Enhancing Blockchain Scalability through the Integration of Hybrid Reputation-Based Consensus and Geo-Based Layer 2 Solutions

Ammar Amjad
Computer and Information Sciences
University of Florida
Gainesville, Florida, United States
ammar.amjad@ufl.edu

Kshitij Maruti Vejre
Computer and Information Sciences
University of Florida
Gainesville, Florida, United States
kvejre@ufl.edu

Suhas Harish
Computer and Information Sciences
University of Florida
Gainesville, Florida, United States
sganjalaguntehar@ufl.edu

***Abstract*—This paper investigates the blockchain trilemma, a challenge in achieving simultaneous security, scalability, and decentralization in blockchain technology. We explore various second-layer solutions, such as sharding, side chains, rollups, and plasma, while discussing their limitations. Our proposed hybrid solution combines reputation-based consensus mechanisms and geo-based layer 2 solutions to address the trilemma effectively. The two-layer blockchain system includes a global blockchain and multiple regional sidechains, which enhance scalability, security, and localization while preserving decentralization and transparency. Reputation points incentivize miners to contribute to regional chains, and an optimization algorithm summarizes transactions for increased efficiency. Our implementation, written in Erlang, demonstrates significant improvements in transaction throughput and reduced block time compared to traditional blockchains. In conclusion, this hybrid approach provides a promising solution to the blockchain trilemma, paving the way for future advancements in the field.**

***Index Terms*—Blockchain Trilemma, Scalability, Decentralization, Security, Rollup, Reputation Points, Regional Transactions.**

I. Introduction

Blockchain was introduced in 2008 yet the blockchain trilemma remains a major obstacle to its widespread adoption.

A. The Blockchain Trilemma

The blockchain trilemma refers to the challenge of balancing decentralization, security, and scalability in a blockchain network. Decentralization ensures no central authority, reducing single-point failure risks but can affect performance and scalability. Security is crucial for data integrity and immutability but can slow down the network and limit scalability. Scalability involves handling increasing transactions and users, but current implementations struggle with transaction throughput, leading to higher transaction times and fees. Ongoing research aims to address these trade-offs and limitations.

B. Blockchain Trilemma and Its Impacts

The scalability in blockchain trilemma comes with a trade-off between security, and decentralization. The blockchain network cannot achieve all three properties simultaneously, making it challenging to maintain them as transaction volumes increase. This poses a significant challenge for developers and emphasizes the need for innovative solutions that balance these properties for practical adoption. As the use of blockchain technology grows, so do scalability issues. The increasing number of blocks and transactions leads to slower calculation times and greater latency, affecting the network's performance and limiting its capacity to meet user demands.

C. Limitations of Bitcoin Blockchain

Blockchain technology has shown its limitations over time, with Bitcoin being a prime example. It takes about an hour for a transaction to be reflected on the receiver's side, with a low rate of 7 transactions per minute. In contrast, Visa can process 24,000 transactions per second.

Mining difficulty has increased since Bitcoin's inception, with the current difficulty requiring 76 leading zeros in the hash value. This means that only 1 hash in 39 trillion has the potential to be a valid hash. The difficulty changes after every 2016 blocks added to the blockchain, which affects the time taken to mine the blocks.

Bitcoin miners use expensive and power-consuming computational setups. The mining uncertainty and low probability of finding valid blocks result in limited transactions that Bitcoin can process per second. As the number of users and transactions increases, the blockchain network struggles to handle more transactions, leading to delayed transaction processing.

The current implementation of blockchain technology calls for a better solution that addresses the trilemma between decentralization, security, and scalability. A highly decentralized and secure network might struggle with scalability, while a scalable network might sacrifice decentralization or security.

II. Literature Review

A. Related Work

Numerous solutions have been proposed in the literature to tackle the issues of security, scalability, and decentralization in

blockchain networks. These solutions include rollups, proof-of-stake cryptocurrencies, and geo-location-based latency reduction techniques. We conducted a survey using Google Scholar, Cervix, IEEE, and other sources to identify relevant papers but found none that specifically matched our proposal. However, the following closely related papers address scalability, incentivization, and latency reduction.

Gudgeon et al. [2] introduced a layer-two solution called Sok, which employs side chains to enhance the scalability of the underlying blockchain network. This solution aims to increase the main blockchain network's transaction limit by offloading some work to the layer 2 chain, thus reducing the main chain's workload and enabling faster execution of more transactions.

Sguanci et al. [10] and Marukhnenko et al. [11] investigated various techniques for improving blockchain network scalability, such as side-chains and plasma, and provided a comparative analysis of their pros and cons. Similarly, Sun et al. [4] proposed a layer-two solution based on state channels, which uses off-chain transactions to boost the scalability of the blockchain network.

Chauhan et al. [3] discussed the scalability challenges encountered by blockchain networks and proposed several solutions, including consensus algorithms, sharding techniques, and layer-two solutions. Our proposed solution aims to enhance the scalability and efficiency of the blockchain network while maintaining security and consistency, akin to the layer-two solutions suggested in [2] and [4].

Thibault et al. [5] offered a comprehensive overview of the roll-up technique, a layer-2 solution on the Ethereum network, discussing its design, implementation, and use. For guidance on selecting an incentive mechanism, Nadal and King's paper [6] introduced Peercoin and its unique consensus algorithm.

Chaudhry et al. [8] and Mingxiao et al. [9] compared various consensus algorithms utilized in blockchain technology, assessing their strengths, weaknesses, and potential improvements.

The existing literature provides a thorough overview of different methods for enhancing blockchain scalability, such as rollups, proof-of-stake cryptocurrencies, and geo-location-based latency reduction techniques. By integrating these solutions, a novel hybrid reputation-based consensus and geo-based layer 2 solution could be developed, allowing for the creation of secure, scalable, and decentralized blockchain networks.

