GeoBlockchain: Geolocation Based Consensus Against 51% Attacks

Franco Moloche-Garcia, Pedro Bustamante-Castro and Willy Ugarte[a]
Universidad Peruana de Ciencias Aplicadas, Lima, Peru

Keywords: Blockchain, Consensus Algorithm, 51% Attacks, Geolocation, Mining Pools.

Abstract: Currently, Blockchain technology has been involved in various areas such as medicine, environment, finance, mining, etc., therefore, the rise of technology causes an increasing use among technology companies, developers and even malicious attackers. The latter, through 51% attacks, could have the ability to manipulate a Blockchain network. The present proposal consists of a consensus algorithm for Blockchain networks based on Proof-of-work (PoW) that, through the use of geolocation, aims to provide protection against 51% attacks. With enough computational power on one of these Blockchain networks, an attacker could reverse transactions and identification might be impossible. It is the mining pools, which together, could have the capacity to carry out these attacks. Using geolocation, it is intended to minimize the probability of generating mining pools, making their formation unfavorable. In our experiments, the algorithm reduces the probability of 51% attacks by an average of 29% compared to PoW, providing a new layer of protection when generating consensus between participating Blockchain nodes.

1 INTRODUCTION

Due to the high popularity of Blockchain and the rise of its use due to various factors such as cryptocurrencies, this growth could represent a danger for this technology since, as there are more interested in using and implementing it, there are also greater desires to attack and violate it. For example, according to Forbes[1] there are currently around 20,000 cryptocurrency projects and where there are around 295 million users in total. If we consider this number of users, we could confirm the great popularity of cryptocurrencies today, but also the impact of being attacked. Since the nodes are the ones that support the network, many of them join in what are called mining pools.

From Figure 1, putting together 4 mining pools is enough to meet the computational power needed to carry out 51% attacks. It is something that should be of concern, however, it is still believed that these groups would not attack the blockchain since they themselves would be harmed. Even so, it is a possibility for which there is no contingency. The use of these mining pools leads to the problem of this research: The 51% attack. Currently this type of attack has not occurred, however, with the passage of time and the growth of these mining pools, they could be carried out with serious damage to cryptocurrencies and other systems that use Blockchain.

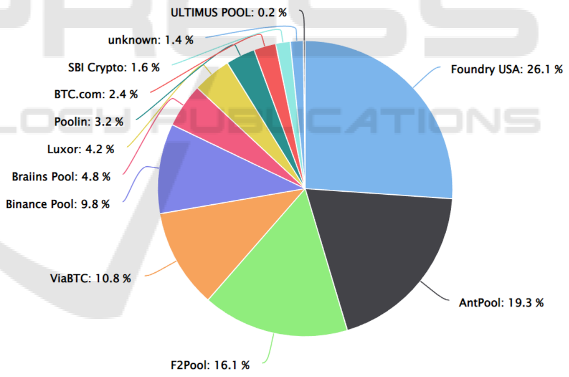


Figure 1: Bitcoin mining pools[2].

A Blockchain is made up of blocks that are added by mining them. Nodes can participate in the mining process and only selected ones are used for validation (Zheng et al., 2018). This mining process is called Proof of Work (PoW) and it is the main cause that could trigger the 51% attack. This attack consists of several malicious nodes joining together so that be-

[a] https://orcid.org/0000-0002-7510-618X

[1] "10 Best Crypto Exchanges Of 2022" - Forbes - https://www.forbes.com/advisor/investing/cryptocurrency/best-crypto-exchanges/

