users, miners, and investors.

A. BLXR *Value*

It is impossible to determine the volume of transactions once machine-to-machine micro-payments are enabled. However, given that (i) credit card companies already process $5,000$ TPS today, despite high fees, (ii) it would require 100,000 TPS to support all Facebook users to perform 4 transactions per day, and (iii), futuristic machine-to-machine micro-payments require considerably more transactions than human interactions, a demand of 200,000 TPS across all cryptocurrencies can be considered a conservative estimate.

Assuming a mining fee of 0.005 USD, *i.e.*, half a cent, a payment of 0.0005 USD to *bloXroute* (10% fee to *bloXroute*), and broad *bloXroute* adoption (200,000 TPS across all cryptocurrencies), *bloXroute* revenues would amount to over 3.1 Billion USD *per year*. All these earnings will be immediately directed to BLXR token holders, as outlined below.

B. BLXR *and Revenue Distribution*

BLXR was designed with two goals in mind: to align the incentives of the entire cryptocurrency ecosystem and to be a vehicle for investment in *bloXroute*. Transaction fees received by *bloXroute* will be automatically directed to a newly created pooled account, with one wallet per supported currency, which we refer to as the BLXR-Reserve. As transactions are processed using *bloXroute*, the BLXR-Reserve will receive cryptocurrencies in the form of transaction fees. Holders of BLXR tokens will be allocated dividends from this pool, equal to their pro rata share of the fees received to date as such fees are received. At any given time, a holder's *user balance* is equal to the dividends accrued to date less any withdrawals. In the event that a holder sells his BLXR tokens, his current user balance is unaffected (and may be withdrawn now or in the future), but his right to receive future dividends is reduced in proportion with the amount of BLXR tokens sold.

Consider a simplified example in which 1 BTC is collected as a transaction fee in time $t1$. If User A owns 10% of BLXR tokens, then User A can withdraw 0.1 BTC at $t1$, or at any future point in time. Assume that User A does not withdraw its share (0.1 BTC) at $t1$, and that User A sells all his BLXR tokens to User B (which held no BLXR tokens) at $t2$. User A will still be able to withdraw its share (0.1 BTC) at any time after $t2$, despite the fact that User A holds no BLXR tokens after $t2$. Further assume that an additional 2 BTC is collected as a transaction fee in time $t3$. At any time after $t3$, User B (now owner of 10% of BLXR tokens) can withdraw its share (0.2 BTC), while User A remains unaffected.

IX. Conclusion

In this paper, we presented *bloXroute*, the first BDN, which features a radically novel approach to resolving the blockchain scaling problem: it introduces a global network infrastructure to boost scalability, yet retains the decentralization of control over transactions in a blockchain, via neutral and auditable network design. It attains scalability by implementing an effective broadcast primitive. It attains neutrality by supporting encrypted blocks and by obscuring blocks' origin via peer relaying. Finally, it attains auditability by enabling users to directly and actively probe, via Gateways, the network in a systematic manner. *bloXroute* is protocol-agnostic, capable of supporting multiple blockchains simultaneously, and fully unleashing their indisputable potential.

bloXroute is supported via BLXR, a token that provides its owners access to a pro-rata share of all payments made to *bloXroute*.

Appendix A
Background

A. Bitcoin and the Blockchain

Bitcoin [1], [19] is the first blockchain system, and the first cryptocurrency to gain considerable traction globally, with a market capitalization measured in hundreds of billions of USD. At its core, Bitcoin is a distributed system which allows its users to hold a balance and make transactions of Bitcoins, *i.e.*, of currency, and distributedly maintain a single ledger of all transactions. Transactions are not added to the ledger individually, rather, they are being added in batches, known as *blocks*. The result is a chain of blocks which contains the entire history of all Bitcoin transactions, known as the *blockchain*.