B. Second-Layer Solutions and Limitations

The blockchain trilemma, which states that it is impossible to achieve security, scalability, and decentralization simultaneously, has led to the development of second-layer solutions to address the scalability issue. These solutions aim to offload part of the work of the main blockchain to improve transaction throughput, reduce fees, and enhance user experience. This paper explores various second-layer solutions, their limitations, and a novel hybrid solution to enhance Blockchain Scalability through the Integration of Hybrid Reputation-Based Consensus and Geo-Based Layer 2 Solutions.

1) Sharding: Sharding involves dividing the blockchain network into smaller pieces or shards to increase its scalability. Ethereum 2.0, for example, employs 64 shards to enhance transaction throughput. However, sharding introduces new security risks, data fragmentation, incentive misalignment, and communication latency across shards. These limitations necessitate a better way to scale the network without compromising transaction processing speed.

2) Side-chains: Lightning Network: Side-chains are parallel chains that coordinate and operate alongside the main blockchain. The Lightning Network is an off-chain ledger used in Bitcoin to reduce transaction fees and facilitate faster transactions. However, side-chains also come with security risks, centralization issues, interoperability challenges, speculative short-term gains, and unclear governance structures.

3) Rollups: Rollups involve bundling multiple transactions into a single transaction off the main blockchain to increase transaction throughput and reduce fees. However, rollups require aggregators, which can lead to centralization and single points of failure. Data storage on the main blockchain can be expensive and limit rollup use, and the waiting period for funds withdrawal can range from one to two weeks.

4) Plasma: Plasma employs child chains to perform transactions faster and more cost-effectively. It relies on smart contracts to ensure the security and validity of transactions in the child chain. However, bugs in smart contracts can lead to attacks on the child chain. Plasma can also cause centralization if a few nodes hold more data than others, leading to power imbalances and potential exploitation by attackers.

5) Payment channels: Payment channels allow two parties to perform transactions off the main blockchain, increasing transaction speed, reducing fees, and providing privacy. However, payment channels lead to centralization, as a third party is required to keep the channel open. Parties must also lock up their capital, limiting liquidity and access to funds until the channel is closed. Payment channels have limited scalability compared to other second-layer solutions.

Our reputation-based hybrid solution applies the Integration of Hybrid Reputation-Based Consensus and Geo-Based Layer 2 Solutions, aimed to address the limitations of existing second-layer solutions. This proposed solution leverages the strengths of existing solutions while mitigating their weaknesses.

The hybrid solution employs a reputation-based consensus mechanism, which rewards honest and trustworthy nodes and punishes malicious ones. This approach increases security and reduces the risk of centralization. Additionally, geo-based layer 2 solutions divide the network based on geographical regions, reducing latency and enhancing communication across shards.

By combining these approaches, the proposed solution can:

Improve security by leveraging a reputation-based consensus mechanism, which discourages malicious behavior and promotes network integrity. Increase decentralization by

employing geo-based layer 2 solutions, which encourage a more equitable distribution of nodes and prevent the formation of centralized power structures. Enhance scalability by utilizing the strengths of multiple second-layer solutions, such as sharding, side-chains, rollups, and plasma, to address the limitations of each method. In conclusion, the proposed hybrid solution offers a promising approach to addressing the blockchain trilemma by integrating reputation-based consensus mechanisms

III. Our Solution

A. Need for Regional Chains

We are trying to propose a two-layer blockchain consensus protocol framework with 4 permanent regional sidechains and one global blockchain to increase the scalability of the existing blockchain. Therefore, four regional blocks and one global block can get added to blockchain parallelly.

Sidechains are not being used as a permanent side-chains in the current scenario, instead they are used a temporary blockchain that is created and aggregated for smaller purposes to deal with smaller transactions. For example, a sidechain is used to handle transactions related to a particular gaming platform and it is handled by the organization. In our case in order to keep the regional blockchain decentralized and by keeping in mind the security idea of original bitcoin implementation we have thought of using PoW as the consensus protocol. Now, we might think what's the difference between regional and global blockchain.

So Regional blockchain will operate with difficulty bit lesser than the global blockchain, which will increase the throughput of transactions and make the regional blockchain faster. To prove this we have simulated our proposed idea on erlang and provided the supporting data for our claim. Major advantages of have regional layer different from global layer are as follows:

- Scalability: Since the workload can be distributed across multiple regional blockchains, the chances of bottlenecks can be reduced, and this improves the overall performance of the system. Therefore, the overall system can handle a larger volume of transactions. Although currently we have proposed Proof-of-Work consensus mechanism, we future we can implement other consensus mechanisms to further improve the scalability of the system.
- Enhanced security: With multiple regional blockchains working parallelly, having decentralized approach increases the system's resistance to various attacks. Even if one regional chain is compromised, others are not being affected and the entire block isn't affected. Since our current proposal uses Proof-of-Work, the system has fault tolerance of 49%.
- Localized efficiency: Regional blockchains can be designed to address specific needs and regulations of the region. This allows for better compliance with local laws and a more efficient system that can adapt to a particular region. Transactions within a region can be processed more quickly and with lower transaction fees. This will help both the users and businesses that uses cryptocurrency.

B. Cryptocurrency Transaction rate vs Visa Transaction rate

With the growth and rise of blockchain and cryptocurrency, the global payments industry has evolved considerably. However, the biggest competition for cryptocurrencies is well-established payment technologies like Visa, MasterCard, etc. For our comparison, we will consider Visa. Since Visa is a centralized system, it can handle up to 24000 transactions per second as compared to decentralized blockchain networks like Bitcoin and Ethereum. Although the actual capacity is 24,000 transactions per second, the average Visa transaction rate is 1700 transactions per second.

Bitcoin adds 6 blocks to the blockchain every hour (1 block per 10 mins) on average. Each block is limited to 1 MB but can store over 2000 transactions. This means that Bitcoin can process around 4.6 tps. Visa can handle 24,000 tps, while Bitcoin can only process a maximum of 7 tps. Ethereum can handle 20 tps due to its smart contracts.

