IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 April 2024, accepted 23 May 2024, date of publication 30 May 2024, date of current version 7 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3407521

RESEARCH ARTICLE

Preventing 51% Attack by Using Consecutive Block Limits in Bitcoin

SOHAIL MAHMOOD BABUR[1], SHAFIQ UR REHMAN KHAN[2], JING YANG[3], (Graduate Student Member, IEEE), YEN-LIN CHEN[4], (Senior Member, IEEE), CHIN SOON KU[5], AND LIP YEE POR[3], (Senior Member, IEEE)

[1]Department of Information Technology, Government Murray College, Sialkot 51010, Pakistan
[2]Department of Computer Science, Capital University of Science and Technology, Islamabad 44730, Pakistan
[3]Department of Computer System and Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[4]Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei 106344, Taiwan
[5]Department of Computer Science, Universiti Tunku Abdul Rahman, Kampar 31900, Malaysia

Corresponding authors: Yen-Lin Chen (ylchen@mail.ntut.edu.tw), Jing Yang (s2147529@siswa.um.edu.my), Lip Yee Por (porlip@um.edu.my), Chin Soon Ku (kucs@utar.edu.my), and Shafiq Ur Rehman Khan (shafiq.rehman@cust.edu.pk)

This work was supported in part by the National Science and Technology Council in Taiwan under Grant NSTC-112-2221-E-027-088-MY2 and Grant NSTC-111-2622-8-027-009; in part by the Ministry of Education of Taiwan Entitled "The Study of Artificial Intelligence and Advanced Semiconductor Manufacturing for Female STEM Talent Education and Industry-University Value-Added Cooperation Promotion" under Grant 1122302319; and in part by Universiti Tunku Abdul Rahman (UTAR), Malaysia, Financial Support for Journal Paper Publication Scheme through UTAR.

ABSTRACT In permissionless blockchain systems, Proof of Work (PoW) is utilized to address the issues of double-spending and transaction starvation. When an attacker acquires more than 50% of the hash power of the entire network, they gain the ability to engage in double-spending activities, posing a significant threat to the PoW consensus algorithm. This research focuses on the consensus algorithm employed in the Bitcoin system, explaining how it operates and the security challenges it faces. The proposed modification to the PoW algorithm imposes a restriction on miners: they are not allowed to accept consecutive blocks from the same miner into the final local blockchain to prevent the 51% attack problem. This modification supports transactions that require six confirmations. In the event an attacker attempts a 51% attack with a private chain that consists of fewer than 6 blocks, it becomes easier to detect a double-spending attack before accepting the attacker's private chain. The modified algorithm introduces a "Safe Mode Detection Algorithm" that scrutinizes incoming blocks for adjustments at the top of the local blockchain. If inconsistencies are identified, the consensus algorithm proceeds cautiously by comparing the UTXO dictionaries from the attacker's chain with those from the miner's own blockchain. This meticulous comparison aims to detect instances of double-spending. If such instances are detected, the miner rejects the attacker's chain, establishing a double-spend-free environment and thwarting 51% attacks.

INDEX TERMS 51% attack, bitcoin and consensus, blockchain, double spending, proof of work (PoW).

I. INTRODUCTION

The first decentralized public ledger system in blockchain technology is Bitcoin, which was developed by Satoshi Nakamoto in 2009 [1]. Bitcoin is a payment system where digital currency (bitcoin) can be sent or received on a distributed peer-to-peer network for trading purposes. To secure the exchange of electronic cash, cryptocurrency is used for all network mediums that employ cryptography. Anyone can install a Bitcoin application and become part of the Bitcoin peer-to-peer network. There are different versions of blockchain, but Bitcoin is based on Blockchain 1.0, designed specifically for digital cryptocurrencies [2].

The Bitcoin network must adhere to rules of ownership, with its key elements outlined as follows [3]:

- How can we identify an owner?
- Can we identify digital currency?
- How can we map the owner and digital currency?

The associate editor coordinating the review of this manuscript and approving it for publication was S. K. Hafizul Islam.

The Bitcoin peer-to-peer network relies on principles of public-key cryptography, digital signatures, and blockchain technology to address these key elements. To function effectively in a purely distributed peer-to-peer network environment, the Bitcoin system must perform the following major tasks:

- How can we describe and protect ownership?
- How can we store transaction data?
- How can we prepare ledgers and distribute them throughout the network?
- How can we add new transactions to the ledgers?
- Which ledgers are considered valid?

"Blockchain.info" is one of the popular blockchain explorers. Double spending is a major problem in digital currency and can be mitigated by the Bitcoin Payment System through the use of a public ledger known as the blockchain.

Another critical aspect of a successful network is maintaining the data integrity of the system. Blockchain achieves this goal through the use of the cryptographic hash function SHA-256 [4]. The blockchain system serves as the lifeblood of the cryptocurrency world, presenting a linked list of blocks containing transactions. Bitcoin transactions are efficiently stored in a hash-based Merkle tree. All operations are performed efficiently on the Merkle tree [5]. Within each block, the first transaction is coin-based, serving as the reward for the winning miner. It's important to note that coin-based transactions have no inputs. The leaf nodes of the Merkle tree represent transactions that are recursively hashed until the Merkle root is obtained. Figure 1 illustrates how the Merkle tree is incorporated into the blockchain.

TABLE 1. Acronyms used in this paper.

Acronym	Explanation
ADT	Advanced digital technology
VPI	Virtual path identifier
5G	Fifth generation of wireless mobile communication
BDLT	Blockchain distributed ledger technology
DLT	Distributed ledger technology
AI	Artificial Intelligence
IoT	Internet of Things
ML	Machine learning

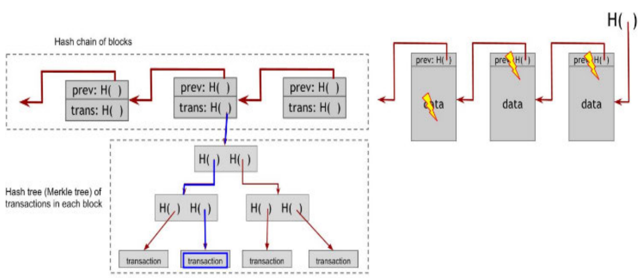


FIGURE 1. Blockchain with merkle tree.

Wallet software generates new transactions that select unspent transaction outputs (UTXO) from the UTXO set, referred to as "Inputs," and constructs new outputs based on the new owner. These transactions are then sent to neighboring nodes and propagate through the network. Invalid transactions are promptly discarded. The sum of all outputs must be slightly less than the sum of all validated inputs, accounting for an implied transaction fee collected by the miner responsible for adding the transaction to the open ledger. Unlike currency notes, bitcoin chunks in transactions cannot be divided and are locked by the owner. Once transactions are validated, they become part of a transaction pool.

Miner nodes are specialized computer hardware systems connected to full Bitcoin nodes. Their primary responsibility is to receive unconfirmed transactions propagating on the network [6]. The transactions that go unselected by a miner for an extended period are eventually discarded [7]. Miners maintain a local copy of the blockchain and are responsible for validating transactions, placing them into a block, and adding them to a globally distributed ledger. The process of adding a new block to the global distributed ledger averages around 10 minutes. Miners can receive two types of incentives:

- Creating new coins as a reward.
- Earning transaction fees.

To obtain these incentives, miners employ a consensus mechanism before adding transactions to the blockchain. A consensus mechanism is a technique employed to achieve the following goals within the Bitcoin network:

- Agreement
- Trust
- Security

The most prevalent consensus mechanisms in the cryptocurrency world include Proof of Work (PoW), Proof of Stake (PoS), and Delegated Proof of Stake (DPoS) [8].

The first challenge is to identify the nodes, how they assert ownership of their digital money, and how they can transfer this ownership to another node. Asymmetric public-key cryptography is employed to accomplish this, generating a pair of keys (public and private). The public key serves for node identification and is publicly disclosed, while the private key is utilized for ownership, specifically to produce a digital signature [6].

Another vital aspect of a successful network is the preservation of data integrity. Blockchain systems achieve this objective by utilizing cryptographic hash functions such as SHA-256. These cryptographic hash functions possess essential properties, including:

- Rapid generation of hash codes and hash values for any type of data.
- Consistent production of the same hash code for identical input data.
- Generation of unpredictable results for minor alterations in input data.
- Prevention of the production or prediction of input data from the output hash code.

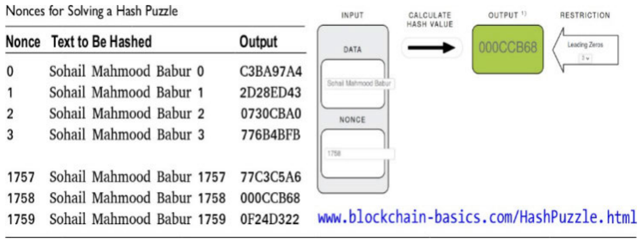


FIGURE 2. Hashing puzzle.

- Collision resistance ensures that it is difficult to find two different data structures with the same hash code.

Another use of hashing is to create hash puzzles that necessitate computational resources for resolution. It is impossible to solve these puzzles based on knowledge or stored data. The sole method to tackle these puzzles is through the consumption of computational power and effort.

In the context of blockchain technology, this computational work is known as Proof of Work (PoW) and includes elements of a hash puzzle, comprising [3]:

- The given data (which remains unchanged during the process).
- The numeric value that can be freely altered is known as a nonce.
- The application of a hash function, such as SHA-256 (as depicted in Figure 2).

The global difficulty adjusts approximately every two weeks, or after 2016 blocks, to ensure that the process of adding blocks to the blockchain maintains an average duration of 10 minutes [9]. Miners can increase their hashing power either by employing more powerful hardware or by participating in mining pools.

Mining pools represent a strategy for boosting hashing power where miners collaborate on solving a single hash puzzle to mine a block. This collaborative effort is referred to as a mining pool. If a pool member discovers the solution, the resulting reward is distributed among all pool members based on the pool's policy. It's worth noting that mining pools may levy a fee on each member [10]. Figure 3 displays some popular mining pools for the year 2021 [11].

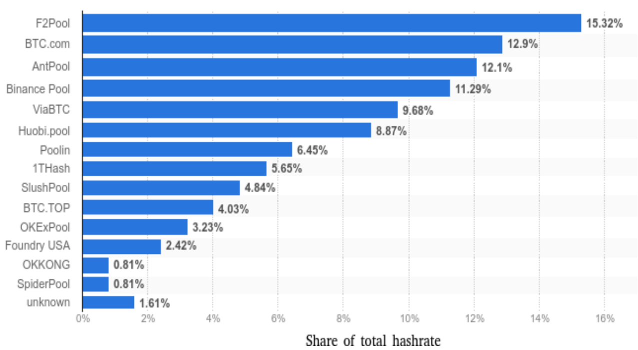


FIGURE 3. Mining pools in 2021.

An attacker with more than 50% of the hash power can initiate a new parallel chain of blocks alongside the legitimate chain, effectively isolating the genuine chain and enabling double-spending. This type of attack is a significant concern within the Bitcoin network [12]. Notably, in May 2018, Bitcoin Gold experienced a substantial loss of 18 million dollars due to such an attack [13].

This attack gives rise to two major problems:

- Transaction starvation: This occurs when certain transactions, not selected by the miner holding 51% of the hash power, are left unprocessed for extended periods. One solution to this problem is the adoption of Proof of Stake (PoS) instead of PoW.
- Double-spending: In this scenario, a single digital currency can be utilized multiple times. A miner possessing 51% or more of the network's hashing power can pre-create a chain but refrain from broadcasting it immediately. Later, the miner decides to release this extended chain in accordance with blockchain policies. If other miners accept this longer chain, the attacker gains an advantage [8].

As the Bitcoin network grows in size and complexity, the threat of a 51% attack has diminished to some extent, making it less of a concern these days [14]. However, it cannot be entirely disregarded because the Bitcoin network has indeed fallen victim to such attacks. Therefore, it is imperative that we explore ways to enhance our consensus algorithm. The 51% attack is fundamentally a double-spending attack [2].

This study aims to accomplish the following objectives:

- To investigate the effectiveness of the PoW consensus algorithm in preventing 51% attacks on the Bitcoin network.
- To examine potential modifications to the PoW consensus algorithm.
- To foster a trustworthy environment for society to use the Bitcoin network, free from the threat of 51% attacks.
- To visualize the entire attack process using a simulator and present the results in the form of graphs to enhance comprehension of the attack.

An open-source simulator named BlockSim, which emulates the Bitcoin network, will be employed for this research. It provides a platform to create miners with varying hashing powers functioning within distributed systems. While all miners initially possess equal hashing power, only the attacker will gradually increase their hashing power at regular intervals during the simulation. Different scenarios involving this simulation will be explored.

The remainder of this paper is structured as follows: Section II delves into related work; Section III outlines the proposed methodology; and Section IV presents a detailed analysis of the experiments. Finally, Section V concludes this research study and offers insights into future directions.

II. RELATED WORK

This section provides a review of the literature concerning Bitcoin, the 51% attack, and the theoretical framework of

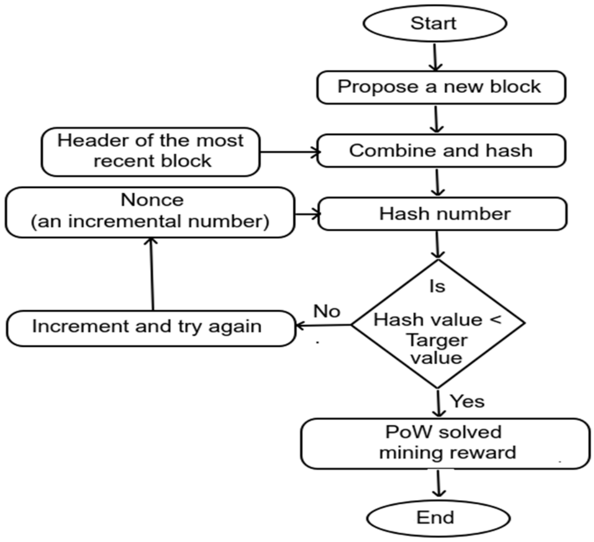


FIGURE 4. **Proof of work (PoW).**

this study. When an attacker (miner) acquires more than 50% of the network's computational power, they can create a new chain. The attacker with the longest chain can then isolate the genuine chain. The 51% attack stands as a significant contributor to the double spending problem. While some solutions, such as Delayed Proof of Work (DPoW), Historical Weighted Difficulty, and Two Phase Proof of Work (2P-PoW) in Bitcoin, as well as concepts like PirlGuard and ChainLocks, manage to address the issue to some extent, the threat remains. These solutions tend to introduce delays in network processes and also reduce transaction speed.

The theoretical framework of this article introduces and elucidates the theory that provides an explanation for the existence of the research problem. It should be noted that PoW, PoS, and DPoS do not entirely eliminate the threat of a 51% attack [8].

A. THEORETICAL FRAMEWORK

1) PROOF OF WORK (POW)

PoW involves solving a mathematical problem using a cryptographic hash algorithm. A nonce is calculated to ensure that the resulting hash value is less than the target value. The target value is adjusted to modulate the difficulty of the puzzle, making it easier or harder. The winning node adds the block to the final blockchain and broadcasts it on the network. If multiple nodes discover the solution simultaneously, temporary forks may occur. However, the protocol eventually ensures that the longest chain (with the maximum PoWs) is selected as the final blockchain, while others are excluded to maintain consistency [6]. Figure 4 illustrates the flowchart of PoW.

However, PoW, which forms the backbone of the Bitcoin network, has some drawbacks [15]:

- It consumes significant extra electricity resources, estimated at 24 terawatt-hours per year.
- In smaller networks, an attacker may gain 51% of the hash power.

The "tragedy of the commons," as discussed by authors in a paper [16], arises when the block reward becomes zero, leaving only transaction fees as profits for miners. In such a scenario, many miners might abandon mining, potentially leading to a 51% attack due to network consolidation.