To understand how Bitcoin transactions are created and the blockchain maintained, assume a user, Alice, is buying an item from another user, Bob, and wishes to pay for it in Bitcoin. Each user controls a *wallet*, which is a simple private key and a public key pair. To pay Bob, Alice locally creates a new transaction $t_{A\to B}$, which passes some amount of Bitcoins from her public key, also known as her *address*, to Bob's address, and signs it using her private key. Alice then propagates $t_{A\to B}$ to all other Bitcoin users. The Bitcoin network, which contain all Bitcoin users, is a peer-to-peer (P2P) network. Every Bitcoin user, also referred to as a *node*, or a *peer*, who receives $t_{A\to B}$ validates that (i) all the transactions paying Bitcoins to Alice's address, minus the sums spent from her address, leave a balance which is equal or greater than the amount spent in $t_{A\to B}$, and (ii) $t_{A\to B}$ contains a signature which requires knowledge of Alice's private key to be created. If these two conditions are met, $t_{A\to B}$ is deemed valid, and users which receive it will propagate it to their peers. It is worth noting that a single entity may control any number of wallets, and for each wallet to control any number of public keys.

1) Miners

In addition to regular Bitcoin users, some nodes in the Bitcoin network attempt to aggregate the transactions they receive into new blocks, which will be added to the blockchain. It is only once a transaction is included in the blockchain that it is considered to have taken place, while transactions which still await to be included are not. Such nodes are called *miners*, and the process of attempting to create a new block is known as *mining*. There are two monetary incentives for mining. First, each block contains a unique transaction, known as the *coinbase transaction*, which passes some amount of Bitcoins to its miner's address. The blocks' coinbase transactions also provides the supply of Bitcoins, as it creates Bitcoins "out of thin air". The amount of Bitcoins produced in each block decreases exponentially, limiting the total supply

to approximately 21 million Bitcoins. Second, each Bitcoin transaction can carry a fee to whichever miner that successfully include it in a block, and miners are incentivized to include the transactions with the highest fees, since the number of transactions included in each block is limited.

To mine a new block, a miner hashes all the transactions to be included in the block, using a double SHA-256 hashing function. In addition to the transactions, the miner also hashes a timestamp, the result of hashing the previous block, and an arbitrary binary value, known as a *nonce*. For a new block to be created, *i.e.*, for successful mining, the result of the hashing must be very small. Thus, miners exhaustively try different nonce values, in an attempt to find one which produces a small enough value. The exact *target* value changes over time, in an attempt to maintain an average of one block every 10 minutes, based on blocks' timestamps.

Once a new block is found, it is propagated to the entire Bitcoin network, similarly to transactions, *i.e.*, it is validated by each node prior to its propagation. It is safe for the successful miner to propagate its block, including the nonce, since the nonce only yields a small enough value for the newly-mined block, without any change done to it. A dishonest node cannot utilize the nonce to create an alternative block, *e.g.*, with a coinbase transaction which passes the Bitcoins to the dishonest node's wallet, since such a change will cause the block's hashing to yield a different value, which deems the original nonce useless.

2) Blockchain Security

The most critical aspect of the blockchain security is that the hashing of each block also includes the value yielded from hashing the block preceding it. The immediate result of this inclusion is that any attacker attempting to alter the history of transactions, *i.e.*, the inclusion, exclusion, or alteration of a transaction in some previous block, will change the value its hashing yields, which in turn will affect the hashing value of *all consecutive blocks*, and will almost certainly invalidate each and every one of them. For such an alteration to succeed, the attacker will have to sequentially find a new nonce for every block. Moreover, it will have to do so at a higher rate than the rate at which all other miners extend the original blockchain. Thus, for such an attack to succeed, the attacker must control the majority of hashing power in the Bitcoin system [20]. While it had been shown that entities control less than 50% of the hashing power can gain unfair advantage [20], and thus can eventually eliminate smaller miners, and gain majority of hashing power, it is this unique primitive which differentiate Bitcoin and blockchain systems from previous decentralized systems.

3) Forks

A unique feature of Bitcoin, and of blockchain systems in general, is their inherent ability to overcome inconsistent views of the transaction history in a distributed manner, by defining the blockchain which required the most computation to produce as the "true" blockchain. To demonstrate this ability, consider two Bitcoin miners which happen to successfully mine a new block at approximately the same time. The two blocks differ from each other as they will contain different coinbase transactions, use different nonces, and it is very likely for them to contain slightly different transactions. Once the two blocks are mined, they are propagated in parallel to the entire Bitcoin network, resulting in some portion of the network considering one history of transactions to take place, while others consider a slightly different version of transaction history to take place. Such a situation, where two or more equally valid blockchain versions coexist is called a *fork*. While a fork is unresolved, there exists some ambiguity regarding which transactions had taken place.