[2] https://btc.com/stats/pool

GeoBlockchain: Geolocation Based Consensus Against 51 .
DOI: 10.5220/0011996600003467
In *Proceedings of the 25th International Conference on Enterprise Information Systems (ICEIS 2023) - Volume 1*, pages 255-262
ISBN: 978-989-758-648-4; ISSN: 2184-4992

ing the majority they can attack the Blockchain. To achieve this, these nodes together must exceed 50% of the total number of nodes used by the Blockchain in order to violate it. If one of these attacks were to happen on, for example, a cryptocurrency such as Bitcoin, fraud or money loss could be generated by different users within the network. Solving this type of problem is complicated because the larger the mining pools, the greater the probability that they will come together to generate the attack. On the other hand, simulating this type of attack is also complicated due to the need to have several nodes available to be able to attack the Blockchain, consuming a lot of computational and economic resources. The use of PoW as a consensus algorithm has been an advantage in the adoption of blockchain technology. However, we also consider it a limitation as it is an energy-intensive solution that promotes centralization. With the development of this work we proposed a decentralized solution based on the use of geographic validations, which generates a more efficient and secure system.

According to Satoshi Nakamoto in the original Bitcoin paper: The system is secure as long as the honest nodes collectively control more CPU power than the attacking nodes (Nakamoto, 2009). Therefore, with the current popularity of Blockchain, this type of attack could be possible and harmful, especially in networks with few nodes. To make a Blockchain network more resistant to this problem, we have modified the PoW consensus algorithm to add geographical restrictions. The project is limited to the simulation of coordinates (longitude and latitude) given an IP, due to the complexity involved in implementing a technology such as Octant. The hash rate represents the computational power that is used in a blockchain for mining. Having more than 50% of the hash rate in a blockchain allows its owner to carry out dishonest actions. With the use of geographical validations, it is possible to proportionally and impartially discriminate said hash rate and its influence and to increase the security and trust in the network, as well as reducing the centralization of power.

The use of geographical validations can help to create a more secure and trust blockchain network by reducing the centralization of power. This validation process can help to ensure that no one entity has control over the majority of the hash rate, and that all users are able to participate in the network fairly. Since this vulnerability (51% of attacks) could go unnoticed, it is difficult to have a metric. For this project we counted the ownership percentage of the last 100 blocks mined by dishonest nodes. This metric represents the distribution of power in the network at a given time and can be used as a detection system for these types of attacks.

Our main contributions are the following:

- We have developed a consensus algorithm with geographical validations on a GeoBlockchain
- We have developed an unpredictable map generator of geographic zones for the restrictions and validations of the GeoBlockchain.
- We have proposed a metric for the comparison of the probability of attacks of 51

This paper is organized as follows. Similar solutions currently implemented in the literature will be explained in Section 2. In Section 3, the technologies related to our proposal will be explained, such as: Blockchain, consensus algorithm and 51% attacks. Finally, Section 4 will detail the experiments and their results. To conclude with Section 5

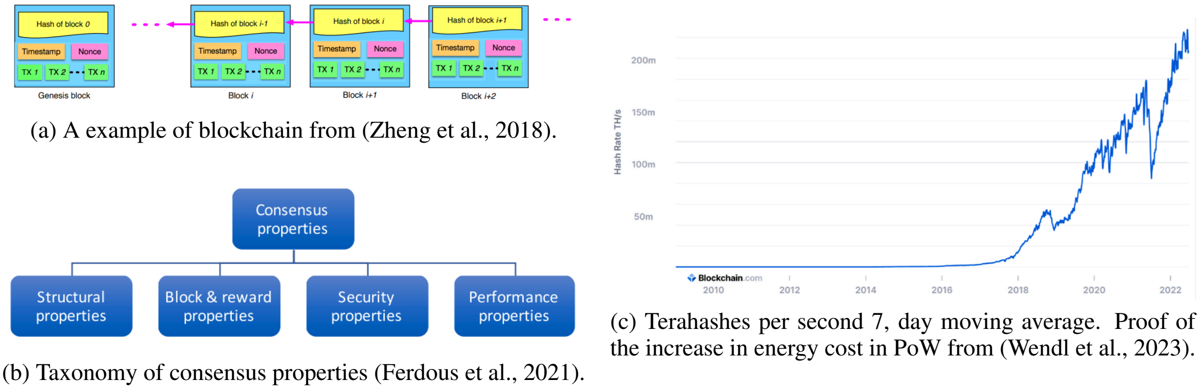
2 RELATED WORKS

Now, the different existing solutions using various consensus mechanisms for the blockchain protocol will be briefly discussed. It is worth mentioning that to the best of our knowledge, none of the related works use geographical validations and do not necessarily seek to protect themselves from 51% attacks.