C. Motivation for our proposal

Visa has been a leading player in the global payments industry for decades, processing millions of transactions every day. It has long been observed that domestic Visa transactions are more than international transactions. The exact ratio is difficult to determine as it depends on various factors such as the time of year, global economic conditions, and travel restrictions. Since we can observe most customers use their cards primarily for everyday purchases in the immediate vicinity. Therefore, we can assume that domestic Visa transactions are at least 3-4 times higher than international Visa transactions.

Considering the above assumption, we have proposed this idea where we will implement a 2-Layer Blockchain. The first Blockchain will be the parent blockchain (Global blockchain). The second layer will have multiple blockchains for different regions for faster transactions. Finally, the second layer transactions will get rolled up to the first layer blockchain.

D. Theoretical Implementation

Implementation for this will require the following steps:

- Design and develop a two-tier transaction system that balances the trade-offs between computation time, number of transactions, and incentives.
- First Layer implementation will be regular blockchain implementation with PoW consensus protocol. Additionally, we will have to check how peer coin uses coinage to reduce the difficulty of the puzzle and implement similar functionality where reduction can be done using reputation points.
- Second Layer implementation will also be regular blockchain implementation with PoW consensus protocol but will have lesser difficulty than layer 1. The Incentives for validating blocks on this layer will be reputation points. Reputation points will be stored along with the

user data and can be verified by the number of blocks signed on the regional layer.
- Integrating the 2 layers will be the toughest part. We will perform rolls up on the global layer by summarizing all the transactions from regional blockchains. This will be done periodically. These rollups will improve the scalability of our system. Apart from the regular blockchain validation tasks, miners will have to validate that the total amount in our blockchain system remains legal.
- After Implementation of this system it needs to be thoroughly tested and evaluated to identify any potential weaknesses or limitations.
- The traditional rollups in blockchain generally provide the summary of all the transactions that occur in the side-chain that are going to be included in the block. Our approach however is including a simplification step where the total number of transactions are going to be minimized based on algorithm in Algorithm 2. A common Maximum Flow Algorithm used is Ford-Fulkerson Algorithm.

E. Implementation of the 2 layers

Our idea here is to have two layers of blockchain so that users can have a regional blockchain where people from the region can transact at a faster speed. These transactions will be rolled up to the global blockchain so that the total cryptocurrency of the network remains fixed. The basic structure of our proposal can be seen in the fig. 1.

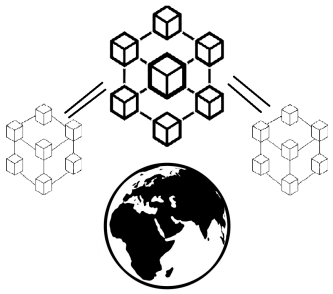

Fig. 1. Global Blockchain with Regional chains

For this, the user will create two different accounts on the blockchain. One will be the mandatory account on the Global blockchain and the second will be the optional account on the blockchain of this preferred region. This structure will be similar to Savings and Current accounts in the banking system. The user can transfer bitcoins from the global blockchain to the regional blockchain easily and vice versa. This will ensure that the total number of cryptocurrencies of a user remains fixed and users can make faster regional transactions seamlessly.

The current difficulty of bitcoin is 76 which means it has 76 leading zeros (difficulty bits). For the global chain, we will have the same difficulty as it is in the current bitcoin. For our regional purpose, we will have difficulty bits set to 74 so that mining becomes easier and thereby increases the speed of adding blocks to the chain. As discussed earlier, every reduction in the difficulty bit increases the transaction rate of the blockchain considerably. We can say that this implementation will be faster and can have more transactions per second. Rolling this data up to the Global blockchain will increase the transaction limit of our blockchain by many folds.

F. Consensus Protocols used in each layer

To ensure the integrity of this Blockchain structure, we have to define a consensus protocol such that the blockchain is secure, scalable, and decentralized. Since our proposal has two layers, we must implement 2 consensus protocols.

1) Layer 2 Consensus Protocol: This Layer is the regional layer, where we expect the blockchain to add blocks at faster rates without affecting the security of the blockchain. For our implementation, we are going to use the Proof of Work consensus protocol on this layer. The Difficulty of this PoW will be lesser than the difficulty of the Parent/Global blockchain. Since we are using PoW and the number of regional transactions will be more, we can afford to add new blocks at a faster rate (more than 1 block per 10 mins). For our theoretical implementation, we have thought of keeping the difficulty bit which is 2 lesser than the parent chain. This will increase the probability of finding the valid block by 4 times. As stated earlier, we can approximately say the ratio of domestic transactions to international transactions is 3:1 or 4:1.

The main question that arises here is what will incentivize the miners to work on this blockchain. How should we incentivize the miners? If we provide bitcoins to validate this blockchain network, that will liquidate the bitcoin value as we will have to introduce more bitcoins to the blockchain system. Hence to avoid this, we have introduced the concept of giving "reputation points" to the miners on this blockchain. The use of these points will be explained in the Layer 1 Consensus protocol.

2) Layer 1 Consensus Protocol: This Layer is the main/global layer, where all the regional blockchains will be rolled up and transactions on the international level can be added. This Layer works the same way as the bitcoin blockchain with small tweaks. This layer has been inspired by the "Peer-Coin" Implementation. In Peer-Coin, users get "coinage" based on the number of crypto coins they have had for the past 90 days. This coinage can be used by these miners to enter the global chain for mining. Where the miners with reputation greater than 5 (that they have earned on regional chains) will be consumed. By mining on this global chain, miners can earn actual coins just like the incentives in bitcoin. After consumption of the reputation points, the miners will then have to enter the regional chain again to earn the reputation points. This overview can be seen in 2.