Forks are resolved once a new block is mined, as it causes one prong of the fork to become longer than the other prongs, which in turn incentivizes miners to abandon the shorter prongs and attempt to mine over the longest blockchain, as any rewards gained on shorter prongs are likely to be orphaned, *i.e.*, discarded. Thus the system converges to the longest blockchain due to the selfish interests of the miners. It is possible, yet rare, for blocks to be mined over two prongs of a fork at approximately the same time, which keeps the fork unresolved, until eventually one becomes longer than the other.

Due to the possibility of forks, it is possible for a transaction to be included in a block, yet not to be included in the blockchain if said block is orphaned. The probability for a block to be orphaned exponentially decreases as more blocks are mined on top of it, in direct relation to the probability for two blocks to be mined on two prongs of a fork at approximately the same time. Thus, transactions are considered more secure as additional blocks are mined on top of the blocks containing them.

Bitcoin's model, as outlined above, does not depend on any centralized entity to track balances or to execute transactions, nor can any single entity undo transactions, confiscate Bitcoins, or alter the blockchain in any way without control over the majority of hashing power. To enforce new transactions on behalf of users, such an entity will be required to break the SHA-256 hashing. Bitcoin is thus considered a trustless system, since users do not depend on any central entity to perform any action on their behalf, nor do they rely on such an entity to provide them with accurate information, such as wallet balances and transactions validity.

APPENDIX B
SCALABILITY ANALYSIS OF THE BITCOIN NETWORK

Consider the state of the Bitcoin network in the year 2017. The network consists of approximately $9,000$ nodes ($N = 9000$) [21], the majority of which are connected to 8–12 of their peers [22], with a median latency between peers of approximately 110 milliseconds [23]. At the 50^{th} percentile, nodes upload rate is 56 Mbps ($bw_{50^{th}} = 56$ Mbps), while the 10^{th} and 1^{st} percentiles have a rate of 3.96 Mbps and 438 Kbps, respectively. Thus, the upload rate at the 50^{th}, 10^{th} and 1^{st} percentile supports $13,000$, 943 and 100 TPS, respectively. We note that global bandwidth measurements [24] show that download rates exceed upload rates by a factor of 1.85–5.81, with the exception of less constrained regions, where the average bandwidth exceeds 100 Mbps.

It is evident that the bandwidth of individual Bitcoin nodes supports increasing the system throughput by orders of mag-

nitude, and the latency among peers is not abnormally high as to pose a barrier. However, while individual nodes can easily support higher TPS, it is the distributed propagation which Bitcoin employs, also used in other blockchain systems, which significantly limits the system throughput.

A. P2P Propagation Model

Traditional P2P systems, such as Bittorrent [25], have been shown to quickly propagate data among peers. However, Bitcoin and other blockchain systems differ from such traditional P2P networks by requiring the *continuous delivery of all blocks to all peers*. They further differ by the different incentives they provide to their participants. Specifically, distributed denial of service (DDoS) attacks are prevalent in blockchain systems, and are used to gain advantages in mining, voting, and other business- and protocol-related activities. To prevent malicious nodes from flooding the network with invalid blocks, nodes use a store-and-forward propagation model, where each node downloads the full block and verifies it prior to propagating it to its peers. This model allows nodes to identify any node which propagates invalid blocks as malicious, and limits the effect of such attacks to the nodes which are directly attacked.

B. Block Propagation Time Analysis

In order for Bitcoin to function as a decentralized system, it must allow nodes to receive blocks at a higher rate than the blocks are produced. Indeed, if blocks are produced at a higher rate than a node is capable of receiving them, then said node cannot keep track of balances stored in the blockchain, cannot determine whether or not transactions and blocks are valid, and is in effect excluded from the Bitcoin Network. The block propagation time to the majority of the network ($t_{50^{th}}$) does not depend solely on a receiving node bandwidth. Rather, it depends on network topology, the bandwidth of all nodes, and the manner in which blocks propagate.