Increasing hashing power can only be achieved by enlarging the pool size, thereby making a 51% attack a possibility with no effective solution [17]. Quantum devices pose another challenge by enhancing computational power using the Proof of Work (PoW) consensus algorithm within the Bitcoin network [18]. A successful 51% attack by a quantum device would enable the attacker to halt and confirm new transactions [19]. Quantum algorithms offer better time and memory complexity, leading to the possibility of a quantum 51% attack [20].

2) DELEGATED PROOF OF STAKE (DPOS)

In the DPoS algorithm, network users vote to elect delegates, often referred to as witnesses or block producers. Once elected, delegates are granted the authority to validate the blocks added to the blockchain. DPoS randomly selects a set number of delegates (typically 20 to 100) from the network before each new block is incorporated into the blockchain. Transaction fees from the new block are distributed among all delegates, while rewards are shared among users who have staked their tokens in the successful delegate's pool. Greater stakes result in a larger share [21].

3) A NEW PROOF OF WORK (POW) MECHANISM FOR BITCOIN

Miners are presented with problems to solve, but if they focus on problems with multiple solutions, their efforts can be justified by finding multiple solutions rather than just one. This approach assigns value to the work of all miners and can be appreciated [22].

4) TWO PHASE PROOF OF WORK (2P-POW) IN BITCOIN

The existence of large public mining pools significantly reduces the reward for individual miners. These pools often require pool operators to hand over private keys or a significant portion of their pools. The 2P-PoW algorithm employs continuous-time Markov chains (CTMCs), with the second difficulty (Y) acting as the inverse of the normal difficulty (X). Pool operators are compelled to cooperate by solving the second difficulty (Y) in order to access the normal difficulty. Funds from coin-based addresses can only be transferred if they successfully address the second difficulty (Y) [9]. Figure 5 displays the transition graph of 2P-PoW. Extended 2P-PoW also indicates that this change doesn't fully mitigate 51% attacks [23].

5) HISTORY WEIGHTED DIFFICULTY

Figure 6 illustrates two branches: an honest branch and an attacker's branch. Incorporating the histories of both branches assists in identifying the attacker. In this technique,

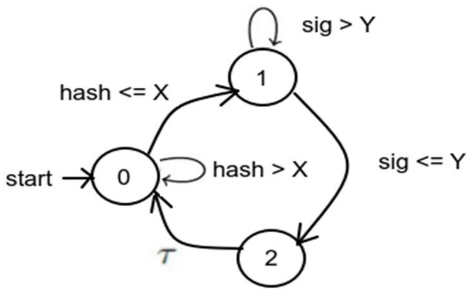


FIGURE 5. Two phase proof of work (2P-PoW).

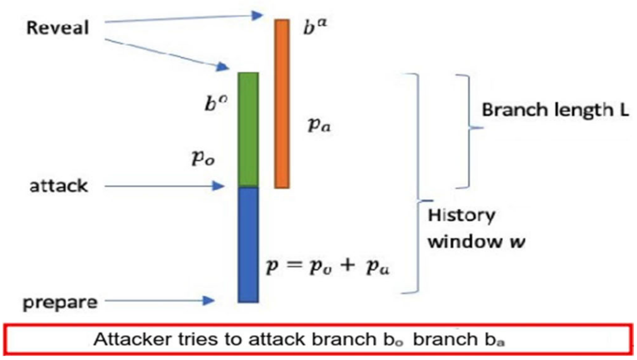


FIGURE 6. Historical weighted difficulty.

history-weighted difficulty is introduced into the difficulty calculation. While it helps in mitigating 51% attacks to some extent, the threat is not entirely eliminated from the network [24].

6) REVISITING DOUBLE SPENDING ATTACKS ON BITCOIN

This study introduces a new type of double spending attack (DSA) called adaptive DSA within the context of the Bitcoin blockchain. It also presents insights related to this attack. In the analytical model, the double spending attack is transformed into a Markov decision process. Stochastic dynamic programming (SDP) is then utilized to derive optimal attack strategies for adaptive DSA. Through this model and the insights into adaptive DSA, the study aims to highlight that the threat of double-spending attacks remains significant within the Bitcoin ecosystem [25].

7) POW BASED ON BLOCK COMPRESSION (POW-BC)

PoW-BC aims to reduce block size, improve transmission efficiency, and reduce disk space requirements for storing blocks. The block compression ratio is used to adjust mining difficulty, reduce block intervals, and minimize energy consumption. This reduction in the chances of a 50% attack is attributed to variations in the block compression ratio resulting from different transaction selections and their orders [26]. Figure 7 illustrates how PoW-BC functions.

B. OPERATIONS AND SERVICES

In the present day, most nodes do not engage in mining as individual entities. Instead, nearly every miner is part of a

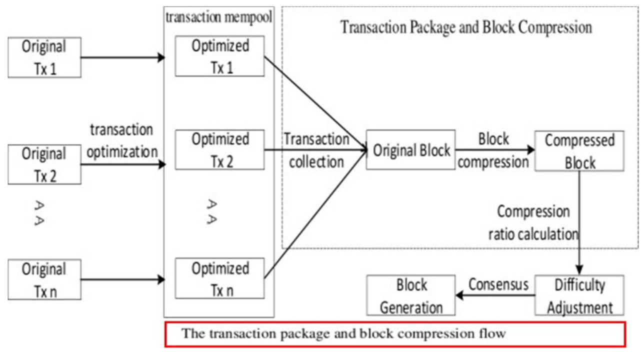


FIGURE 7. PoW based on block compression (PoW-PC).

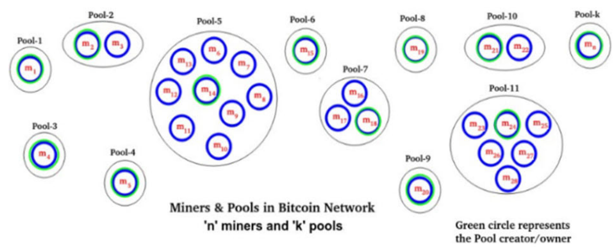


FIGURE 8. Hypothesis model.

mining pool. To address the challenge of consecutive blocks by the same miner, a proposed PoW protocol restricts the acceptance of six consecutive blocks from the same miner in the blockchain. However, if pool members cooperate to create blocks one by one with different miner IDs, they could still generate six consecutive blocks with different miner IDs, potentially enabling a 51% attack.

This research is premised on the assumption that complete information about pools is available, including details about pool members and pool creators. According to this premise, individual miners or pool creators are only allowed to create a new block, ensuring that "only those miners who mine as individuals or act as pool creators can participate in the consensus."

Pool members have the flexibility to leave or join other pools. A heuristic algorithm facilitates the extraction of payout flows from mining pools, enabling anyone to gather information about miners operating as pool members in specific pools [27]. Additionally, techniques involving block INV messages can be employed to identify mining nodes in the Bitcoin network [28]. Another algorithm, Heuristic 1, can be utilized to pinpoint mining nodes [29].

III. PROPOSED METHODOLOGY

This section presents the proposed methodology for addressing the 51% attack on the Bitcoin network. When a node possesses more than 51% computational power, it becomes capable of introducing deceptive information into the blockchain. Therefore, implementing restrictions in Bitcoin is crucial to prevent nodes from frequently adding fraudulent blocks to the blockchain.

In the Bitcoin network, transactions are confirmed as blocks are added to the blockchain. For significant transactions, it is recommended to have six confirmations for security reasons. Waiting for these confirmations can help mitigate 51% of attacks [30]. While having more computational power in the network can be beneficial when miners are honest, there's a risk that miners in a pool may misuse this power for malicious purposes, such as conducting a double-spending attack.

There are two distinct scenarios in which attackers can mount an attack:

- **Scenario-1:** The attacker aims to create a private chain whose length is greater than or equal to 6, known as the long private chain (LPC), to execute double spending on transactions requiring at least 6 confirmations.
- **Scenario-2:** The attacker aims to create a private chain whose length is less than 6, known as the short private chain (SPC), to execute double-spending on transactions requiring fewer confirmations.

Scenario-1 Solution:

In Scenario-1, the attacker tries to broadcast its private chain in such a way that the length of the private chain is greater than or equal to six (6), as well as the length of the private chain being greater than the length of the bypass chain.

The proposed algorithm, "Safe Mode Detection," is designed to handle Scenario-1 effectively. A restriction is imposed that miners can have at most five consecutive blocks by the same miner in their local honest blockchain. This modification in the consensus algorithm ensures the safety of major transactions. Figure 9 illustrates how the attacker initiates a private chain after the kth block runs parallel to the honest chain. The attacker attempts double spending in the (k+1)th block of the private chain and the bypass chain. Consequently, this private chain cannot be accepted by the network due to the presence of six consecutive blocks by the same miner.

When an attacker broadcasts a block and other nodes receive it, two cases may occur:

Case 1: The receiving block attempts to attach to the local blockchain. The receiving node first assesses the number of consecutive blocks by the same miner at the end of the local blockchain. If the count of consecutive blocks is less than five, the incoming block is safe to be appended to the local blockchain; otherwise, it is ignored.

Case 2: The receiving block requests an update of the local blockchain.

If the receiving block cannot be attached to the top of the local blockchain and its depth exceeds that of the last block in the local blockchain, the receiving node asks the miner to update its local blockchain. The miner keeps track of changes.

The miner reverts the changes if six consecutive blocks by the same miner are detected in its local blockchain; otherwise, the changes are accepted.

Scenario-2 Solution:

In Scenario-2, the attacker tries to broadcast its private chain in such a way that the length of the private chain is less than six (6) and the length of the private chain is greater than the length of the bypass chain.

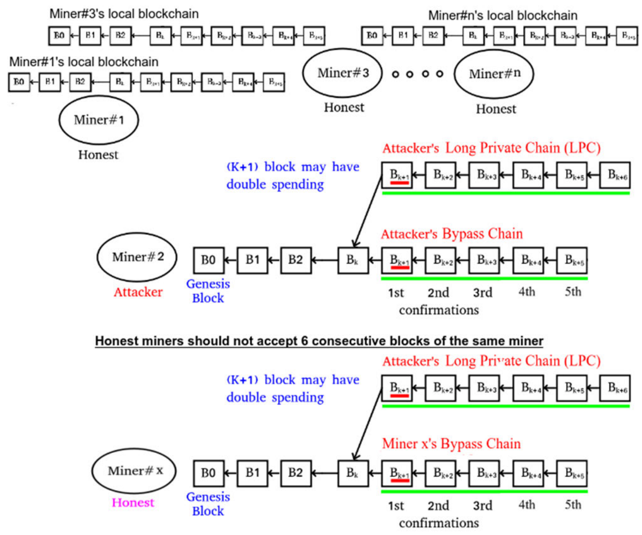


FIGURE 9. Methodology model for long private chain (LPC).

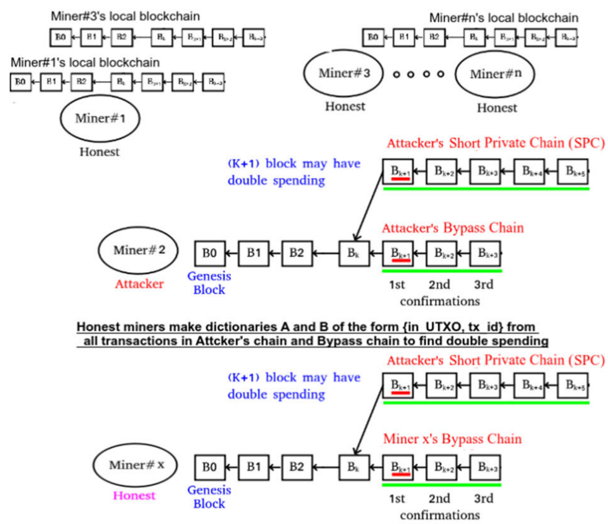


FIGURE 10. Methodology model for short private chain (SPC).

The proposed algorithm, "Safe Mode Detection," also handles Scenario-2 effectively. It requires an assessment of two parts of the chain:

- The receiving part of the attacker's blockchain.
- The bypass is part of the local, honest blockchain.

Figure 10 illustrates that an attacker's private chain length is less than six (6), so by creating dictionaries A and B, containing input UTXOs from the receiving part of the attacker's blockchain and the bypass part of the local honest blockchain, respectively, one can find common keys (input UTXOs) with different values (transaction IDs, tx.id). The presence of such entries indicates double spending; otherwise, the system will be in safe mode (see Algorithm 1).

Algorithm 1 The proposed modified consensus algorithm.

```
A = {}
for blk in Receiving_part:
    for tx in blk.transactions:
        for in_utxo in tx.in_utxo:
            A.update ({in_utxo: tx.id})
B = {}
    for blk in Bypass_part:
        for tx in blk.transactions:
            for in_utxo in tx.in_utxo:
                B.update ({in_utxo: tx.id})
intersect = [k for k in A if k in B and A[k] != B[k]]
if intersect == [ ]:
    SafeMode = True
else:
    SafeMode = False
```

Taking into account the average number of transactions in one block and the typical number of input UTXOs [31] in a Bitcoin transaction, it becomes feasible to compare dictionaries A and B to detect double spending. If there are approximately 2000 transactions in one block of the Bitcoin network [31], then there are 10,000 transactions in 5 blocks, which will be compared with the bypass chain. It is also noted that an average of 2.12 input UTXOs are used in one transaction on the Bitcoin network, and most transactions use only one input UTXO [32]. So dictionaries A or B may contain a maximum of 20,000 key values of input UTXOs that can easily be compared.

In the BlockSim simulator environment, the Bitcoin network conducts transactions at regular intervals. Miners select these transactions for validation and group them into a block. After solving the Proof of Work (PoW) algorithm by finding a nonce, the block is added to the blockchain, and the process continues with the next block. The fork resolution process helps determine which miner has mined the most blocks in the chain. The analysis also reveals how the Bitcoin network can avoid a miner's attempt to execute a 51% attack.

IV. EXPERIMENTAL SETUP AND RESULTS

The "Experimental Setup and Results" section describes the methodology used in the BlockSim simulator to analyze miner behavior in a dynamic environment. The section explains the simulation of a Bitcoin network with miners, including how the simulator handles 51% attacks and how modifying the consensus algorithm can prevent such attacks.

A. BLOCKSIM SIMULATOR

BlockSim is an open-source simulator written in Python designed for analyzing and experimenting with blockchain-based systems like Bitcoin. It provides a flexible environment for studying various blockchain networks and consensus algorithms. Key files in BlockSim include:

- InputsConfig.py: Initializes global variables for the simulation process.
- Node.py: Defines the basic properties of a node in the Bitcoin network, such as node ID, local blockchain, and transaction pool.
- Transaction.py: Maintains transaction properties, including the transaction hash, timestamp, sender, recipient, amount, size, and fee.
- Event.py: Manages event properties generated by nodes, with two event types: create_block and receive_block. It also maintains event queues.
- Scheduler.py: Creates events and manages them, including maintaining event queues.
- Consensus.py: Contains protocol and fork_resolution functions for generating global chains.
- Incentives.py: Calculates and distributes rewards among participating nodes.
- Statistics.py: Computes and prints simulation results.
- BlockCommit.py: Handles events during the simulation.

B. EXPERIMENTAL SETUP

1) SIMULATION LOGIC BEHIND THE SCENE

In this simulator, there are 100 nodes in this Bitcoin network that are working as miners. These miners are designated as M0, M1, M2,..., M100. In these miners, M2 is designated as a special miner that is increasing hash power gradually such that the whole hash power of the network is 100%. Each miner maintains a local blockchain whose first block is called the genesis block.

- It increases the hashing power of M2 gradually.
- It creates an event by calling a method.

Scheduler.createblockevent(node, blockTime).

The currentTime is the time when a node starts mining, and the blockTime is the time when the node finds the solution to the puzzle in the proof-of-work (PoW) consensus algorithm. This newly created event is then added to the queue. It is clear that there will initially be 100 events added to the queue.