G. Theoretical Proof of Why Regional Blockchain is faster

As Hash function provides random values, our proof that the regional blockchain is faster than the global blockchain is based on probabilities.

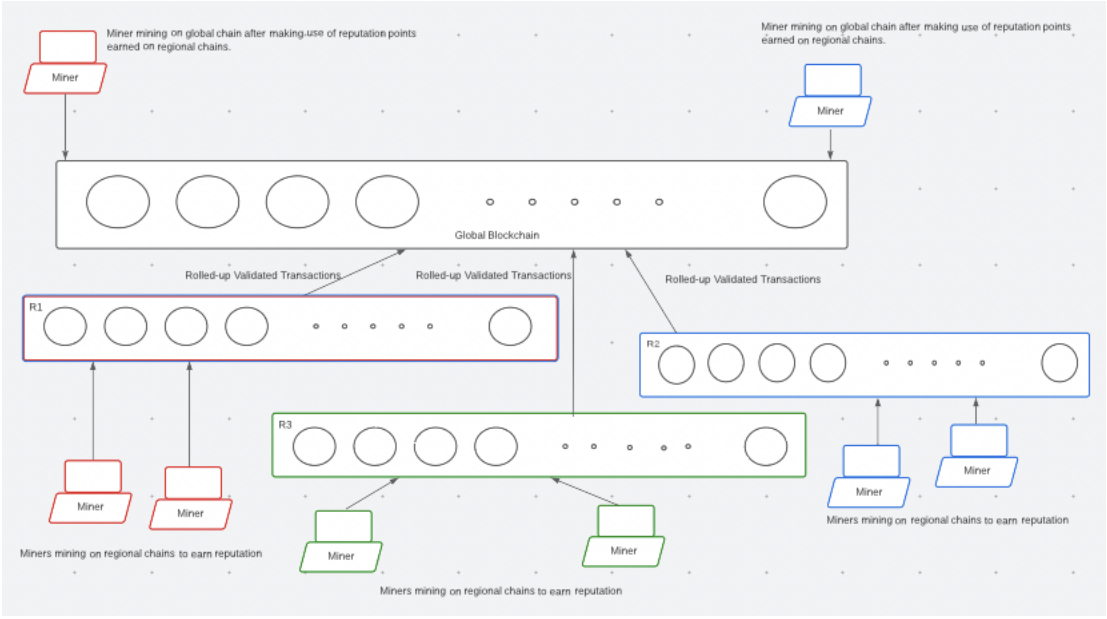

Fig. 2. Overview of our solution

Let x be the difficulty bits of the bitcoin.

Therefore, Probability P of finding a valid block $= \frac{2^x}{2^{256}}$

Max time required to find valid block (t_{global}) can be found using the formula below

$$\frac{2^{256}}{2^x \cdot \textit{computation power}} \tag{1}$$

For regional transaction validation, difficulty bits = $x - 2$.
Max time required to find valid block ($t_{regional}$) can be found by substituting the difficulty bits as:

$$\frac{2^{256}}{2^{x-2} \cdot \textit{computation power}} \tag{2}$$

Therefore, we divide Equation 1 by Equation 2

$$\frac{t_{global}}{t_{regional}} = 2^2 \tag{3}$$

$$\implies t_{global} = 4 \cdot t_{regional}$$

Since TPS (Transaction Per Second) is inversely proportional to time,

$$TPS_{regional} = 4 \cdot TPS_{global}$$

We have 4 Regional chains in our proposal and 1 Global chain.

$$TPS_{OurProposal} = 4 \cdot 4 \cdot TPS_{global} + TPS_{global}$$

$$TPS_{OurProposal} = 17 \cdot TPS_{global}$$

Therefore, Our proposal increases the throughput by 17 times.

H. Incentivization

Our solution incentivizes the miners with reputation points for mining in the regional blockchain. This encourages miners to mine on the regional chains more so that they can make use of the reputation points for mining on the global chain.

The miners can make use of the reputation points to mine blocks in global chain to earn incentives for mining on global chain just like earning bitcoins when they mine on bitcoin blockchain.

Algorithm 1 gives an overview of how the reputation points can be made used of. We have considered the *reputation_threshold* to be 5.

Algorithm 1 Reputation Algorithm

```
for i ← 1 → num_iterations do
    user ← random_user
    if user.reputation ≥ reputation.threshold then
        block.difficult −= 1
        user.reputation = 0
    end if
    Blockchain.append(Block;
end for
```

I. How the two chains work together

Since we have both the global chain and the regional chain being active in our solution and the regional chain also being

capable of handling independent transactions in its own region there might be problems that can be caused by the volume of transactions.

The transactions that are performed in the regional chain are pushed into the global chain. But, we know that the concept of rollups is used to create a summary of the transactions that were performed in the regional chain from the last checkpoint till the present point. Algorithm 2 is an example as to how the transactions can be minimized and summarized so as to reduce the volume of transactions and also make the summary smaller than the actual size of transactions.

Algorithm 2 Rollup Optimization Algorithm

```
Graph G(V, E) → All nodes and transactions as a directed graph.
Input ← G
for unvisited edge (u, v) in G do
    max_flow ← MaxFlowAlgorithm on (u,v)
    Compute Residual Graph G'
    if max_flow > 0 then
        Add (u, v) to G' with weight
    end if
    Input ← G'
end for
```

IV. Implementation

In this section, we present the implementation of the proposed blockchain system with a reputation-based incentive mechanism. Our implementation is written in Erlang, a functional programming language well-suited for distributed systems and concurrent processes.

TABLE I
Experiment configuration

Parameter	Value
Number of transactions	10,000,000
Block size	1, 8, 16, 32 MB
Transactions per block	500
Network bandwidth	20 Mbps
Number of regional blockchains	1, 4
Transactions rate (tps)	2000, 3000, 4000, 5000, 6000
Blockchain types	Bitcoin, Proposed Hybrid

The basic architecture includes a main blockchain representing the global blockchain and side chains representing the regional blockchains. The experimental configuration is given in table I for reference.