The time required for a block to propagate through the system, and how it is affected by the block size (B), can be roughly approximated based on the number of nodes (N) and their median bandwidth ($bw_{50^{th}}$). For the median Bitcoin node, the time required to transmit a single 1 MB block to a single peer (t_{hop}) is roughly

$$t_{hop} = \frac{B}{bw_{50^{th}}} = \frac{1MB}{56Mbps} = 0.143sec$$

Assuming 8 peers, the average Bitcoin node will require approximately $8t_{hop}$ to propagate a block to its peers, regardless whether if done sequentially or in parallel. However, sequential propagation allows the node's first peer to propagate the received block after t_{hop} had passed, while parallel propagation will only allow peers to propagate the block after $8t_{hop}$ have passed. Thus, to hasten block propagation when bandwidth is limited, nodes would ideally propagate blocks to their peers sequentially, rather than in parallel.

Using sequential propagation, a newly-mined block is known only to a single node, *i.e.*, its miner, at time $t = 0$, to two nodes, the miner and its first peer, at time $t = t_{hop}$, to 4 nodes at time $t = 2t_{hop}$, and to the majority of the network at time

$$t_{50^{th}} = \lfloor log_2(N) \rfloor t_{hop} = 13t_{hop} = 1.86sec$$

While this approximation does not account for network congestion, download bandwidth, messages exchange overhead, latency, packet loss, processing delay, the arbitrary topology of the nodes, and bandwidth consumed for transactions propagation, it does provide insights regarding block propagation time ($t_{50^{th}}$).

We note that while the network size (N) effect over the block propagation time ($t_{50^{th}}$) is logarithmic, *the block size's (B) effect is linear*. For example, increasing the system's TPS by a factor of 10 by increasing the block size to $B = 10$ MB would increase the time required for the median node to transmit a block to a single peer by the same factor, to $t_{hop} = \frac{10\ MB}{56\ Mbps} = 1.43$ seconds. This, in turn, would increase the block propagation time to the majority of the network by the same factor to $t_{50^{th}} = 13t_{hop} = 18.6$ seconds. The linear effect of block size (B) on block propagation time ($t_{50^{th}}$) was also empirically found in previous studies [4], [5], when block size (B) exceeds 20 KB.

C. Block Size and Inter-Block Time Interval

The positive and negative effects of increasing block size (B), *i.e.*, increased system throughput, reduced blockchain security, and node exclusion, are symmetric to the effects of reducing the average time required to mine a new block (t_B) by the same factor. For example, doubling the block size (B) would increase the system throughout by a factor of 2, and the same is achieved by halving the inter-block time interval (t_B). Similarly, doubling the block size will approximately double the block propagation time ($t_{90^{th}}$), which in turn will double the probability of forks

$$P(fork) = 1 - e^{\frac{-t_{90^{th}}}{t_B}}$$

while halving the inter-block time interval (t_B) will have the same effect. Lastly, to include 90% of the nodes in the network, the block propagation time at the 90^{th} percentile must be smaller than the inter-block time interval ($t_{90^{th}} < t_B$). Doubling the block size (B) would increase block propagation time ($t_{90^{th}}$) by a similar factor, which will have the same node exclusion effect as halving t_B.

REFERENCES

[1] S. Nakamoto. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. [Online]. Available: www.bitcoin.org

[2] Bitcion Charts and Graphs - Blockchain. [Online]. Available: www.blockchain.info/charts

[3] "Bitcoin Will Need to Scale to Levels Much Higher Than Visa, Mastercard, and PayPal Combined," 2017, www.coinjournal.net/bitcoin-will-need-to-scale-to-levels-much-higher-than-visa-mastercard-and-paypal-combined/.

[4] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, D. Song, and R. Wattenhofer, "On scaling decentralized blockchains," in *International Conference on Financial Cryptography and Data Security*. Springer, 2016, pp. 106–125.

[5] C. Decker and R. Wattenhofer, "Information propagation in the Bitcoin network," in *Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on*. IEEE, 2013, pp. 1–10.