The simulation starts here, as presented in Figure 11. It is time to handle these events with less time than the simulation time. A variable simTime holds the simulation time for 12 hours, i.e., 12*60*60 = 43200 seconds. A while loop handles all the events until the queue becomes empty. This queue is not a simple queue. It is a priority queue in which events are removed with the lowest blockTime. A variable next_event holds the time of the event (recently removed from the queue) generated. The method BlockCommit.handle_event (nextevent) is called in the loop to handle this event.

There are two types of events in the queue, as shown in Figure 12:

- **create_block event:**
  A method BlockCommit.generate_block(event) is called to create a block. In this method, the block is first validated and added to its local blockchain. After that, it is propagated to all other nodes in the Bitcoin network.
- **receive_block event:**
  A method BlockCommit.receive_block(event) is called to receive a block. In this method, the arriving block is

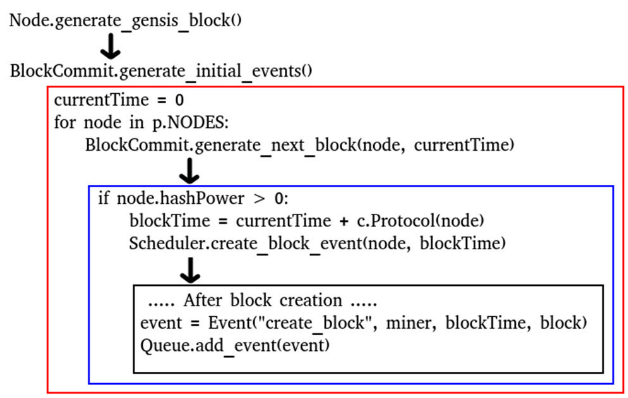


FIGURE 11. Simulation start-up.

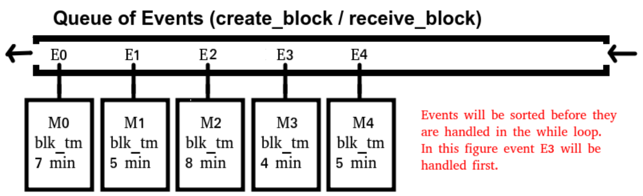


FIGURE 12. Priority queue.

first validated and adjusted to its local blockchain. This adjustment is done using two cases.
  - Case 1: The receiving block is adjusted at the top of the local blockchain.
  - Case 2: A method update_local_blockchain (node, miner, depth) is called for proper adjustment.

Figure 13 illustrates that a while loop manages all the events in the queue. Suppose the attacker decides to attack based on certain fundamental criteria. A specific method, BlockCommit.check_for_broadcast_private_chain(event), is invoked when the attacker creates or receives a block. This method evaluates two conditions prior to broadcasting the private chain. If the network approves the entire private chain, then 51% attacks can take place; conversely, if the network does not accept the private chain as a whole, the attacker's attempt to execute a 51% attack fails.

2) SYSTEM SAFE MODE DETECTION

The method Consensus.is_node_in_safe_mode(node, block, case) is designed to prevent other nodes from accepting the potentially risky private chain of the attacker, which can have a length equal to or greater than 6 blocks (referred to as LPC) or a length less than 6 blocks (referred to as SPC). This method determines whether the system is in a safe mode or not and is invoked in three different cases to handle the LPC situation initially:

- Case 1: Upon the creation of a new block, the system enters a safe mode if the newly created block is added to the local blockchain and there are no more than six consecutive blocks from the same miner.
- Case 2: When receiving a block adjusted to the top of the local blockchain, the system is in safe mode if the receiving block is added to the top of its local blockchain and there are no more than six consecutive blocks from the same miner.
- Case 3: When receiving a block that is not adjusted to the top of the local blockchain, a request is sent to the block miner to update its local blockchain. The system remains in safe mode if there are no more than six consecutive blocks from the same miner.

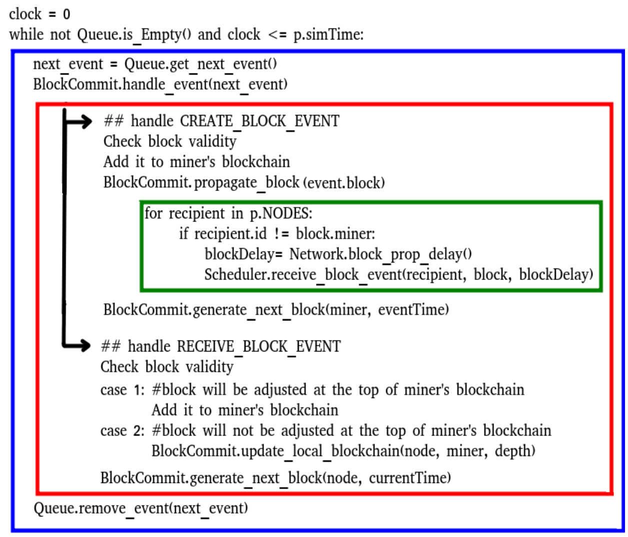


FIGURE 13. The while loop handling events.

To handle the SPC situation in case 3, this method employs two dictionaries, A and B. Dictionary A is created to store the receiving part of the attacker's private chain blockchain with dictionary items in the format of in_utxo:tx_id. Dictionary B is used to store the Bypass part of the attacker's honest local chain with dictionary items in the same format as A. The intersection of these dictionaries is then calculated as follows:

intersect = [k for k in A if k in B and A[k] != B[k]]

If there exists an in_utxo in both the private chain and the bypass chain but with different transaction IDs, it signifies double spending. The system remains in safe mode if the intersection is empty. Figure 14 provides a pseudo-code representation of the safe mode detection process.

After processing all the events in the queue, a global blockchain is generated using the method Consensus.fork_resolution The attacker does not participate in creating the global chain because its local blockchain is inconsistent. The distribution of rewards among the nodes is computed by invoking the method Incentives.distribute_rewards Simulation results, including block statistics and miner rewards, are calculated by calling the method Statistics.calculate

C. ALL CASES WITHOUT MODIFIED CONSENSUS ALGORITHM

In this Bitcoin network environment created by the BlockSim simulator, there are 100 miners (M1…M100), each possessing 1% of the total hashing power of the network, i.e., all miners have the same hash power. The simulation time has been set to 12 hours. During this simulation, miner

```
def is_node_in_safe_mode(node, block, case)

# FIRST CONDITION: CHECK 6 CONSECTIVE BLOCKS OF SAME MINER OCCUR OR NOT
#*****************************************************************************
# case_1      create_block Event:   next block will be adjusted at top
# case_2      receive_block Event: next block will be adjusted at top
# case_3      receive_block Event: next block will not be adjusted at top

if    there exists 6 consecutive block afer creating / receving a new block in any case:
      SafeMode = False
      return SafeMode
else:
      SafeMode = True

# SECOND CONDITION: CHECK DOUBLE SPENDING IN RECEIVING & BYPASS CHAIN
#*****************************************************************************
A = { }                                        B = { }
    for blk in node.receivingchain:                for blk in node.bypasschain:
        for tx in blk.transactions:                    for tx in blk.transactions:
            for in_utxo in tx.in_utxo:                     for in_utxo in tx.in_utxo:
                A.update ( {in_utxo: tx.id} )                  b.update( {in_utxo: tx.id} )

        intersect = [k for k in A if k in B and A[k] != B[k]]
```

FIGURE 14. Safe mode detection.

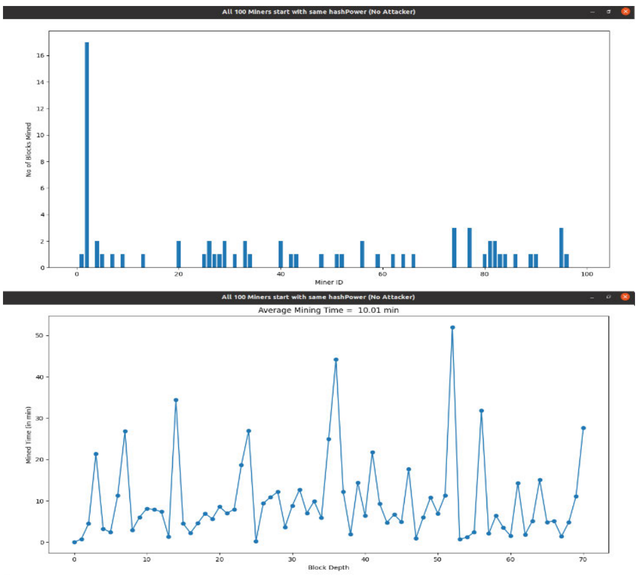


FIGURE 15. Case 1 – Number of blocks mined and average mining time.

M2 gradually increases its hashing power at regular, equal intervals of 2 hours over the entire duration. Concurrently, the remaining miners adjust their hashing power randomly to ensure that the combined hashing power of all miners remains at 100%. The objective is to observe how the miners mine blocks and how the network ultimately accepts the final global chain.

1) CASE 1: NO ATTACKER

Figure 15 presents two graphs, one depicting miner IDs and the number of mined blocks, while the other illustrates block depth and mining time in minutes. In contrast, Figure 16 displays a graph illustrating the relationship between the top 5 miner IDs, the number of blocks they have mined, simulation time, and corresponding timestamps. Figure 17 provides a detailed representation of the global chain that was accepted in Case 1.

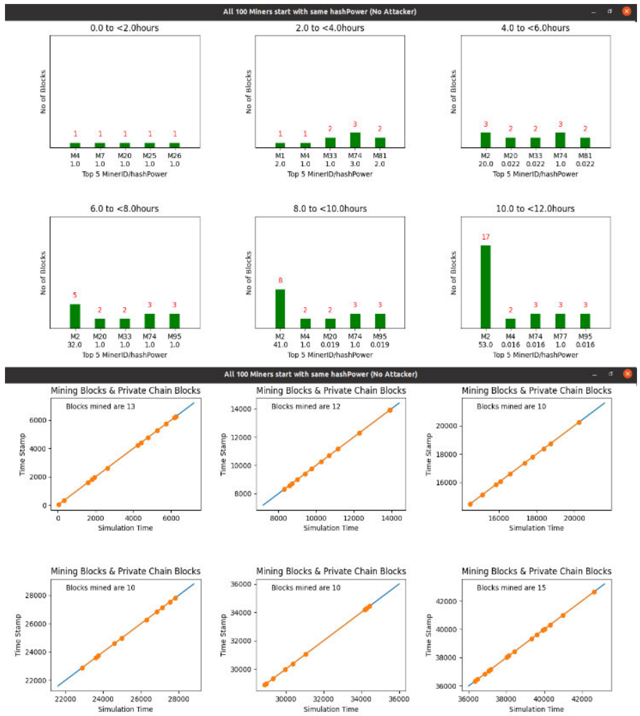


FIGURE 16. Case 1 – no attacker.

Final Accepted Global Blockchain::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
0	0	-1	0.00000000000000	None	0	1.0	0.00
1	13831512690	0	45.50534478459116	40	1936	1.0	0.76
2	43045823287	13831512690	316.90218212920200	48	1763	1.0	4.52
3	59195072854	43045823287	1600.83010909184623	95	1838	1.0	21.40
4	88652406026	59195072854	1796.28621130954048	4	1785	1.0	3.26
5	11697827442	88652406026	1939.43766574400524	74	1860	1.0	2.39
6	34173742910	11697827442	2616.82000803275514	26	1811	1.0	11.29
7	76007643906	34173742910	4229.95772201664749	20	1831	1.0	26.89
8	8757301010	76007643906	4406.41844414646221	81	1901	1.0	2.94
9	80685154551	8757301010	4769.94429510705049	25	1923	1.0	6.06
10	43674605445	80685154551	5257.57303471485284	80	1805	1.0	8.13
11	83559410720	43674605445	5730.27546662963323	27	1850	1.0	7.88
12	75539631272	83559410720	6172.53094642715769	51	1860	1.0	7.37
13	6129321051	75539631272	6255.14198540879079	7	1855	1.0	1.38
14	1746096949	6129321051	8318.28601946954950	5	1834	1.0	34.39
15	83770253888	1746096949	8587.27999038634698	1	1807	1.0	4.48
16	14315669828	83770253888	8720.92047913623355	33	1802	1.0	2.23
17	93002587063	14315669828	8995.89250261971938	74	1792	1.0	4.58
18	75808002735	93002587063	9411.80196056938075	28	1866	1.0	6.93
19	46534641299	75808002735	9751.61422975508140	81	1872	1.0	5.66
20	59819634622	46534641299	10266.58841688362918	82	1848	1.0	8.58
21	45987009853	59819634622	10686.10180473863329	64	1866	1.0	6.99
22	3849501018	45987009853	11162.39354551518591	74	1793	1.0	7.94
23	10955183669	3849501018	12282.95545182313253	33	1840	1.0	18.68
24	40459387111	10955183669	13898.92583344286868	89	1804	1.0	26.93
25	59207846359	40459387111	13910.97425514683164	59	1812	1.0	0.20
26	96901118867	59207846359	14473.43535866659113	20	1830	1.0	9.37
27	87169332554	96901118867	15126.71248616467346	13	1820	1.0	10.89
28	78772172090	87169332554	15856.43794958882245	2	1859	1.0	12.16
29	85883198008	78772172090	16076.48274156845764	96	1816	1.0	3.67
30	47658891137	85883198008	16606.54700740306725	2	1910	1.0	8.83
31	42836456711	47658891137	17370.57367723072457	2	1869	1.0	12.73
32	26598287929	42836456711	17793.46071218108773	9	1879	1.0	7.05
33	11870406026	26598287929	18386.17010088780808	95	1819	1.0	9.88
34	29468292039	11870406026	18739.33695382710357	77	1893	1.0	5.89
35	56155003490	29468292039	20234.15345682908446	42	2003	1.0	24.91
36	7866213461	56155003490	22886.09374419623782	31	1759	1.0	44.20
37	21417617398	7866213461	23615.58963702414258	34	1780	1.0	12.16
38	25008542613	21417617398	23730.48814271195442	2	1923	1.0	1.91
39	79634670426	25008542613	24594.06303977282369	2	1845	1.0	14.39
40	81253342099	79634670426	24980.72790891034674	40	1880	1.0	6.44
41	73793935627	81253342099	26285.59483218547030	56	1861	1.0	21.75
42	70578571767	73793935627	26845.01421502738231	29	1877	1.0	9.32
43	3887574815	70578571767	27130.90365106251920	95	1824	1.0	4.76
44	33696853994	3887574815	27532.53689484373535	56	1806	1.0	6.69
45	71610596283	33696853994	27825.24128710979494	52	1899	1.0	4.88
46	47520991993	71610596283	28885.60777538311959	77	1835	1.0	17.67
47	49008467415	47520991993	28944.08974475316063	29	1791	1.0	0.97
48	49359604606	49008467415	29307.88011484587696	4	1849	1.0	6.06
49	25400709016	49359604606	29958.63131248369973	2	1856	1.0	10.85
50	93022349177	25400709016	30375.57856711099521	84	1801	1.0	6.95
51	39389485606	93022349177	31055.58339890962816	82	1834	1.0	11.33
52	72002395567	39389485606	34176.18819133890793	90	1893	1.0	52.01
53	40980806553	72002395567	34218.82813043743954	2	1795	1.0	0.71
54	15944154685	40980806553	34290.16473251807474	2	1820	1.0	1.19
55	14029306196	15944154685	34434.61523887935618	26	1821	1.0	2.41
56	75003054649	14029306196	36346.33901438637258	83	1824	1.0	31.86
57	70607249022	75003054649	36470.95728209168010	2	1830	1.0	2.08
58	23805285467	70607249022	36855.94676534825703	2	1856	1.0	6.42
59	72763365297	23805285467	37066.89941206423100	66	1902	1.0	3.52
60	13222403096	72763365297	37158.94980087751901	2	1835	1.0	1.53
61	43606890355	13222403096	38017.32758304292656	62	1794	1.0	14.31
62	23915064978	43606890355	38129.63609652827290	2	1732	1.0	1.87
63	60930833351	23915064978	38437.31377265557239	86	1780	1.0	5.13
64	17280320112	60930833351	39343.15734679383604	2	1877	1.0	15.10
65	72836117571	17280320112	39634.03846013899602	2	1775	1.0	4.85
66	79770114707	72836117571	39941.64739929424104	2	1812	1.0	5.13
67	93119995035	79770114707	40025.16876718941785	2	1858	1.0	1.39
68	93428984578	93119995035	40314.27254166237981	43	1924	1.0	4.82
69	62412274640	93428984578	40981.71170564778004	77	1820	1.0	11.12
70	29708891678	62412274640	42641.65732753135671	2	1754	1.0	27.67

Average Mined Time = 10.01

ATTACK_INFO= [None, None, None]

FIGURE 17. Case 1 – global chain.