A. Nodes

Each user is represented by a node. A node can be either in one of the regional blockchains' networks or in the global blockchain's network. Each user or node initially starts off in the global blockchain and later on proceeds to the respective regional blockchain.

B. Global Blockchain

1) Node Addition: A node is added in the global blockchain at the start. It is later on transfered to the regional blockchain based on their region. All nodes are initialized and new nodes are added based on the chord algorithm.

2) Node Removal: A node is randomly removed after every 10 transactions. Before being removed, the node broadcasts its termination to its neighbors to inform them about it. Once the neighboring nodes receive the message, they update their tables holding information about their neighbors called the fingers table.

3) Node Transfer to Regional: A node is transferred from global to regional chain if it reaches the required amount of reputation. After which it starts to operate in the regional blockchain. Each node is assigned to a regional chain when added to the global blockchain. The concept is basically to replicate a user based on their IP which is indicative of its region. For the sake of simplicity, each node's region information is stored with the supervisor when the node is initialized. Ideally, each node would be assigned to its region based on its IP or CIDR (Classless Inter-Domain Routing).

C. Regional Blockchains

The regional blockchain ideally have nodes operating in them which also have a presence in global chain. The miners are allowed to mine here only if they have contributed to the global blockchain prior and have built their reputation already making them "deserving" to be trusted more. This is intuitive similar to human interactions. Based on the past record of people, they are considered reputable and the reputable person is trusted more with sensitive tasks. In our scenario, the difficulty for PoW is reduced in the regional chain leading to a vulnerability which can be exploited by a small amount of greedy nodes. The aim is to achieve greater overall TPS(Transaction Processing Speed) at the cost of security but the addition of reputation based reward and responsibility mechanism, it becomes unlikely that the reputable nodes would leave the incentive being offered for behaving in a non-malicious manner and perform an attack or exploit the system. This is similar to how humans trust each other and here the reputation threshold is currently a static value. We might experiment with a dynamically adapting reputation threshold later on.

1) Node Addition: Nodes are added in the respective regional blockchain from global blockchain based on the node's region. In implementation, each miner is awarded one reputation point if it finds a hash that meets the difficulty requirement. Each miner accumulates reputation until it reaches a threshold. The miner then allowed to to operate in the global chain where it is incentivized with actual coins. Before, a node is moved to the regional chain, its current reputation points are reset to zero. The node then begins to operate in the regional chain. Regional blockchain being a geo-based Layer 2 solution enhances blockchain scalability by partitioning the network into localized clusters based on geographic locations.

Each cluster operates independently, processing and validating transactions within its region.

For transactions between regions, the regional committee communicate with each other and relay transaction details. The destination cluster processes the transaction using its hybrid reputation-based consensus mechanism. Once validated, the transaction is added to both the originating and destination clusters' blockchains.

2) Node Removal: Nodes are not currently being removed form the regional blockchains.

3) Node Transfer to Global: Nodes work in the regional blockchain and are miners rewarded reputation points which were earlier reset to zero. They gain points similar to how they would earn coins in the global chain. Once the reputation points reach a threshold, the nodes are moved back to the global blockchain and rewarded coins.

D. Consensus - Proof of Work

The proof of work consensus protocol is used in both global and regional chains along with the reputation score modification. The PoW algorithm is designed to find a hash that meets a specific requirement. The requirement is determined by the number of leading zeros needed in the hash. Our PoW algorithm is based on the widely-used SHA-256 cryptographic hash function. The algorithm searches for a hash value that meets a specific requirement, which is determined by the number of leading zeros in the hash. This requirement is commonly referred to as the target, and the process of searching for a hash that meets the target is called mining.

Algorithm 3 Proof-of-Work Algorithm

```
String ← Transaction
while True do
    Hash ← SHA-256(String)
    if leading_zeros ≥ Target then
        return (String, Hash)
    end if
    String ← Hash
end while
```

In our implementation, the mining process begins with a random input string which represents a transaction. The algorithm iteratively hashes the input string and checks if the resulting hash meets the target requirement. If the requirement is met, the algorithm returns the input string and the valid hash. If the requirement is not met, the algorithm updates the input string with the new hash and repeats the process until a valid hash is found.

The algorithm above takes two arguments: RPIDMap, a map containing RPID keys and their associated counts, and RPID, the key to be updated. The function starts by setting the variable Reputation to 5. It then attempts to find the RPID key in the RPIDMap. If the RPID is found, the function checks if the incremented count is greater than or equal to the Reputation. If it is, the RPID value is set to 0 in the RPIDMap. Otherwise, the RPID value is incremented by 1. In both cases, the updated RPIDMap is returned. If the RPID is not found, the function inserts the RPID into the RPIDMap with a value of 1 and returns the updated RPIDMap. The basic concept is to store the reputation of neighboring nodes and to update them each time a node finds the correct hash.

Algorithm 4 Reputation-based mining algorithm

```
Input ← RPIDMap, RPID
Reputation ← 5
Count ← Call → {maps.find}(RPID, RPIDMap)
if Count = OK then
    NewRPIDMap ←
    if Count + 1 ≥ Reputation then
        Call → {maps.put}(RPID, 0, RPIDMap)
    else
        Call → {maps.put}(RPID, Count + 1, RPIDMap)
    end if
else
    NewRPIDMap ← Call → {maps.put}(RPID, 1, RPIDMap)
end if
Return NewRPIDMap
```

In the current implementation, we use the server to manage the blockchains. To store and update the reputation of each node when they find a valid block. Secondly, they also instruct a node to move to regional or global blockchain. They also inform their neighbors and the node to update their respective finger tables according to the chord algorithm. This was done only to maintain the simplicity of the implementation and can be implemented in practice to maintain decentralization at the cost of simplicity.