[6] A. W. Marshall and I. Olkin, "A multivariate exponential distribution," *Journal of the American Statistical Association*, vol. 62, no. 317, pp. 30–44, 1967.

[7] BitcoinStats. [Online]. Available: www.bitcoinstats.com/network/propagation/

[8] P. Kermani and L. Kleinrock, "Virtual cut-through: A new computer communication switching technique," *Computer Networks (1976)*, vol. 3, no. 4, pp. 267–286, 1979.

[9] Lightning Network. [Online]. Available: https://lightning.network/

[10] J. Lind, I. Eyal, P. Pietzuch, and E. G. Sirer, "Teechan: Payment channels using trusted execution environments, 2017," *URL https://arxiv.org/pdf/1612.07766. pdf.[Online*.

[11] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-ng: A scalable blockchain protocol." in *NSDI*, 2016, pp. 45–59.

[12] Where's Casper? Inside Ethereum's Race to Reinvent its Blockchain. [Online]. Available: https://www.coindesk.com/ethereum-casper-proof-stake-rewrite-rules-blockchain/

[13] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling byzantine agreements for cryptocurrencies," 2017.

[14] C. Cachin and M. Vukolić, "Blockchains consensus protocols in the wild," *arXiv preprint arXiv:1707.01873*, 2017.

[15] Ripple. [Online]. Available: https://ripple.com/

[16] EOS. [Online]. Available: https://eos.io/

[17] BitShares. [Online]. Available: https://bitshares.org/

[18] Steem. [Online]. Available: https://steem.io/

[19] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better? How to make Bitcoin a better currency," in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 399–414.

[20] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *International conference on financial cryptography and data security*. Springer, 2014, pp. 436–454.

[21] Global Bitcoin Nodes Distribution - Bitnodes. [Online]. Available: www.bitnodes.21.co

[22] A. Miller, J. Litton, A. Pachulski, N. Gupta, D. Levin, N. Spring, and B. Bhattacharjee, "Discovering Bitcoin's public topology and influential nodes," 2015.

[23] State of the Bitcoin Network. [Online]. Available: www.hackingdistributed.com/2017/02/15/state-of-the-bitcoin-network

[24] Speedtest Market Report. [Online]. Available: www.speedtest.net/reports/

[25] D. Qiu and R. Srikant, "Modeling and performance analysis of bittorrent-like peer-to-peer networks," in *ACM SIGCOMM computer communication review*, vol. 34, no. 4. ACM, 2004, pp. 367–378.

Uri Klarman, bloXroute Labs CEO, is the most vocal proponent of bloXroute, which is his doctoral dissertation work at Northwestern University. Uri is an interdisciplinary networks researcher, and his work encompasses innovative uses of Computer Networks, disruptive blockchain networking schemes, alternative content distribution networks, trustless peer coordination, and security.

Soumya Basu, bloXroute Labs CTO, is a member of the Initiative for Cryptocurrencies and Contracts (IC3) group at Cornell University. He is most well known for creating the Falcon Network, which has been operational in the Bitcoin network since April 2016. Soumya's work aims to remove trust without reducing performance in cryptocurrency systems. He was awarded the NSF Graduate Research Fellowship and a paper award at ACM SIGCOMM.

Aleksandar Kuzmanovic, bloXroute Labs Chief Architect, is a Net Neutrality expert and a Full Professor in the Department of Electrical Engineering and Computer Science at Northwestern University, where he is currently on leave of absence. Prof. Kuzmanovic's work on Net Neutrality had awarded him an NSF CAREER Award, and he is one of the founders and a member of the steering committee of Google's Measurement Lab initiative for monitoring global Net Neutrality. His work and systems on congestion control, traffic analysis, and content distribution have been widely disseminated on the Internet, finding its way to millions of users.

Emin Gün Sirer, bloXroute Labs Chief Scientist, is counted among the very top blockchain and cryptocurrency researchers in the world. He is the co-director of IC3, the Initiative for Cryptocurrencies and Smart Contracts, and is an Associate Professor of Computer Science at Cornell University. His research interests span distributed systems, cryptocurrencies, and software infrastructure for large scale services, for which he received the NSF CAREER Award, and has been named in Brilliant-10 by Popular Science magazine.