2) CASE 2: M2 IS ATTACKER

In Case 2, where all miners have the same hash power, with M2 acting as the attacker, an illegal activity is initiated as

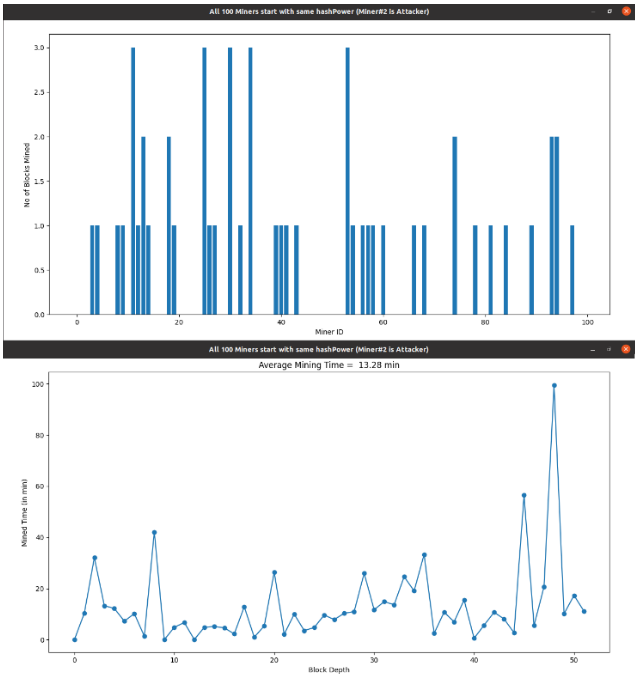


FIGURE 18. Case 2, Situation-1 – number of blocks mined and average mining time.

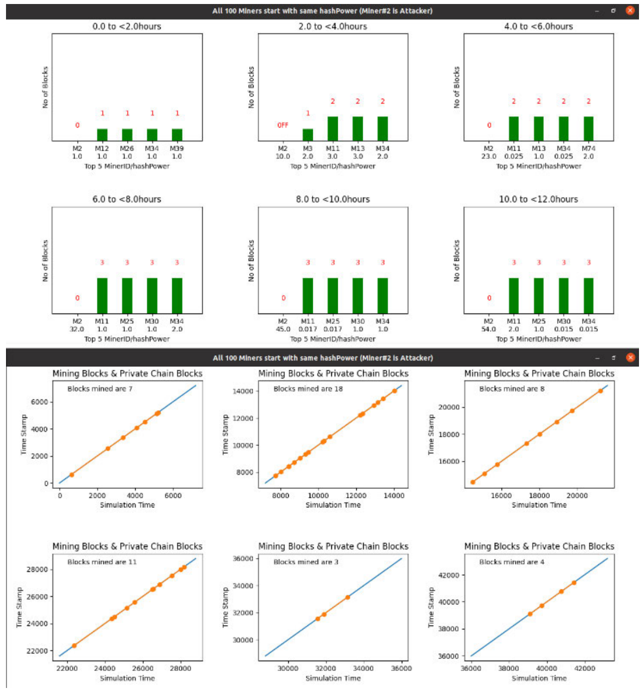


FIGURE 19. Case 2, Situation-1 – fails to broadcast private chain.

M2 attempts a 51% attack. The attacker begins to construct a private chain, and when its length reaches or exceeds 6 blocks, surpassing the length of the honest chain, it broadcasts the last block of the private chain. The attacker's start time is set to two hours, denoted as ATTACK_START_TIME = 2*60*60 seconds. Following this, the attacker proceeds to create a private chain and broadcast it.

There are three situations that may occur:

- Situation-1: Failure to Broadcast a Private Chain
- Situation-2: Failure to Execute a 51% Attack
- Situation-3: Successful 51% Attack

Situation-1: Failure to Broadcast a Private Chain:

Figure 18 illustrates that the attacker fails to meet the basic criteria for broadcasting the private chain due to simulation time constraints. Figure 19 provides a graph depicting the Top 5 Miner IDs, the number of blocks they have mined, simulation time, and corresponding timestamps. Additionally, Figure 20 presents details regarding the global chain accepted in Case 2, Situation-1.

Situation-2: Failure to Execute a 51% Attack:

The simulation suggests that this scenario may occur under rare conditions. If the attacker constructs a private chain and satisfies the basic criteria just before concluding the simulation and broadcasting it, it's possible that the final global chain won't accept the private chain because the simulation time limit is exceeded. In Case 2, Situation-2, if a 51% attack fails to occur, the figure will display TF. The first T indicates that the attacker broadcasted the private chain, while the second F indicates that the final global chain did not accept the entire private chain. However, it's important to note

Final Accepted Global Blockchain::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
0	0	-1	0.00000000000000	None	0	1.0	0.00
1	79161873552	0	624.77310885853228	94	1837	1.0	10.41
2	68113451977	79161873552	2548.60436097003731	78	1792	1.0	32.06
3	97134247099	68113451977	3347.97829163978395	26	1804	1.0	13.32
4	48724227299	97134247099	4080.83719217545968	57	1879	1.0	12.21
5	25293956125	48724227299	4516.82663779705581	12	1851	1.0	7.27
6	20325305062	25293956125	5128.42637319928144	39	1866	1.0	10.19
7	21968356793	20325305062	5212.27215137592430	34	1908	1.0	1.40
8	82558570756	21968356793	7740.36960941610050	30	1811	1.0	42.13
9	86765927072	82558570756	7741.67651801173997	11	1902	1.0	0.02
10	52240139852	86765927072	8033.76654705548481	68	1881	1.0	4.87
11	75986844441	52240139852	8436.96760395887031	93	1859	1.0	6.72
12	77403181563	75986844441	8441.67056257053991	32	1859	1.0	0.08
13	33963420578	77403181563	8733.26926344774802	40	1841	1.0	4.86
14	93343020185	33963420578	9045.10836346003271	13	1883	1.0	5.20
15	68760388963	93343020185	9323.80270764195302	19	1833	1.0	4.64
16	63505094949	68760388963	9460.05605849997482	81	1847	1.0	2.27
17	21049674041	63505094949	10233.86300951055273	25	1838	1.0	12.90
18	24146467436	21049674041	10293.69621520460169	43	1862	1.0	1.00
19	24884390926	24146467436	10616.66073540036814	97	1914	1.0	5.38
20	10708153268	24884390926	12202.89474635507577	3	1906	1.0	26.44
21	71395884803	10708153268	12333.31978426175556	34	1806	1.0	2.17
22	36877829149	71395884803	12933.13235910832373	11	1838	1.0	10.00
23	72010392179	36877829149	13148.26677718267456	74	1844	1.0	3.59
24	25961769740	72010392179	13433.26069430682219	13	1951	1.0	4.75
25	48632208594	25961769740	14014.00519677512784	58	1870	1.0	9.68
26	45305452373	48632208594	14485.65088711124736	41	1867	1.0	7.86
27	4186386628	45305452373	15105.80481854785467	89	1830	1.0	10.34
28	36144911864	4186386628	15766.88843526578785	4	1850	1.0	11.02
29	42737761032	36144911864	17323.85218003669070	94	1861	1.0	25.95
30	89404289433	42737761032	18026.96301624724947	27	1782	1.0	11.72
31	34324286272	89404289433	18919.47591132192247	56	1869	1.0	14.88
32	54218285616	34324286272	19739.41188758891440	14	1847	1.0	13.67
33	55513055143	54218285616	21217.44426779725472	74	1807	1.0	24.63
34	97457544546	55513055143	22361.21864225510217	30	1893	1.0	19.06
35	72180686880	97457544546	24360.79475475322397	18	1753	1.0	33.33
36	9573483733	72180686880	24509.60218824393087	53	1883	1.0	2.48
37	12620673775	9573483733	25155.60847996290977	18	1831	1.0	10.77
38	37649165058	12620673775	25575.64995406114031	93	1832	1.0	7.00
39	70348195241	37649165058	26509.42053859614316	34	1798	1.0	15.56
40	22607129725	70348195241	26543.00641422986519	11	1838	1.0	0.56
41	5387023913	22607129725	26883.85098635942995	25	1867	1.0	5.68
42	79951846874	5387023913	27527.04498820229492	30	1947	1.0	10.72
43	75120912370	79951846874	28014.34261129398510	84	1780	1.0	8.12
44	80040923369	75120912370	28177.48652184435923	25	1860	1.0	2.72
45	318091884	80040923369	31564.84020228454756	53	1855	1.0	56.46
46	67814085120	318091884	31896.86846666723795	53	1859	1.0	5.53
47	94673204376	67814085120	33140.78330154813011	60	1820	1.0	20.73
48	30997151009	94673204376	39114.84421136526362	9	1850	1.0	99.57
49	12117815554	30997151009	39721.11271537595167	8	1877	1.0	10.10
50	7674153949	12117815554	40760.89097289991332	54	1906	1.0	17.33
51	17392034058	7674153949	41428.27453031746700	66	1864	1.0	11.12

Average Mined Time = 13.28

```
ATTACK_INFO= [False, 36, -1]
Attacker Broadcast the Private Chain = False
Attacker's Private Chain Length = 36
Attacker Fails to broadcast private chain = -1
```

FIGURE 20. Case 2, Situation-1 – global chain.

that in real-time scenarios, this situation is unlikely to occur. Thus, it can be concluded that if an attacker successfully broadcasts the private chain, the network would typically accept it, resulting in a 51% attack.

Situation-3: Successful 51% Attack:

Figure 21 illustrates the graph for Case 2, Situation-3. In the event that the attacker constructs a private chain and meets the basic criteria for broadcasting it, it's possible for the final global chain to accept the private chain. Consider the

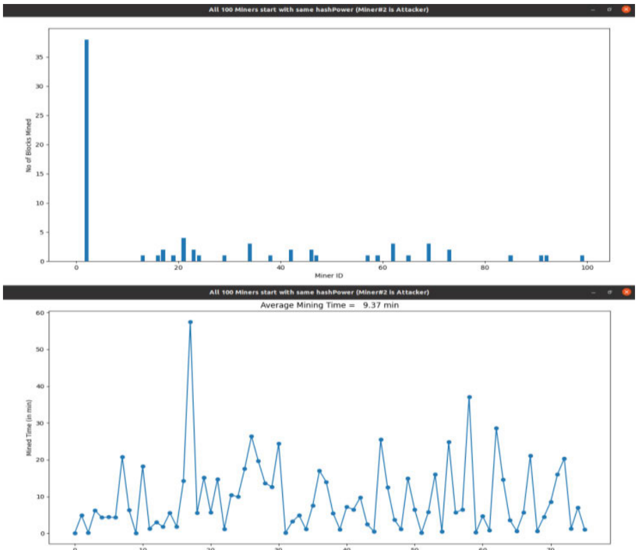


FIGURE 21. Case 2, Situation-3 – no of blocks mined and average mining time.

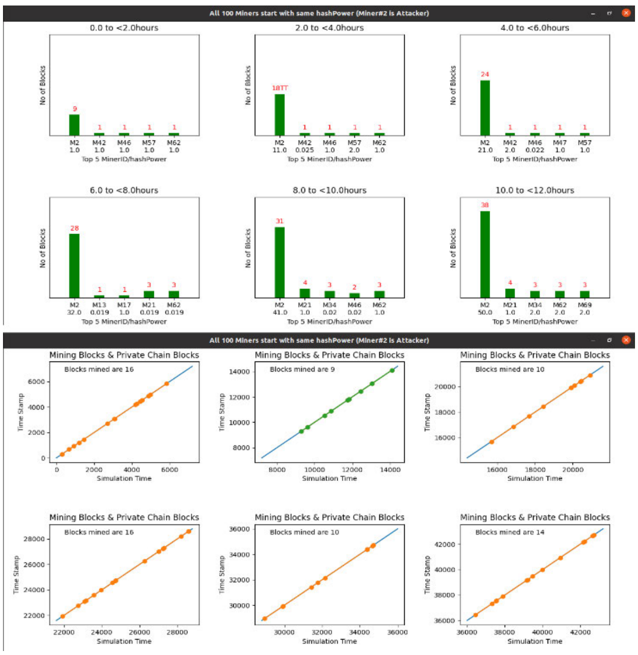
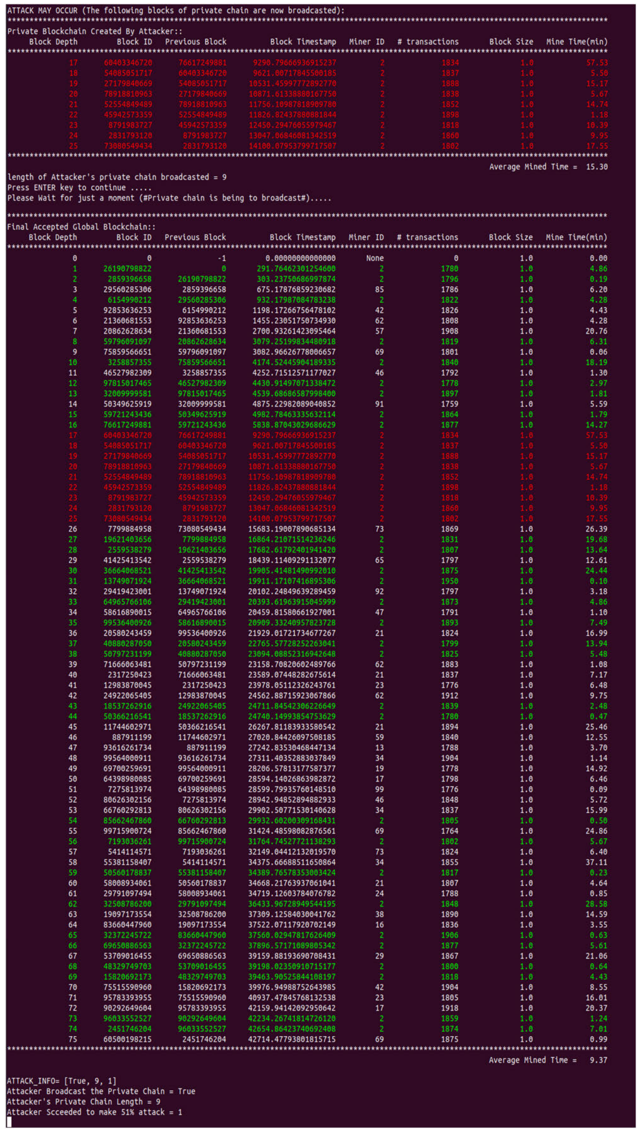


FIGURE 22. Case 2, Situation-3 – 51% attack.

TT in Figure 22, where the first T indicates that the attacker broadcasted the private chain, and the second T signifies that the final global chain accepted the entire private chain. Figure 22 displays the graph featuring the Top 5 Miner IDs, the number of blocks they have mined, simulation time, and corresponding timestamps. Furthermore, Figure 23 provides details about the global chain accepted in Case 2, Situation-3.

D. ALL CASES WITH MODIFIED CONSENSUS ALGORITHM

In the preceding section, all scenarios were examined without the implementation of a modified algorithm. In this section, we will utilize an adjusted version of the consensus algorithm and reevaluate all the cases.