E. Communication Protocol - Chord

In this section, we present the implementation of the Chord algorithm and discuss its primary functionalities. The Chord algorithm is a distributed hash table (DHT) protocol that enables efficient lookup and storage of key-value pairs in a scalable and fault-tolerant manner. It is particularly suitable for large-scale distributed systems, where nodes may join or leave the network frequently.

1) Node Representation and Identifier Space: The Chord algorithm represents each node using a unique identifier (ID) in a circular ID space. The ID space ranges from 0 to $2^m - 1$, where m is the number of bits in the node ID. Each node is assigned a unique ID through a consistent hashing function, such as SHA-1.

2) Key-Value Pair Distribution: Chord assigns each key-value pair to the node with the smallest ID equal to or greater than the key. If no such node exists, the key-value pair is assigned to the first node in the ID space. This process ensures an even distribution of keys across the network.

3) Lookup: The Chord algorithm provides an efficient lookup procedure to locate the node responsible for a specific key. Each node maintains a finger table with information about other nodes in the system. The finger table contains

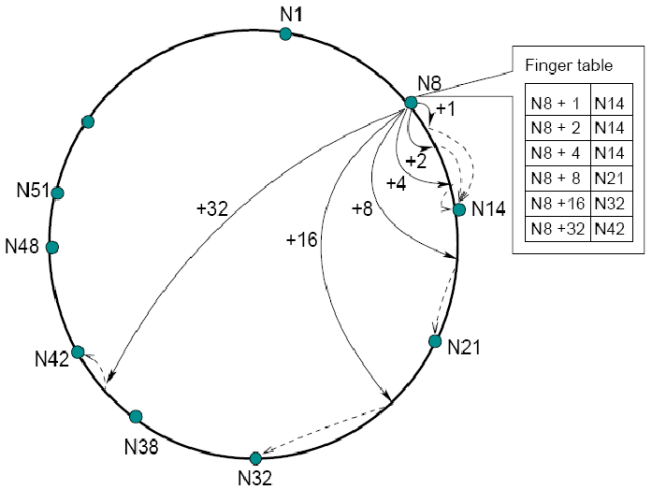

Fig. 3. Chord Communication Protocol

m entries, each pointing to a node 2^{i-1} positions ahead of the current node in the ID space, where i = 1, 2, ..., m. This structure enables nodes to forward lookup queries logarithmically, resulting in O(log N) lookup time, where N is the number of nodes.

4) Joining the Network: When a new node joins the Chord network, it initializes its finger table based on the information from an existing node. It then updates the finger tables of other nodes and transfers the appropriate key-value pairs to itself. This process ensures that the network remains consistent and balanced as nodes join.

5) Leaving the Network: When a node leaves the Chord network, it transfers its key-value pairs to its immediate successor and notifies its predecessor to update its successor pointer. Additionally, it informs nodes with finger table entries pointing to itself to update their finger tables. This process ensures that the network remains consistent and balanced as nodes leave.

6) Stabilization and Fault-Tolerance: The Chord algorithm periodically runs a stabilization protocol to update the network's state and ensure fault tolerance. The stabilization protocol updates each node's successor pointer, verifies and corrects finger table entries, and replicates key-value pairs to provide redundancy. This process enables the Chord network to recover from node failures and maintain consistency.

F. Block Validation

During the block validation phase, a group of validators is selected based on their reputation scores. The higher the reputation score, the higher the chance of being selected as a validator. Validators verify the transactions in the proposed block and come to a consensus on the block's validity.

G. Roll-ups

The nodes also periodically roll up transactions from the localized clusters to the main blockchain. This process further enhances the overall scalability and efficiency of the network while ensuring data consistency and security. This is done so that the global blockchain maintains consistency across the network. They exchange block headers, and any discrepancies are resolved using the hybrid reputation-based consensus mechanism. The nodes collect and aggregate transaction data from their local blockchains. A cryptographic hash representing the aggregated transactions, is then generated. This process compresses the transaction data, enabling more efficient storage and verification on the main blockchain. The hash is incorporated into a new block and added to the main blockchain. By periodically rolling up transactions from the geo-based Layer 2 clusters to the main blockchain, we can effectively reduce the storage and computational requirements on the main chain.
[1]

Algorithm 5 Chord Algorithms

Lookup*(Key)*
Find node responsible for the key using finger table.
Return responsible_node

Join*(node)*
Initialize node's finger table using an existing node
Update finger tables of other nodes
Transfer appropriate key-value pairs to the joining node

Leave*(node)*
Transfer key-value pairs to the node's immediate successor
Notify predecessor to update successor pointer
Update finger tables of other nodes

Stabilization*()*
while *Network is active* **do**
  Update successor pointers and finger tables
  Verify and correct finger table entries
  Replicate key-value pairs for fault-tolerance
  Sleep for a certain time
end while

V. Results and Analysis

A. Analysis

This following data shows how our solution demonstrates considerable improvements in TPS and reduced block time compared to existing systems. The analysis section provides an overview of our results. The table III compares our proposed solution and existing blockchain networks across various difficulty levels (bits). For each difficulty level, the table shows our solution's TPS, existing TPS, our solution's block time, and existing block time.

Our analysis reveals that our solution consistently outperforms existing systems in terms of TPS and block time across all difficulty levels. For example, at the lowest difficulty (10 bits), our solution achieves a TPS of 621,599.38, almost 10

[1]Project Code: https://github.com/Ammar-Amjad/ReputationBlockchain

TABLE II
BLOCK TIME WHILE CHANGING NUMBER OF NODES AND DATA SIZES

Block Size	Number of Nodes			
	1	2	4	8
1 MB	1.53	1.24	1.15	1.14
8 MB	3.31	2.66	2.59	2.32
16 MB	4.45	3.75	4.23	3.43
32 MB	6.15	5.59	4.99	5.07

times higher than the existing TPS of 62,348.44. At the highest difficulty level (21 bits), our solution maintains a TPS of 638.57, over 12 times higher than the existing TPS of 49.16.