In (Nakamoto, 2009), part of the author's proposal is the called PoW algorithm, it is intended to be a source of truth where only the largest chain will be the chosen one. The blockchain is secure if honest nodes have most of the participation (Yan, 2022). Also, the protocol makes the attackers less favored if they have enough power to support the chain and get more rewards. Instead, in our work we have mentioned that this process is based on trusting potential attackers and can lead to a 51% attack and we use geographical validation, restricting an attacker to be enough distributed around the globe to achieve this attack.

Another related work is ReCon (Sybil-resistant consensus) (Biryukov and Feher, 2020), a consensus mechanism that takes external reputation ranking as input. The consensus can tolerate larger threshold of malicious nodes compared to another's. The main difference with our work is that ReCon is aimed at protecting the blockchain from sybil attacks, while ours seeks to add protection against 51% attacks. This type of attack is known to require a significant number of nodes in the network, regardless of the computational power of the attacker. This proposal could be integrated with the present project with the intention of having a more protected network.

Proof-of-Activity is a fork-free hybrid consensus algorithm (Liu et al., 2019). It is based on PoW and



(a) A example of blockchain from (Zheng et al., 2018).

(b) Taxonomy of consensus properties (Ferdous et al., 2021).

(c) Terahashes per second 7, day moving average. Proof of the increase in energy cost in PoW from (Wendl et al., 2023).

Figure 2: Examples of blockchain and the taxonomy of consensus.

Proof-of-Stake (PoS). Initially a node can use PoW to mine a block consisting of meta information, which is then used to choose a set of validators via PoS (Ferdous et al., 2021). The algorithm is characterized by protecting itself from selfish mining attacks and has a fairer committee of validators. As with ReCon, this proposal can be implemented with our work. However, our project is also indirectly protected from selfish mining attacks, as it could not be done without most of the computing power on the network.

In (Xue et al., 2018), they propose Proof-of-Contribution (PoC), a consensus algorithm that has lower power consumption than PoW. This consensus, in addition to rewarding the node that managed to mine a block, also rewards the difficulty exerted by the other miners who did not win the block. Another variant is Delegated-Proof-of-Contribution (DPoC) (Song et al., 2021), in which the main contribution is the support of intellectual property. Unlike these proposals, our project is not intended to reduce energy consumption directly. However, the fact of blocking a portion of the miners leads to lower energy consumption indirectly.

3 MAIN CONTRIBUTION

3.1 Preliminary Concepts

In the blockchain protocol, the consensus algorithm is involved, which is the main support to maintain the correct chain. Specifically the proof-of-work algorithm is involved in issues like mining pools and 51% attacks. Approaches adapted by other research on these terms are presented below.

3.1.1 Blockchain

According to Z. Zheng et al. (Qiao et al., 2021), blockchain is the key technology in the digitization of cryptocurrency systems such as bitcoin. Within the development of this technology, its consensus mechanism and distributed storage technology are involved. In addition, it provides an effective solution to trust problems in open networks and the security of data storage in centralized institutions. Figure 2a depicts an example of blockchain (Zheng et al., 2018).

3.1.2 Consensus Algorithm

Figure 2b depicts the taxonomy of consensus properties (Ferdous et al., 2021).

Proof-of-Work (PoW): Being one of the first consensus algorithms, its main operation lies in what is defined as mining (Qiao et al., 2021). In order for a node to add a block in this system, it must first solve a puzzle that has a dynamic difficulty that depends on the network computational power. The node that manages to resolve and distribute the fastest gets a reward. Bitcoin is credited with an environmental impact whose annual emissions in 2021 are responsible for around 19,000 future deaths (Truby et al., 2022).