ATTACK MAY OCCUR (The following blocks of private chain are now broadcasted):

Private Blockchain Created By Attacker::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
17	60403346720	76617249881	9290.79666936915237	2	1834	1.0	57.53
18	54085051717	60403346720	9621.00717845500185	2	1837	1.0	5.50
19	27179840669	54085051717	10531.45997772892770	2	1888	1.0	15.17
20	78918810963	27179840669	10871.61338880167750	2	1838	1.0	5.67
21	52554849489	78918810963	11756.10987818909780	2	1852	1.0	14.74
22	45942573359	52554849489	11826.82437880881844	2	1898	1.0	1.18
23	8791983727	45942573359	12450.29476055979467	2	1818	1.0	10.39
24	2831793120	8791983727	13047.06846081342519	2	1860	1.0	9.95
25	73080549434	2831793120	14100.07953799717507	2	1802	1.0	17.55

Average Mined Time = 15.30

length of Attacker's private chain broadcasted = 9
Press ENTER key to continue .....
Please Wait for just a moment (#Private chain is being to broadcast#).....

Final Accepted Global Blockchain::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
0	0	-1	0.00000000000000	None	0	1.0	0.00
1	26190798822	0	291.76462301254600	2	1780	1.0	4.86
2	2859396658	26190798822	303.23750686997874	2	1796	1.0	0.19
3	29560285306	2859396658	675.17876859230682	85	1786	1.0	6.20
4	6154990212	29560285306	932.17987084783238	2	1822	1.0	4.28
5	92853636253	6154990212	1198.17266756478102	42	1826	1.0	4.43
6	21360681553	92853636253	1455.23051750734930	62	1808	1.0	4.28
7	20862628634	21360681553	2700.93261423095464	57	1908	1.0	20.76
8	59796091097	20862628634	3079.25199834480918	2	1819	1.0	6.31
9	75859566651	59796091097	3082.96626778006657	69	1801	1.0	0.06
10	3258857355	75859566651	4174.52445904189335	2	1840	1.0	18.19
11	46527982309	3258857355	4252.71512571177027	46	1792	1.0	1.30
12	97815017465	46527982309	4430.91497071338472	2	1778	1.0	2.97
13	32009999581	97815017465	4539.68686587998400	2	1897	1.0	1.81
14	50349625919	32009999581	4875.22982089040852	91	1759	1.0	5.59
15	59721243436	50349625919	4982.78463335632114	2	1864	1.0	1.79
16	76617249881	59721243436	5838.87043029686629	2	1877	1.0	14.27
17	60403346720	76617249881	9290.79666936915237	2	1834	1.0	57.53
18	54085051717	60403346720	9621.00717845500185	2	1837	1.0	5.50
19	27179840669	54085051717	10531.45997772892770	2	1888	1.0	15.17
20	78918810963	27179840669	10871.61338880167750	2	1838	1.0	5.67
21	52554849489	78918810963	11756.10987818909780	2	1852	1.0	14.74
22	45942573359	52554849489	11826.82437880881844	2	1898	1.0	1.18
23	8791983727	45942573359	12450.29476055979467	2	1818	1.0	10.39
24	2831793120	8791983727	13047.06846081342519	2	1860	1.0	9.95
25	73080549434	2831793120	14100.07953799717507	2	1802	1.0	17.55
26	7799884958	73080549434	15683.19007890685134	73	1869	1.0	26.39
27	19621403656	7799884958	16864.21071514236246	2	1831	1.0	19.68
28	2559538279	19621403656	17682.61792401941420	2	1807	1.0	13.64
29	41425413542	2559538279	18439.11409291132077	65	1797	1.0	12.61
30	36664068521	41425413542	19905.41481490992010	2	1875	1.0	24.44
31	13749071924	36664068521	19911.17107416895306	2	1950	1.0	0.10
32	29419423001	13749071924	20102.24849639289459	92	1797	1.0	3.18
33	64965766106	29419423001	20393.61963915045999	2	1873	1.0	4.86
34	58616890015	64965766106	20459.81580661927001	47	1791	1.0	1.10
35	99536400926	58616890015	20909.33240957823728	2	1893	1.0	7.49
36	20580243459	99536400926	21929.01721734677267	21	1824	1.0	16.99
37	40880287050	20580243459	22765.57728252263041	2	1799	1.0	13.94
38	50797231199	40880287050	23094.08852316942648	2	1825	1.0	5.48
39	71666063481	50797231199	23158.70820602489766	62	1883	1.0	1.08
40	2317250423	71666063481	23589.07448282675614	21	1837	1.0	7.17
41	12983870045	2317250423	23978.05112326243761	23	1776	1.0	6.48
42	24922065405	12983870045	24562.88715923067866	62	1912	1.0	9.75
43	18537262916	24922065405	24711.84542306226649	2	1839	1.0	2.48
44	50366216541	18537262916	24740.14993854753629	2	1780	1.0	0.47
45	11744602971	50366216541	26267.81183933580542	21	1894	1.0	25.46
46	887911199	11744602971	27020.84426097508185	59	1840	1.0	12.55
47	93616261734	887911199	27242.83530468447134	13	1788	1.0	3.70
48	99564000911	93616261734	27311.40352883037849	34	1904	1.0	1.14
49	69700259691	99564000911	28206.57813177587377	19	1778	1.0	14.92
50	64398980085	69700259691	28594.14026863982872	17	1798	1.0	6.46
51	7275813974	64398980085	28599.79935760148510	99	1776	1.0	0.09
52	80626302156	7275813974	28942.94852894882933	46	1848	1.0	5.72
53	66760292813	80626302156	29902.50771530140628	34	1837	1.0	15.99
54	85662467860	66760292813	29932.60200309168431	2	1805	1.0	0.50
55	99715900724	85662467860	31424.48598082876561	69	1764	1.0	24.86
56	7193036261	99715900724	31764.74527721138293	2	1802	1.0	5.67
57	5414114571	7193036261	32149.04412132019570	73	1824	1.0	6.40
58	55381158407	5414114571	34375.66688511650864	34	1855	1.0	37.11
59	50560178837	55381158407	34389.76578353003424	2	1817	1.0	0.23
60	58008934061	50560178837	34668.21763937061041	21	1807	1.0	4.64
61	29791097494	58008934061	34719.12603784076782	24	1788	1.0	0.85
62	32508786200	29791097494	36433.96728949544195	2	1848	1.0	28.58
63	19097173554	32508786200	37309.12584030041762	38	1890	1.0	14.59
64	83660447960	19097173554	37522.07117920702149	16	1836	1.0	3.55
65	32372245722	83660447960	37560.02947817626409	2	1906	1.0	0.63
66	69650886563	32372245722	37896.57171089805342	2	1877	1.0	5.61
67	53709016455	69650886563	39159.88193690708431	29	1867	1.0	21.06
68	48329749703	53709016455	39198.02350910715177	2	1800	1.0	0.64
69	15820692173	48329749703	39463.90525844108197	2	1818	1.0	4.43
70	75515590960	15820692173	39976.94988752643985	42	1904	1.0	8.55
71	95783393955	75515590960	40937.47845768132538	23	1805	1.0	16.01
72	90292649604	95783393955	42159.94142092950642	17	1918	1.0	20.37
73	96033552527	90292649604	42234.26741814726120	2	1859	1.0	1.24
74	2451746204	96033552527	42654.86423740692408	2	1874	1.0	7.01
75	60500198215	2451746204	42714.47793801815715	69	1875	1.0	0.99

Average Mined Time = 9.37

ATTACK_INFO= [True, 9, 1]
Attacker Broadcast the Private Chain = True
Attacker's Private Chain Length = 9
Attacker Scceeded to make 51% attack = 1

FIGURE 23. Case 2, Situation-3 – global chain.

1) CASE 1: NO ATTACKER

Figure 24 presents two graphs illustrating the Miner IDs and the number of blocks they have mined, as well as block depth and mining time in minutes. Additionally, Figure 25 displays a graph representing the Top 5 Miner IDs, the quantity of blocks they have mined, simulation time, and corresponding timestamps. Furthermore, Figure 26 provides an in-depth analysis of the global chain's acceptance in Case 1.

2) CASE 2: M2 IS ATTACKER

In this scenario, once more, Miner M2 assumes the role of the attacker attempting a 51% attack. However, this time, the modified algorithm monitors the system's state to ensure it remains in a secure and consistent condition. Two distinct situations may now arise.

- Situation-1: Failure to Broadcast a Private Chain
- Situation-2: Inability to Execute a 51% Attack

Situation-1: Failure to Broadcast a Private Chain:

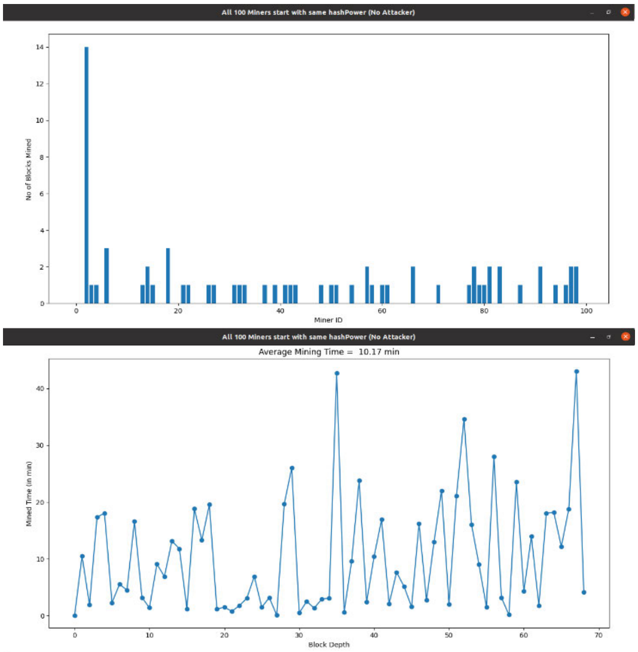


FIGURE 24. Modified algorithm case 1 no of blocks mined and average mining time.

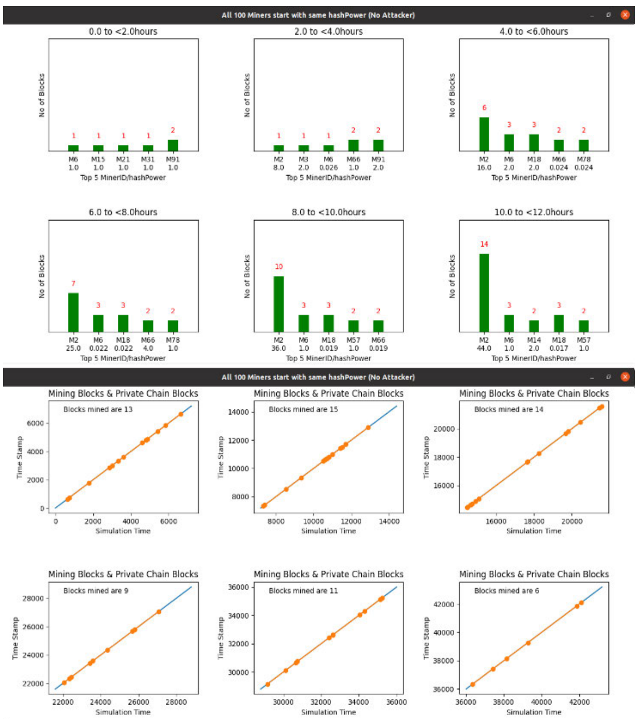
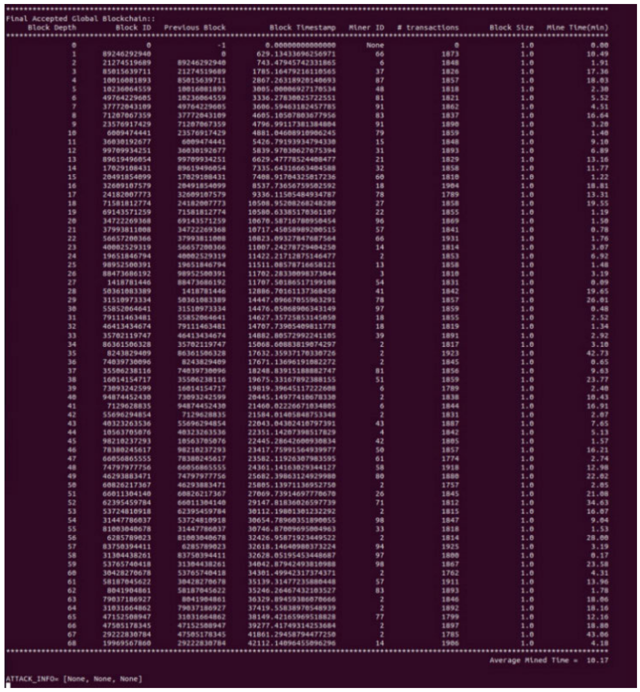


FIGURE 25. Modified Algorithm Case 1 – no attacker.


Final Accepted Global Blockchain::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
0	0	-1	0.0000000000000	None	0	1.0	0.00
1	89246292940	0	629.13433696256971	66	1873	1.0	10.49
2	21274519689	89246292940	743.47945742331865	6	1848	1.0	1.91
3	85015639711	21274519689	1785.16479216110565	37	1826	1.0	17.36
4	10016081893	85015639711	2867.26318920140693	87	1857	1.0	18.03
5	10236064559	10016081893	3005.00006927170534	48	1818	1.0	2.30
6	49764229605	10236064559	3336.27830025722551	81	1821	1.0	5.52
7	37772043109	49764229605	3606.59463182457785	91	1862	1.0	4.51
8	71207067359	37772043109	4605.10507803677956	83	1837	1.0	16.64
9	23576917429	71207067359	4796.99117381384804	91	1890	1.0	3.20
10	6009474441	23576917429	4881.04608910906245	79	1859	1.0	1.40
11	36030192677	6009474441	5426.79193934794330	15	1848	1.0	9.10
12	99709934251	36030192677	5839.97030627675394	31	1893	1.0	6.89
13	89619496054	99709934251	6629.47778524408477	21	1829	1.0	13.16
14	17029108431	89619496054	7335.64316663404588	32	1858	1.0	11.77
15	20491854099	17029108431	7408.91704325817236	60	1810	1.0	1.22
16	32609107579	20491854099	8537.73656759502592	18	1904	1.0	18.81
17	24182007773	32609107579	9336.11505484934787	78	1789	1.0	13.31
18	71581812774	24182007773	10508.95208268248280	27	1858	1.0	19.55
19	69143571259	71581812774	10580.63385170361107	22	1855	1.0	1.19
20	34722269368	69143571259	10670.58716780950454	96	1869	1.0	1.50
21	37993811008	34722269368	10717.45058989200515	57	1841	1.0	0.78
22	56657200366	37993811008	10823.09327847687564	66	1931	1.0	1.76
23	40002529319	56657200366	11007.24278729404250	14	1814	1.0	3.07
24	19651846794	40002529319	11422.21712875146477	2	1853	1.0	6.92
25	98952500391	19651846794	11511.08578716658121	13	1858	1.0	1.48
26	88473686192	98952500391	11702.28330098373044	3	1810	1.0	3.19
27	1418781446	88473686192	11707.50186517199108	54	1831	1.0	0.09
28	50361083389	1418781446	12886.70161137368450	41	1842	1.0	19.65
29	31510973334	50361083389	14447.09667055963291	78	1857	1.0	26.01
30	55852064641	31510973334	14476.05068906343149	97	1859	1.0	0.48
31	79111463481	55852064641	14627.35725853145050	18	1855	1.0	2.52
32	46413434674	79111463481	14707.73985409811778	18	1819	1.0	1.34
33	35702119747	46413434674	14882.80572992241105	39	1891	1.0	2.92
34	86361506328	35702119747	15068.60883819074297	2	1817	1.0	3.10
35	8243829409	86361506328	17632.35937170330726	2	1923	1.0	42.73
36	74039730096	8243829409	17671.13696191082272	2	1845	1.0	0.65
37	35506238116	74039730096	18248.83915188882747	81	1856	1.0	9.63
38	16014154717	35506238116	19675.33167892388155	51	1859	1.0	23.77
39	73093242599	16014154717	19819.39645117222608	6	1789	1.0	2.40
40	94874452430	73093242599	20445.14977410678330	2	1838	1.0	10.43
41	7129628835	94874452430	21460.02226671034805	6	1844	1.0	16.91
42	55696294854	7129628835	21584.01405848753348	2	1831	1.0	2.07
43	40323263536	55696294854	22043.04302410797391	43	1887	1.0	7.65
44	10563705076	40323263536	22351.14207398517829	4	1842	1.0	5.13
45	98210237293	10563705076	22445.28642600930834	42	1805	1.0	1.57
46	78380245617	98210237293	23417.75991564939977	50	1857	1.0	16.21
47	66056865555	78380245617	23582.11926307983595	61	1774	1.0	2.74
48	74797977756	66056865555	24361.14163029344127	58	1918	1.0	12.98
49	46293883471	74797977756	25682.39863124929980	80	1880	1.0	22.02
50	60826217367	46293883471	25805.13971136952750	2	1757	1.0	2.05
51	66011304140	60826217367	27069.73914697770670	26	1845	1.0	21.08
52	62395459784	66011304140	29147.81836026597739	71	1812	1.0	34.63
53	53724810918	62395459784	30112.19801301232292	2	1815	1.0	16.07
54	31447786037	53724810918	30654.78960351890055	98	1847	1.0	9.04
55	81003040678	31447786037	30746.87009695004963	33	1818	1.0	1.53
56	6285789023	81003040678	32426.95871923449522	2	1814	1.0	28.00
57	83750394411	6285789023	32618.14640980373224	94	1925	1.0	3.19
58	31304438261	83750394411	32628.05195453448687	97	1800	1.0	0.17
59	53765740418	31304438261	34042.87942493810988	98	1867	1.0	23.58
60	30428270678	53765740418	34301.49942317374371	2	1762	1.0	4.31
61	58187045622	30428270678	35139.31477235880448	57	1911	1.0	13.96
62	8041904861	58187045622	35246.26467432103527	83	1893	1.0	1.78
63	79037186927	8041904861	36329.89459386070666	2	1846	1.0	18.06
64	31031664862	79037186927	37419.55838970548939	2	1892	1.0	18.16
65	47152508947	31031664862	38149.42165969518828	77	1799	1.0	12.16
66	47505178345	47152508947	39277.41749314253684	2	1897	1.0	18.80
67	29222830784	47505178345	41861.29458794477250	2	1785	1.0	43.06
68	19969567860	29222830784	42112.14096455096296	14	1906	1.0	4.18