Furthermore, our solution significantly reduces block time across all difficulty levels. At the lowest difficulty (10 bits), the block time of our solution is around 0.000804377 seconds, approximately 90% faster than the existing block time of 0.008019447 seconds. At the highest difficulty level (21 bits), our solution achieves a block time of 0.782994699 seconds, about 92% faster than the existing block time of 10.17006874 seconds.

The table II is a comparison of time taken to find valid block with different number of nodes and changing size of block. As the block size increases, the time taken to find a valid block also increases. This is evident from the increasing values along each row. For example, with 1.53 nodes, the time taken increases from 1 MB (1.53) to 8 MB (3.31) and further to 32 MB (6.15). The data suggests that increasing the number of nodes and decreasing the block size can help improve the efficiency of finding valid blocks. This relationship could have implications for optimizing performance in blockchain networks or other systems that rely on finding valid blocks.

In conclusion, our research highlights the potential of our novel approach to enhance blockchain performance. The results showcase substantial improvements in TPS and block time across various difficulty levels. Future work will focus on refining our solution and exploring its real-world applications, aiming to promote scalable and efficient blockchain systems.

B. Comparison with Bitcoin Transaction rate

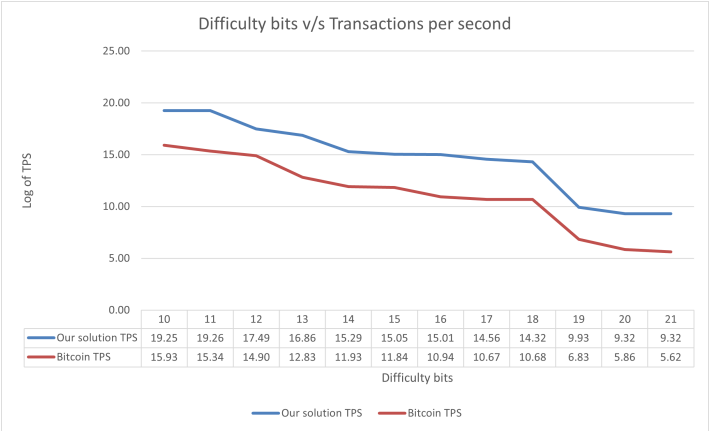

Fig. 4. TPS

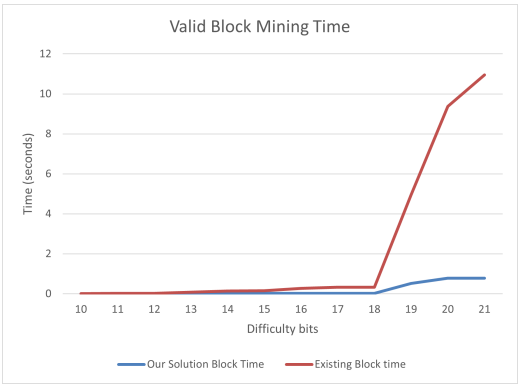

Fig. 5. Block Mining Time

1) Comparison with Bitcoin Blockchain: As mentioned earlier, currently block chain has 76 leading zeros. Which takes around 10 minutes to validate a block. A comparison between Bitcoin blockchain and our hybrid blockchain can be seen in table III. The comparison can also be seen in figure 4 which shows a pictoral comparsion transactions per second(TPS) achieved. Its evident that our solution outperforms bitcoin blockchian by performing significantly more transactions per second. Thereby, achieving greater scalability. Following are the scalability advantages that our proposed idea provides over the bitcoin blockchain. Furthermore, from figure 5, we observe that our solution

2) From Blockchain Layer-1: In our solution, miners with reputation points can redeem it to reduce the difficulty of the puzzle by half. Therefore, Increasing the probability of finding the valid block by twice. This will motivate the miners to mine on regional blockchain and gain reputation points as shown in algorithm 1. This will improve the time taken to find the valid block. Hereby Increasing the scalability of the blockchain by a small value.

3) From Blockchain Layer-2: In our implementation, we will be having lower difficulty leading zeros for this layer. This will increase the rate of adding blocks to the System which will improve the time taken to find the valid block. Even this will considerably improve the scalability of the Blockchain.

4) From Integration of layers: When we are merging data from regional blockchains we plan to implement rollups so that the scalability of the Systems increases many folds.

The second layer is secure due to local blockchain implementation and faster due to less network overhead for communication. It's also worth mentioning that there is a massive throughput increase expected simply because of this reason.

5) Comparison with Sharding: With sharding, not all nodes validate the transactions which leads to centralization. Our solution has a layer 2 blockchain with Proof-of-Work (PoW) consensus protocol, which works in the same way as the regular blockchain thereby making our system decentralized.

The major flaw of Sharding is that the Security of the network might be compromised as it makes a single-shard

TABLE III
COMPARISON OF OUR SOLUTION AND EXISTING SOLUTION

Difficulty Bits	Our Solution TPS	Existing TPS	Our Solution Block Time	Existing Block Time
10	621599.38	62348.44	0.0008	0.0080
11	627965.03	41603.55	0.0008	0.0120
12	184631.46	30512.02	0.0027	0.0164
13	119087.35	7288.36	0.0042	0.0686
14	40003.36	3903.36	0.0125	0.1281
15	33968.47	3673.54	0.0147	0.1361
16	33052.67	1966.01	0.0151	0.2543
17	24112.79	1634.07	0.0207	0.3060
18	20406.74	1641.74	0.0245	0.3046
19	976.79	113.79	0.5119	4.4390
20	638.21	58.17	0.7834	8.5952
21	638.57	49.16	0.7830	10.1701

takeover attack (1% attack) possible. It's worth mentioning that the 1% attack is an attack to take control over 1% of the network to take over the shard. Instead for our case 51% control of the network is needed to perform the 51% attack. This makes our approach more robust and secure.