Average Mined Time = 10.17

ATTACK_INFO= [None, None, None]

FIGURE 26. Modified Algorithm Case 1 – no attacker.

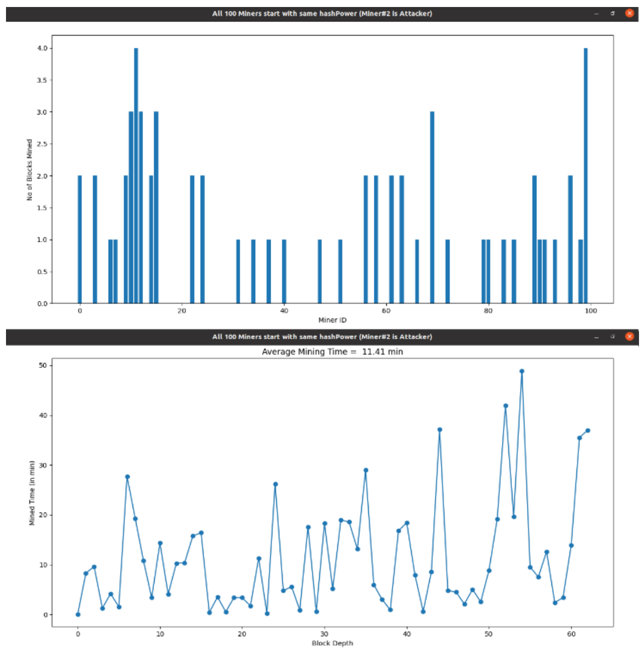


FIGURE 27. Modified Algorithm Case 2, Situation-1 – number of blocks mined and average mining time.

Figure 27 illustrates that if the attacker fails to meet the basic criteria for broadcasting the private chain and the simulation time exceeds the threshold, then the attacker is unable to broadcast the private chain. Meanwhile, Figure 28 displays the graph depicting the Top 5 Miner IDs, the number of blocks mined, and the simulation time along with timestamps. Additionally, Figure 29 presents details regarding the global chain accepted in Case 2, Situation-1.

Situation-2: Inability to Execute a 51% Attack:

The modified algorithm guarantees that once a block is created or received into the local blockchain, the system remains in a safe mode.

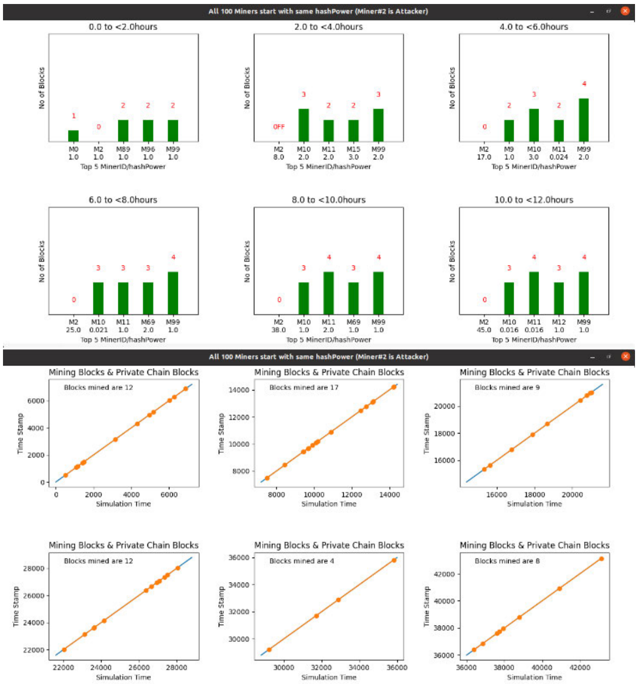


FIGURE 28. Modified Algorithm Case 2, Situation-1 – fails to broadcast private chain.

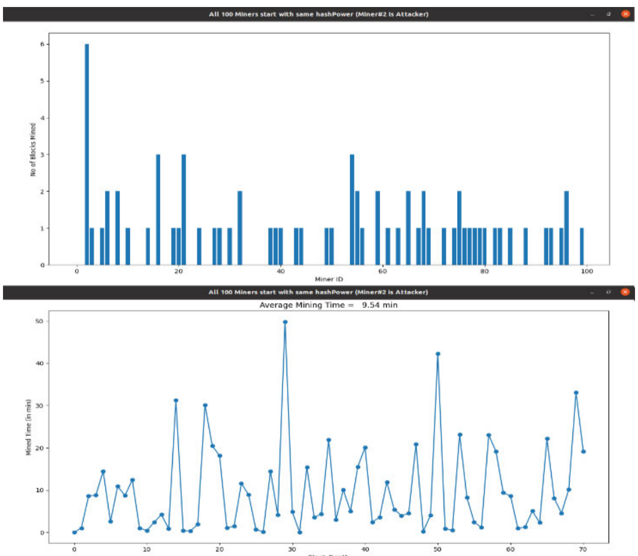
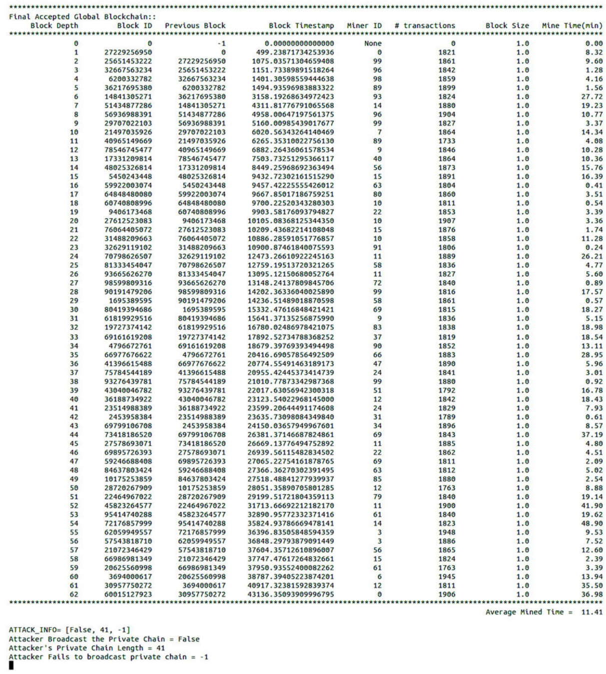


FIGURE 29. Modified Algorithm Case 2, Situation-1 – global chain.

If the system attempts to deviate from this safe mode, it simply disregards the block and refrains from adding it to the local blockchain. The modified algorithm consistently ignores the private chain when the attacker attempts to broadcast it. Figure 30 illustrates the graph for Case 2, Situation-2, where a 51% attack fails to materialize. Additionally, Figure 31 presents the graph that showcases the Top 5 Miner IDs, the number of blocks mined, the simulation time, and the associated timestamps. Additionally, Figure 32 presents details regarding the global chain accepted in Case 2, Situation-2.



Final Accepted Global Blockchain::

Block Depth	Block ID	Previous Block	Block Timestamp	Miner ID	# transactions	Block Size	Mine Time(min)
0	0	-1	0.00000000000000	None	0	1.0	0.00
1	27229256950	0	499.23871734253936	0	1821	1.0	8.32
2	25651453222	27229256950	1075.03571304659408	99	1861	1.0	9.60
3	32667563234	25651453222	1151.73389891518264	96	1842	1.0	1.28
4	6200332782	32667563234	1401.30598559444638	98	1859	1.0	4.16
5	36217695380	6200332782	1494.93596983883322	89	1899	1.0	1.56
6	14841305271	36217695380	3158.19268634972423	93	1824	1.0	27.72
7	51434877286	14841305271	4311.81776791065568	14	1880	1.0	19.23
8	56936988391	51434877286	4958.00647197561375	96	1904	1.0	10.77
9	29707022103	56936988391	5160.00985439017677	99	1827	1.0	3.37
10	21497035926	29707022103	6020.56343264140469	7	1864	1.0	14.34
11	40965149669	21497035926	6265.35310022756130	89	1733	1.0	4.08
12	78546745477	40965149669	6882.26436061578534	9	1846	1.0	10.28
13	17331209814	78546745477	7503.73251295366117	40	1864	1.0	10.36
14	48025326814	17331209814	8449.25968692363494	56	1873	1.0	15.76
15	5450243448	48025326814	9432.72302161515290	15	1891	1.0	16.39
16	59922003074	5450243448	9457.42225555426012	63	1804	1.0	0.41
17	64848480080	59922003074	9667.85017186759251	80	1860	1.0	3.51
18	60740808996	64848480080	9700.22520343280303	10	1811	1.0	0.54
19	9406173468	60740808996	9903.58176093794827	22	1853	1.0	3.39
20	27612523083	9406173468	10105.08368125344350	10	1907	1.0	3.36
21	76064405072	27612523083	10209.43682214108048	15	1876	1.0	1.74
22	31488209663	76064405072	10886.28591051776857	10	1858	1.0	11.28
23	32629119102	31488209663	10900.87461840075593	91	1806	1.0	0.24
24	70798626507	32629119102	12473.26610922245163	11	1889	1.0	26.21
25	81333454047	70798626507	12759.19513720321265	58	1836	1.0	4.77
26	93665626270	81333454047	13095.12150680052764	11	1827	1.0	5.60
27	98599809316	93665626270	13148.24137809845706	72	1840	1.0	0.89
28	90191479206	98599809316	14202.36336040025890	99	1816	1.0	17.57
29	1695389595	90191479206	14236.51489018870598	58	1861	1.0	0.57
30	80419394686	1695389595	15332.47616848421421	69	1815	1.0	18.27
31	61819929516	80419394686	15641.37135256875990	9	1836	1.0	5.15
32	19727374142	61819929516	16780.02486978421075	83	1838	1.0	18.98
33	69161619208	19727374142	17892.52734788368252	37	1819	1.0	18.54
34	4796672761	69161619208	18679.39769393494498	90	1852	1.0	13.11
35	66977676622	4796672761	20416.69057856492509	66	1883	1.0	28.95
36	41396615488	66977676622	20774.55491463189173	47	1890	1.0	5.96
37	75784544189	41396615488	20955.42445373414739	24	1841	1.0	3.01
38	93276439781	75784544189	21010.77873342987368	99	1880	1.0	0.92
39	43040046782	93276439781	22017.63056942300318	51	1792	1.0	16.78
40	36188734922	43040046782	23123.54022968145000	12	1842	1.0	18.43
41	23514988389	36188734922	23599.20644491174608	24	1829	1.0	7.93
42	2453958384	23514988389	23635.73098084349840	31	1789	1.0	0.61
43	69799106708	2453958384	24150.03657949967601	34	1896	1.0	8.57
44	73418186520	69799106708	26381.37146687824861	69	1843	1.0	37.19
45	27578693071	73418186520	26669.13776494752892	11	1885	1.0	4.80
46	69895726393	27578693071	26939.56115482834502	22	1862	1.0	4.51
47	59246688408	69895726393	27065.22754161878765	69	1811	1.0	2.09
48	84637803424	59246688408	27366.36270302391495	63	1812	1.0	5.02
49	10175253859	84637803424	27518.48841277939937	85	1880	1.0	2.54
50	28720267909	10175253859	28051.35890705801285	12	1763	1.0	8.88
51	22464967022	28720267909	29199.51721804359113	79	1840	1.0	19.14
52	45823264577	22464967022	31713.66692212182170	11	1900	1.0	41.90
53	95414740288	45823264577	32890.95772332371416	61	1840	1.0	19.62
54	72176857999	95414740288	35824.93786669478141	14	1823	1.0	48.90
55	62059949557	72176857999	36396.83505848594359	3	1948	1.0	9.53
56	57543818710	62059949557	36848.29793879091449	3	1886	1.0	7.52
57	21072346429	57543818710	37604.35712610896007	56	1865	1.0	12.60
58	66986981349	21072346429	37747.47617264832661	15	1824	1.0	2.39
59	20625560998	66986981349	37950.93552400082262	61	1763	1.0	3.39
60	3694000617	20625560998	38787.39405223874201	6	1945	1.0	13.94
61	30957750272	3694000617	40917.32381592839374	12	1811	1.0	35.50
62	60015127923	30957750272	43136.35093909996795	0	1906	1.0	36.98

Average Mined Time = 11.41

ATTACK_INFO= [False, 41, -1]
Attacker Broadcast the Private Chain = False
Attacker's Private Chain Length = 41
Attacker Fails to broadcast private chain = -1

FIGURE 30. Modified Algorithm Case 2 – Situation-2 – no of blocks mined & avg mining time.

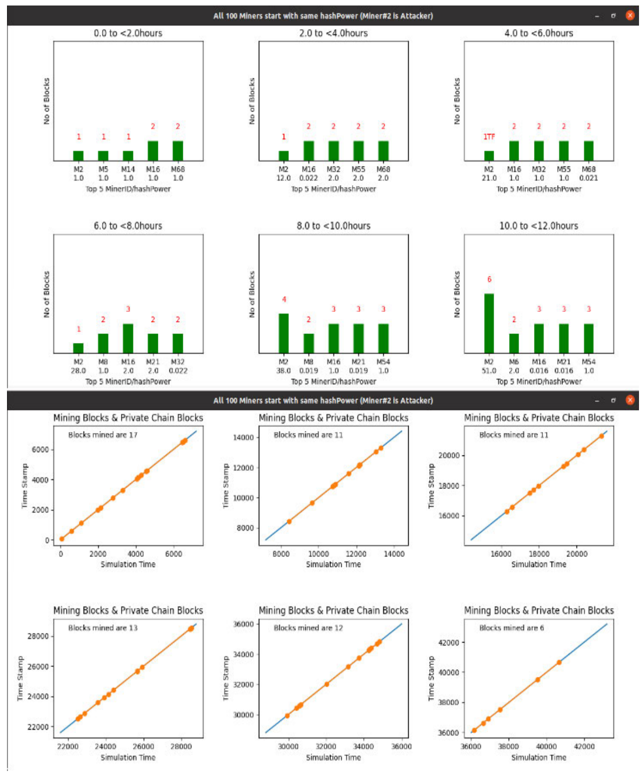


FIGURE 31. Modified Algorithm Case 2 – Situation-2 – fails to make 51% attack.

E. SHORT PRIVATE CHAIN (SPC) DOUBLE SPENDING CASE (WITHOUT MODIFIED CONSENSUS ALGORITHM)

In the preceding section, it was established that the system would not accept a longest private chain (LPC) with a length

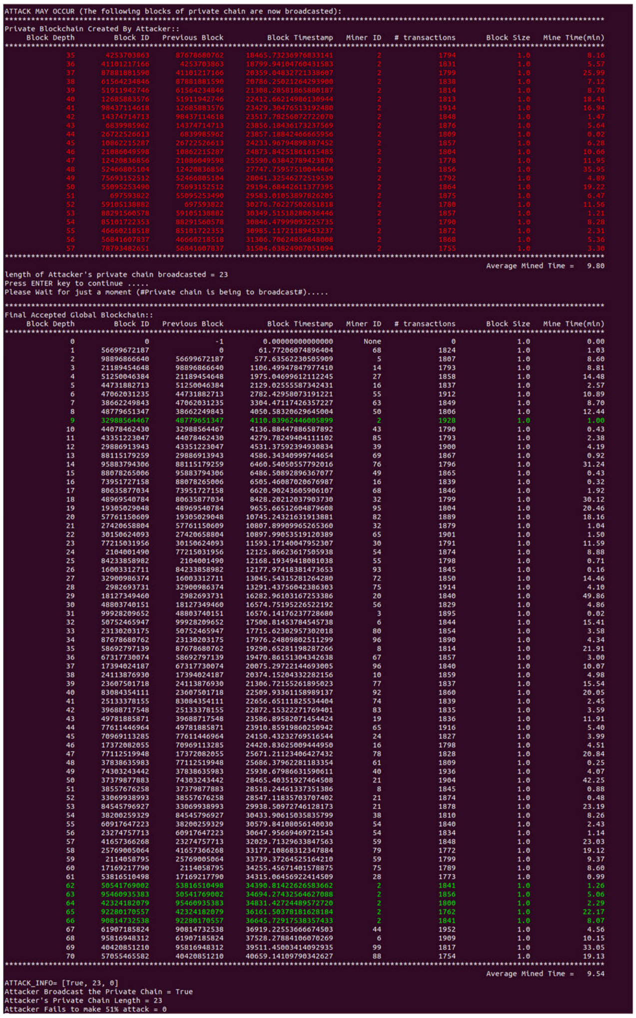


FIGURE 32. Modified Algorithm Case 2 – Situation-2 – global chain.

greater than or equal to six blocks from the same miner. However, the attacker retains the option to broadcast a short private chain (SPC), which may consist of fewer than 6 blocks, typically around 4 or 5 blocks, all from the same miner. This scenario opens the possibility of engaging in double-spending activities.

Figure 33 visually presents two graphs: one detailing miner IDs and the number of mined blocks, and the other depicting block depth and mining time in minutes. Concurrently, Figure 34 displays a graph illustrating the Top 5 Miner IDs, the number of blocks mined, and their correlation with simulation time and timestamps.

Figure 35 further illustrates instances of double spending added to both the private chain and bypass chain. For example, the figure showcases the initial transaction within the first block of the private chain, featuring input UTXO 3978543203 and transaction ID 83869265366. Notably, this same input UTXO is observed within the first block transactions of the bypass chain, albeit bearing a distinct transaction ID of 729729714. This specific double-spending scenario involves the transfer of 0.72 bitcoins to different recipient IDs. The intricate details regarding the global chain's acceptance in the context of a short private chain (SPC), double spending, and potential 51% attacks are outlined in Figure 36.

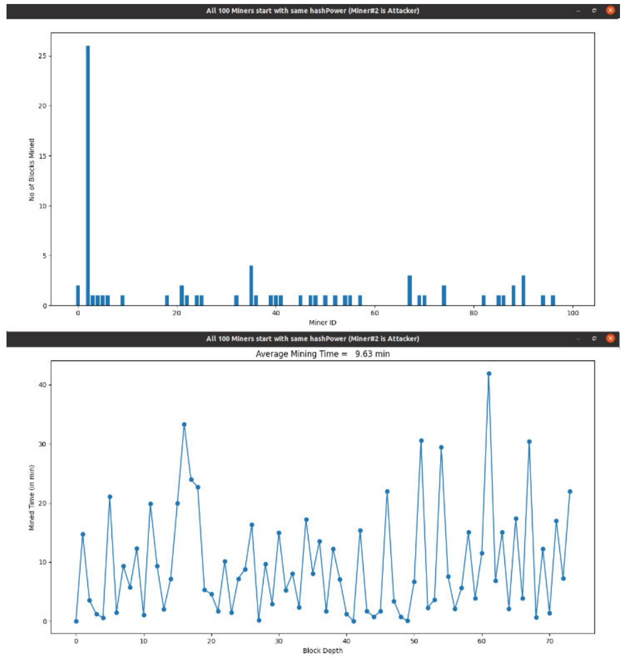


FIGURE 33. Without modified algorithm (SPC) – number of blocks mined and average mining time.

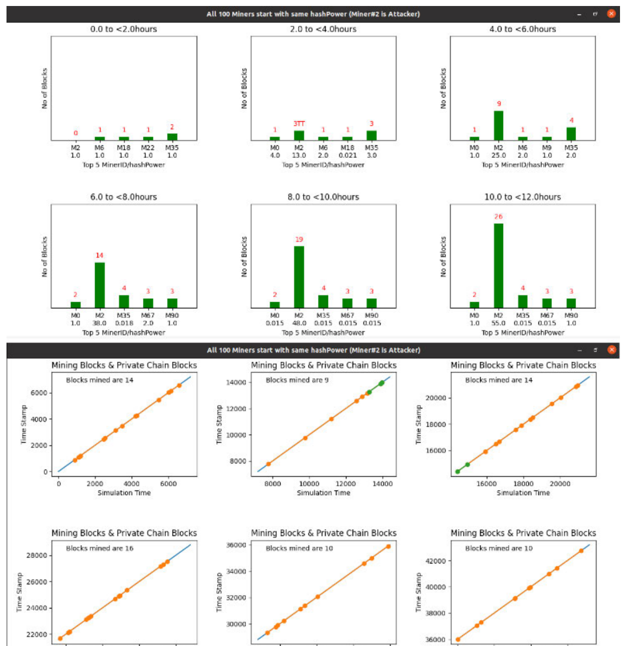


FIGURE 34. Without modified algorithm (SPC) – 51% attack.

F. SHORT PRIVATE CHAIN (SPC) DOUBLE SPENDING CASE (WITH MODIFIED CONSENSUS ALGORITHM)

The shortest private chain (SPC), comprising fewer than or up to six blocks—typically around 4 or 5 blocks—from the

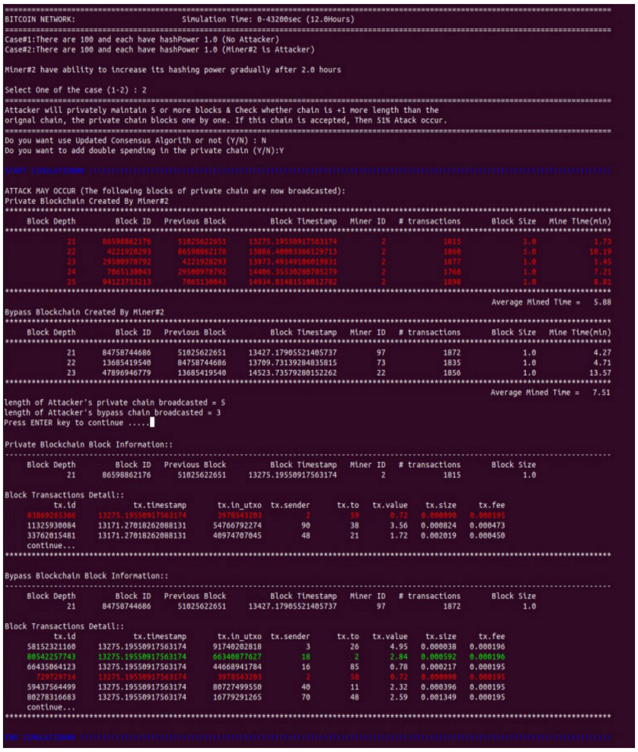


FIGURE 35. **Without modified algorithm (SPC) – double spending added.**

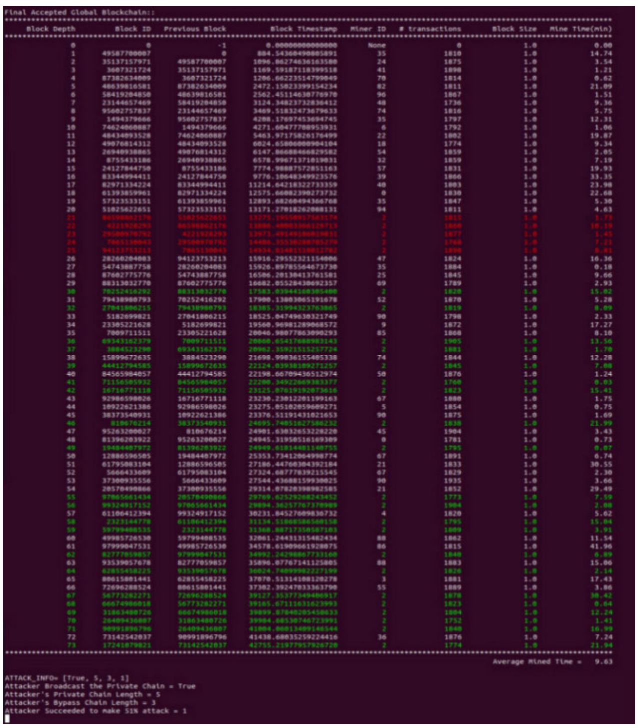


FIGURE 36. **Without modified algorithm (SPC) – global chain with double spending.**

same miner, can be accepted by the system. In such cases, the potential for double-spending activities exists. However, when implementing the proposed modified algorithm, the

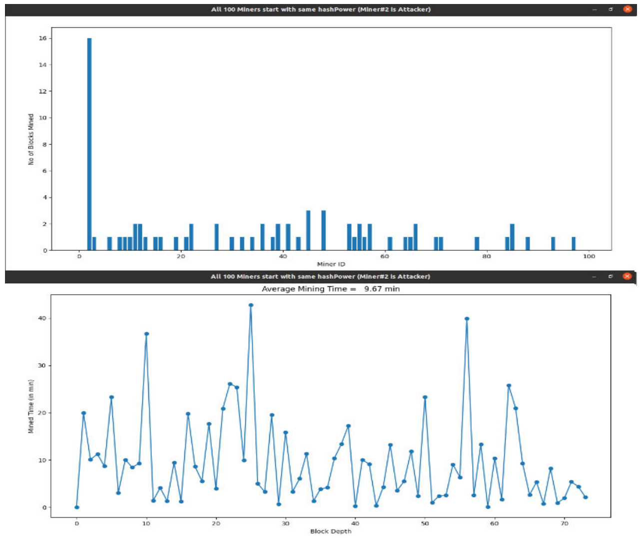


FIGURE 37. **With modified algorithm (SPC) – number of blocks mined and average mining time.**

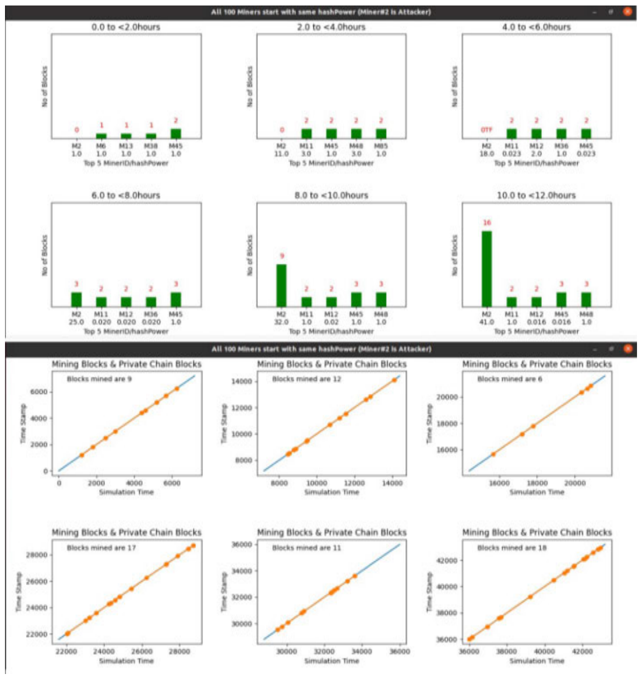


FIGURE 38. **With modified algorithm (SPC) – no 51% attack.**

occurrence of a 51% attack is prevented. Figure 37 visually presents two graphs: one detailing miner IDs and the number of mined blocks, and the other depicting block depth and mining time in minutes. In addition, Figure 38 provides a graph illustrating the Top 5 Miner IDs, the number of blocks mined, and their correlation with simulation time and timestamps.

Figure 39 offers a detailed view of double spending within both the private chain and bypass chain. Specifically, the figure highlights the initial transaction within the first block of the private chain, featuring input UTXO 86424129216 and transaction ID 81373710733. Notably, this

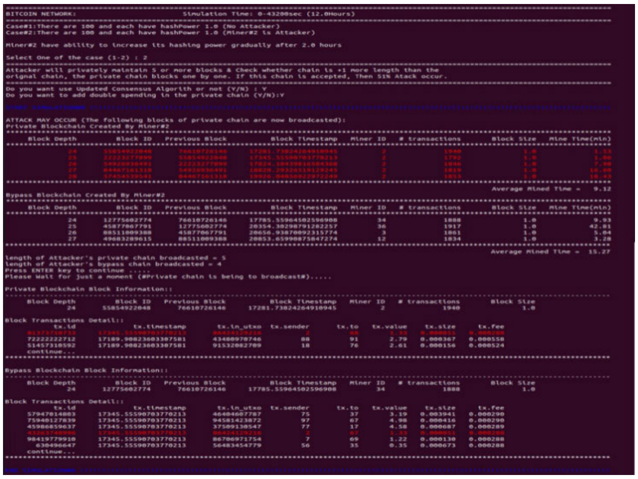


FIGURE 39. With modified algorithm (SPC) – double spending added.

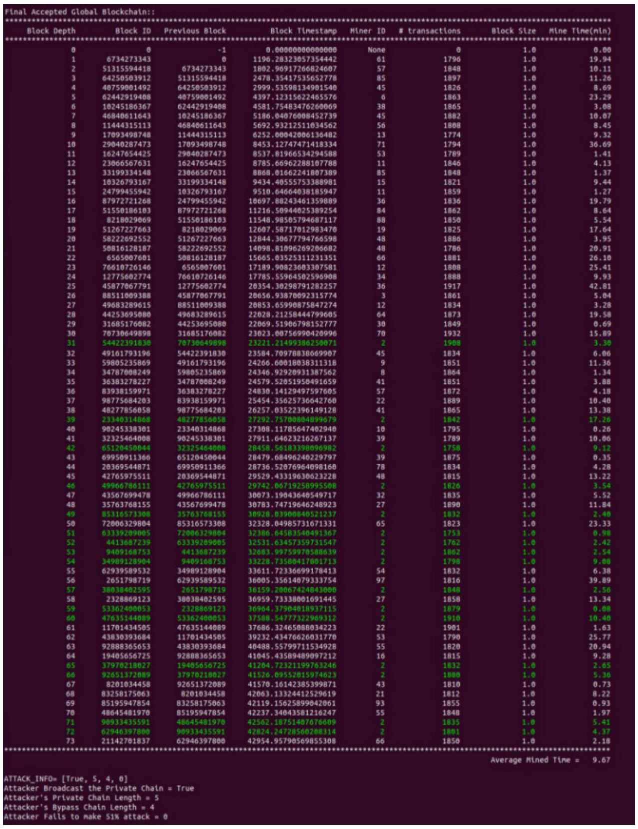


FIGURE 40. Without modified algorithm (SPC) – global chain with no double spending.

same input UTXO is observed within the first block transactions of the bypass chain, albeit bearing a distinct transaction ID of 42263748996. In this particular double-spending scenario, 1.33 bitcoins are directed to different recipient IDs.