Another flaw of sharding is that data fragmentation makes it cumbersome to access historic data and trace the history of assets for data validation. Our solution avoids this con of sharding by avoiding fragmentation altogether.

6) Comparison with Side-chains: Lightning Network and Payment Channels which aim to increase transaction rate at the **expense of security**. Our solution has a second-level blockchain setup that is governed and validated by the miner. So, our solution is more secure than traditional sidechains.

The lightning network works best in a one-to-one transaction as people on both ends of the channel trust each other. Additionally, a third party is needed to keep the channels open for payment, leading to centralization. Our solution has a layer 2 blockchain, which works in the same way as the regular blockchain thereby making our system **decentralized**.

7) Comparison with Rollups: In rollups, aggregators bundle together transactions and send them to the blockchain but since aggregators are few, this makes the system dependent upon aggregators which can cause **centralization**. In our approach, every node validates the blocks thereby remaining decentralized.

Unlike optimistic rollups which take 1-2 weeks for aggregation, after which a user has access to his assets. Our hybrid solution performs a roll-up every day in the regional chains so changes will be reflected quickly.

Its also worth mentioning that due to our hybrid reputation based approach, the reduction in difficulty and therefore time to find valid block, we end up having less energy consumption compared to other approaches which need PoW including sharding, sidechains, etc. Our solution has a second-level blockchain setup that is governed and validated by the miner. These miners are incentivized to validate the block by awarding reputation points. Therefore, another system is more secure and supports decentralization. Our proposed hybrid solution achieves comparable speed and scalability without compromising on security, as states are not transferred but a blockchain is maintained at the regional level, ensuring trust and trust in every transaction at the regional level.

This helps our system to increase the scalability without significant effect on the security and decentralization of the blockchain

VI. CONCLUSION

The proposed solution offers some advantages over the existing blockchain network.

1) Improved Scalability: By having a global blockchain and multiple region-based blockchains, the load on the global blockchain would be reduced as the number of transactions processed on the global chain would be lesser in number.
2) Enhanced Security and Consistency: Since the global blockchain stores all the details such as transaction histories, miners' reputations, etc. ensuring the network's integrity. All the region-based chains would be securely linked to the parent chain allowing the network to remain secure and consistent.
3) Flexibility: This proposal is adaptable to future modifications for further optimization using any of the layer-2 solution techniques like sharding, rollups, state channels, etc.
4) Incentive Mechanism: Our solution encourages the miners to participate in the network by providing them with score-based reputation incentivization. The miners with a higher reputation would be able to solve the parent chain's PoW (Proof-of-Work) puzzle with reduced difficulty.
5) Geographic Distribution: The region-based chain takes into account the geographical location of the user allowing for faster transaction processing in localized areas.
6) Reduced Network Congestion: By offloading most local transactions to the region-based chains, this solution alleviates the congestion that is a problem in a parent blockchain which is a problem in the existing blockchain implementation.

Conclusively, the blockchain trilemma is a major impediment to the widespread adoption of blockchain solutions. This paper introduces a novel way to circumvent the trilemma, particularly focusing on the scalability limitation. The merger of 2 layer solution with reputation points incentive and geolocation chains is expected to push the boundaries of blockchain research and open new avenues for further improvements.

REFERENCES

[1] Hafid, Abdelatif, Abdelhakim Senhaji Hafid, and Mustapha Samih. "Scaling blockchains: A comprehensive survey." IEEE Access 8 (2020): 125244-125262.

[2] Gudgeon, L., Moreno-Sanchez, P., Roos, S., McCorry, P., & Gervais, A. (2020). Sok: Layer-two blockchain protocols. In Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24 (pp. 201-226). Springer International Publishing.

[3] A. Chauhan, O. P. Malviya, M. Verma and T. S. Mor, "Blockchain and Scalability," 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), Lisbon, Portugal, 2018, pp. 122-128, doi: 10.1109/QRS-C.2018.00034. New York: Academic, 1963, pp. 271–350.

[4] X. Sun, F. R. Yu, P. Zhang, Z. Sun, W. Xie and X. Peng, "A Survey on Zero-Knowledge Proof in Blockchain," in IEEE Network, vol. 35, no. 4, pp. 198-205, July/August 2021, doi: 10.1109/MNET.011.2000473.

[5] L. T. Thibault, T. Sarry and A. S. Hafid, "Blockchain Scaling Using Rollups: A Comprehensive Survey," in IEEE Access, vol. 10, pp. 93039-93054, 2022, doi: 10.1109/ACCESS.2022.3200051.

[6] King, S. and Nadal, S., 2012. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. self-published paper, August, 19.

[7] Abdurrashid Ibrahim Sanka, Ray C.C. Cheung, A systematic review of blockchain scalability: Issues, solutions, analysis and future research, Journal of Network and Computer Applications, Volume 195, 2021, 103232, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2021.103232.

[8] N. Chaudhry and M. M. Yousaf, "Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities," 2018 12th International Conference on Open-Source Systems and Technologies (ICOSST), Lahore, Pakistan, 2018, pp. 54-63, doi: 10.1109/ICOSST.2018.8632190.

[9] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei and C. Qijun, "A review on consensus algorithm of blockchain," 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 2017, pp. 2567-2572, doi: 10.1109/SMC.2017.8123011.

[10] Sguanci, C., Spatafora, R., & Vergani, A. M. (2021). Layer 2 blockchain scaling: A survey. arXiv preprint arXiv:2107.10881.

[11] Marukhnenko, O., & Khalimov, G. (2021). The overview of decentralized systems scaling methods. COMPUTER AND INFORMATION SYSTEMS AND TECHNOLOGIES.