Lastly, Figure 40 outlines the intricate details regarding the global chain's acceptance in the context of a short private chain (SPC), where no instances of double spending are observed and the potential for a 51% attack is mitigated.

V. DISCUSSION

The discussion, articulated by crypto expert Jameson Lopp (Crypto Expert), meticulously delineates the risk landscape inherent in Bitcoin transactions contingent upon confirmation levels. Lopp underscores the hazards of zero-confirm (0-conf) transactions, illuminating vulnerabilities to race attacks, Finney attacks, and 51% attacks. Through a pragmatic lens, he proposes a graduated framework for confirmation thresholds based on transaction values: 1 confirmation for modest transactions under $1,000, 3 confirmations for mid-range payments spanning $1,000 to $10,000, 6 confirmations for larger transactions ranging from $10,000 to $1,000,000, and finally, a recommended 10 confirmations for substantial payments surpassing $1,000,000. Lopp's expertise shapes a nuanced understanding of transaction security, encapsulating both theoretical vulnerabilities and practical strategies for risk mitigation within the Bitcoin ecosystem. (https://blog.lopp.net/how-many-bitcoin-confirmations-is-enough/)

The proposed methodology for mitigating 51% attacks on the Bitcoin network stands out in comparison to existing literature due to its innovative approach and comprehensive solution. Former approaches, as outlined in works such as [33], focus on analyzing the rational behavior of the miner and exploring game-theoretic models to understand the miner's strategic decisions. However, in comparison, this paper introduces a consensus algorithm and a safe mode detection mechanism, which are more practical and can be implemented in existing block chain technologies. In [34], the random mining group selection approach focuses on mitigating the risks associated with concentrated mining power by advocating for a more decentralized distribution of miners across mining pools. In contrast, the proposed methodology emphasizes algorithmic modifications and detection mechanisms within the existing Bitcoin network framework to detect and prevent 51% attacks in real-time. Finally, [8] provides a comprehensive assessment of various consensus mechanisms, including Proof of Work (PoW), Proof of Stake (PoS), and Delegated Proof of Stake (DPoS), among others. It evaluates these mechanisms based on their susceptibility to 51% attacks and other security vulnerabilities. Conversely, the proposed methodology in the present study focuses specifically on the Bitcoin network and introduces algorithmic modifications to the existing PoW consensus mechanism to prevent 51% attacks. The proposed algorithm, "Safe Mode Detection," operates independently of hash power constraints. In scenarios where a mining pool possesses over 51% of the hash power, the algorithm rejects any attempt to accept a private chain with a length equal to or greater than six blocks. Conversely, private chains with lengths less than six blocks undergo a comparison process with the honest blockchain (referred to as the Bypass chain). Despite the potential time complexity overhead incurred by this comparison due to shorter chain lengths, the algorithm effectively identifies instances of double spending within the private chain. Detected instances result in the rejection of the private chain, while absence leads to acceptance. The advantage of this restriction is paramount; it incentivizes other mining pools and miners to persist in their efforts to mine new blocks concurrently, thus maintaining a decentralized and competitive mining environment.

VI. CONCLUSION AND FUTURE WORK

In conclusion, our research represents a pivotal step in fortifying the security and integrity of blockchain networks, with a specific focus on mitigating the looming threat of 51% attacks within the Bitcoin ecosystem. We have proposed and rigorously tested a modified consensus algorithm that stands as a robust bulwark against such malicious endeavors.

The significance of our contributions lies in the establishment of a defense mechanism that consistently thwarts attackers' attempts to manipulate the network, thereby safeguarding major transactions and providing greater peace of mind to users [35], [36], [37]. By introducing stringent checks for double spending prior to accepting broadcasts of small private chains, we have raised the level of security for cryptocurrency transactions to new heights.

Furthermore, as part of our future work, we envision the development of even more dynamic and adaptive security mechanisms that can respond to emerging threats in real-time. This proactive approach will ensure that blockchain networks remain resilient in the face of evolving challenges.

This research not only enhances the resilience of individual miners but also anticipates the evolving landscape of blockchain mining, where collaborative efforts in pools are becoming increasingly prevalent. Our work lays the foundation for future exploration and adaptation, as it paves the way for further efficiency assessments and implementations within the expanding realm of blockchain technology.

Ultimately, our findings underscore the imperative of proactive measures to fortify the very foundations of decentralized systems, ensuring their robustness and trustworthiness in the face of potential threats. Through innovative research and steadfast dedication, we continue to drive advancements that bolster the security of blockchain networks, making them more resilient and reliable than ever before. By pursuing the avenues of dynamic security adjustments, quantum-resistant algorithms, and cross-blockchain security, we aim to keep blockchain technology at the forefront of secure and decentralized solutions for the future.

REFERENCES

[1] M. Ahmed and A. K. Pathan, "Blockchain: Can it be trusted?" *Computer*, vol. 53, no. 4, pp. 31–35, Apr. 2020, doi: 10.1109/MC.2019.2922950.

[2] R. S. Raju, S. Gurung, and P. Rai, "An overview of 51% attack over Bitcoin network," in *Contemporary Issues in Communication, Cloud and Big Data Analytics*, vol. 281. Cham, Switzerland: Springer, 2022, pp. 39–55, doi: 10.1007/978-981-16-4244-9_4.

[3] N. Kube, *Daniel Drescher: Blockchain Basics: A Non-Technical Introduction in 25 Steps*. New York, NY, USA: Apress, 2017.

[4] A. Bahalul Haque and M. Rahman, "Blockchain technology: Methodology, application and security issues," 2020, *arXiv:2012.13366*.

[5] M. Bosamia and D. Patel, "Current trends and future implementation possibilities of the Merkel tree," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 8, pp. 294–301, Aug. 2018, doi: 10.26438/ijcse/v6i8.294301.

[6] A. M. Antonopoulos and D. A. Harding, *Mastering Bitcoin*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, Inc., 2023.

[7] M. Saad, J. Spaulding, L. Njilla, C. Kamhoua, S. Shetty, D. Nyang, and A. Mohaisen, "Exploring the attack surface of blockchain: A systematic overview," 2019, *arXiv:1904.03487*.

[8] S. Sayeed and H. Marco-Gisbert, "Assessing blockchain consensus and security mechanisms against the 51% attack," *Appl. Sci.*, vol. 9, no. 9, p. 1788, Apr. 2019.

[9] M. Bastiaan, "Preventing the 51%-attack: A stochastic analysis of two phase proof of work in Bitcoin," in *Proc. 22nd student Conf.*, Jan. 2015, pp. 1–10.

[10] N. Tovanich, N. Soulié, and P. Isenberg, "Visual analytics of Bitcoin mining pool evolution: On the road toward stability?" in *Proc. 11th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Apr. 2021, pp. 1–5, doi: 10.1109/NTMS49979.2021.9432675.

[11] (2024). *Distribution of Bitcoin's Network Hashrate in the Last 24 Hours Until January 12, 2024*. Accessed: Oct. 09, 2021. [Online]. Available: https://www.statista.com/statistics/731416/market-share-of-mining-pools/

[12] C. Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda, "Analysis of security in blockchain: Case study in 51%-attack detecting," in *Proc. 5th Int. Conf. Dependable Syst. Their Appl. (DSA)*, Sep. 2018, pp. 15–24, doi: 10.1109/DSA.2018.00015.

[13] N. Anita. and M. Vijayalakshmi., "Blockchain security attack: A brief survey," in *Proc. 10th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2019, pp. 1–6, doi: 10.1109/ICCCNT45670.2019.8944615.

[14] J. B. Higuera, J. R. B. Higuer, J. A. S. Montalvo, and R. G. Crespo, *Introduction to Cryptography in Blockchain*. Cham, Switzerland: Springer, 2022, pp. 1–34.

[15] M. R. Amin, "51% attacks on blockchain: A solution architecture for blockchain to secure IoT with proof of work," Bachelor Thesis, Dept. Comput. Sci. Eng., Int. Univ. Bus. Agricult. Technol., Dhaka, Bangladesh, 2020.

[16] S. M. H. Bamakan, A. Motavali, and A. Babaei Bondarti, "A survey of blockchain consensus algorithms performance evaluation criteria," *Expert Syst. Appl.*, vol. 154, Sep. 2020, Art. no. 113385, doi: 10.1016/j.eswa.2020.113385.

[17] F. A. Aponte-Novoa, A. L. S. Orozco, R. Villanueva-Polanco, and P. Wightman, "The 51% attack on blockchains: A mining behavior study," *IEEE Access*, vol. 9, pp. 140549–140564, 2021, doi: 10.1109/ACCESS.2021.3119291.

[18] D. A. Bard, J. J. Kearney, and C. A. Perez-Delgado, "Quantum advantage on proof of work," *Array*, vol. 15, Sep. 2022, Art. no. 100225, doi: 10.1016/j.array.2022.100225.

[19] K. Jahnavi and G. Swain, "The blockchain technology and attacks on it," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 13, pp. 571–581, Jun. 2021.

[20] J. J. Kearney and C. A. Perez-Delgado, "Vulnerability of blockchain technologies to quantum attacks," *Array*, vol. 10, Jul. 2021, Art. no. 100065, doi: 10.1016/j.array.2021.100065.

[21] S. M. S. Saad and R. Z. R. M. Radzi, "Comparative review of the blockchain consensus algorithm between proof of stake (POS) and delegated proof of stake (DPOS)," *Int. J. Innov. Comput.*, vol. 10, no. 2, pp. 27–32, Nov. 2020.

[22] N. Shi, "A new proof-of-work mechanism for Bitcoin," *Financial Innov.*, vol. 2, no. 1, p. 31, Dec. 2016, doi: 10.1186/s40854-016-0045-6.

[23] K. Chaudhary, V. Chand, and A. Fehnker, "Double-spending analysis of Bitcoin," in *Proc. 24th Pacific Asia Conf. Inf. Syst., Inf. Syst.*, 2020, pp. 1–15.

[24] X. Yang, Y. Chen, and X. Chen, "Effective scheme against 51% attack on proof-of-work blockchain with history weighted information," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 261–265, doi: 10.1109/BLOCKCHAIN.2019.00041.

[25] J. Zheng, H. Huang, C. Li, Z. Zheng, and S. Guo, "Revisiting double-spending attacks on the Bitcoin blockchain: New findings," in *Proc. IEEE/ACM 29th Int. Symp. Qual. Service (IWQOS)*, Jun. 2021, pp. 1–6, doi: 10.1109/IWQOS52092.2021.9521306.

[26] B. Yu, X. Li, and H. Zhao, "PoW-BC: A PoW consensus protocol based on block compression," *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 4, pp. 1–15, Apr. 2021.

[27] N. Tovanich, N. Soulié, N. Heulot, and P. Isenberg, "An empirical analysis of pool hopping behavior in the Bitcoin blockchain," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC)*, May 2021, pp. 1–9, doi: 10.1109/ICBC51069.2021.9461118.

[28] M. Apostolaki, A. Zohar, and L. Vanbever, "Hijacking Bitcoin: Routing attacks on cryptocurrencies," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2017, pp. 375–392, doi: 10.1109/SP.2017.29.

[29] M. Saad, A. Anwar, S. Ravi, and D. Mohaisen. (2021). *HashSplit: Exploiting Bitcoin Asynchrony to Violate Common Prefix and Chain Quality*. Accessed: Jan. 27, 2024. [Online]. Available: https://eprint.iacr.org/2021/299

[30] M. Rosenfeld, "Analysis of hashrate-based double spending," 2014, *arXiv:1402.2009*.

[31] *Bitcoin Average Transactions Per Block*. Accessed: Feb. 28, 2010. [Online]. Available: https://ycharts.com/indicators/bitcoin_average_transactions_per_block

[32] *Average Input UTXOs in One Transaction Bitcoin—Google Search*. Accessed: Feb. 28, 2010. [Online]. Available: https://www.google.com/search?channel=fs&client=ubuntu&q=average+input+UTXOs+in+one+transaction+bitcoin

[33] C. Badertscher, Y. Lu, and V. Zikas, "A rational protocol treatment of 51% attacks," in *Proc. 41st Annu. Int. Cryptol. Conf.*, 2021, pp. 3–32.

[34] J. Bae and H. Lim, "Random mining group selection to prevent 51% attacks on Bitcoin," in *Proc. 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops*, Jun. 2018, pp. 81–82.

[35] M. Monem, M. T. Hossain, M. G. R. Alam, M. S. Munir, M. M. Rahman, S. A. AlQahtani, S. Almutlaq, and M. M. Hassan, "A sustainable Bitcoin blockchain network through introducing dynamic block size adjustment using predictive analytics," *Future Gener. Comput. Syst.*, vol. 153, pp. 12–26, Jun. 2024.

[36] C. W. Purnadi and S. Yazid, "Sidechain implementation strategies to improve blockchain scalability," in *Proc. AIP Conf.*, 2024, pp. 1–19.

[37] D. Aronoff and I. Ardis, "ADESS: A proof-of-work protocol to deter double-spend attacks," in *Proc. Future Inf. Commun. Conf.*, 2024, pp. 131–157.

SOHAIL MAHMOOD BABUR received the M.S. degree in computer science from the University of Sialkot, Pakistan. He is currently an Assistant Professor and the Head of the Department, Government Murray College, Sialkot, Pakistan. His research interests include blockchain technologies, the Internet of Things (IoT), machine learning, and computer vision.

SHAFIQ UR REHMAN KHAN received the Ph.D. degree from the Capital University of Science and Technology, Pakistan. He is currently an Assistant Professor with the Capital University of Science and Technology. He is also a collaborator between industry and academia. His research interests include natural language processing, machine learning, explainable AI, and blockchain technologies.

JING YANG (Graduate Student Member, IEEE) received the Bachelor of Engineering degree majoring in navigation technology from Shandong Jiaotong University in 2022, and the master's degree (Hons.) in data science from Universiti Malaya, Malaysia, in 2024, where he is currently pursuing the Ph.D. degree. His primary research interests lie in the fields of medical image processing, deep learning, the IoT, and blockchain.

YEN-LIN CHEN (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical and control engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2000 and 2006, respectively. From February 2007 to July 2009, he was an Assistant Professor with the Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan. From August 2009 to January 2012, he was an Assistant Professor with the Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan, where he was an Associate Professor, from February 2012 to July 2015, and since August 2015, he has been a Full Professor with the National Taipei University of Technology. His research interests include artificial intelligence, intelligent image analytics, embedded systems, pattern recognition, intelligent vehicles, and intelligent transportation systems. His research results have been published on over 100 journals and conference papers. He is a fellow of IET and a member of ACM, IAPR, and IEICE.

CHIN SOON KU received the Ph.D. degree from Universiti Malaya, Malaysia, in 2019. He is currently an Assistant Professor with the Department of Computer Science, Universiti Tunku Abdul Rahman, Malaysia. His research interests include AI techniques (such as genetic algorithm), computer vision, decision support tools, graphical authentication (authentication, picture-based password, and graphical password), machine learning, deep learning, speech processing, natural language processing, and unmanned logistics fleets.

LIP YEE POR (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees from Universiti Malaya, Malaysia. He currently holds the position of an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Malaya. His research interests include various aspects of information security and quality assurance (NEC 2020: 0611), including authentication, graphic passwords, PIN-entry, cryptography, data hiding, steganography, and watermarking. Additionally, he specializes in machine learning (NEC 2020: 0613), with expertise in extreme learning machines, support vector machines, deep learning, long-short-term memory, computer vision, and AIoT.