Figure 2c depicts the quantity of Terahashes per second to prove the increase in energy cost in PoW (Wendl et al., 2023). Although an attack on large blockchain networks such as Bitcoin has yet to be orchestrated, the likelihood of one happening in the near future is increasingly high. This is mainly due to the growth that blockchain is having due to the arrival of Web3, decentralized applications (dapps) and the growing interest in mining pools. However, there are currently very small blockchain applications that can be quickly compromised with the 51% attack or immutability attack if the neces-

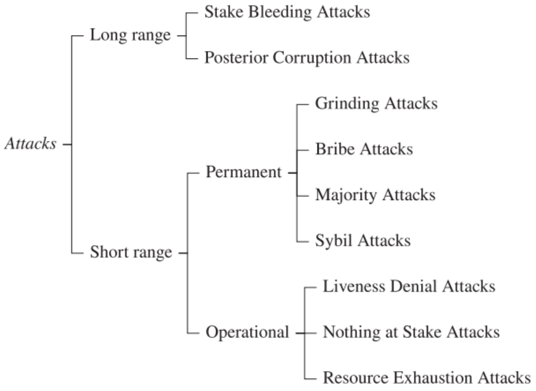


Figure 3: Categorisation of attacks on identity-augmented Proof-of-Stake systems from (Platt and McBurney, 2021).

sary security measures are not considered. Regional blockchains are created to be able to create small decentralized networks according to the region where it is located (Shrestha and Nam, 2019). The relationship between regional blockchain networks and the 51% attack are more likely, therefore, this research could be perfectly suited to this type of blockchain today.

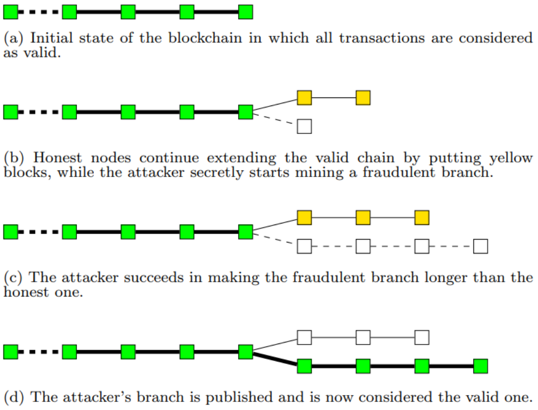
Proof-of-Stake (PoS): This algorithm, unlike PoW, does not require mining, therefore, its environmental impact is less. Its operation is to randomly choose a node to validate the next block, to be eligible as a validator it is necessary to deposit a number of coins in the network (Qiao et al., 2021). The more coins are deposited, the chances of being chosen increase, this being its main disadvantage since it promotes centralization. Below is a diagram of vulnerabilities in PoS. Figure 3 depicts the categorisation of attacks on identity-augmented Proof-of-Stake systems (Platt and McBurney, 2021).

3.1.3 Mining Pool

Blockchains have a degree of decentralization, however, it is possible that the nodes decide to join together to form what is known as a mining pool. This is due to the fact that the rewards end up being distributed in the network, this favors nodes with little computational capacity, increasing the profits they could obtain. Unfortunately, this represents a security risk since gathering enough computational power makes it possible to perform dishonest or fraudulent actions (Zheng et al., 2018).

3.1.4 51 % Attack

Computational Power: The hash rate is the number of hashes that a computer can calculate per second (Kausar et al., 2022). With a sufficient hash rate, fraudulent attacks and actions can be carried out on the network, such as 51% attacks. The only protection you have against this “power” is to trust that someone who has this ability is honest, since, otherwise, he would disadvantage himself (Nakamoto, 2009).


(a) Initial state of the blockchain in which all transactions are considered as valid.

(b) Honest nodes continue extending the valid chain by putting yellow blocks, while the attacker secretly starts mining a fraudulent branch.

(c) The attacker succeeds in making the fraudulent branch longer than the honest one.

(d) The attacker’s branch is published and is now considered the valid one.

Figure 4: Example of the double spending problem from (Pinzón and Rocha, 2016), a problem that can be done through the 51%.

Accessibility: Although reaching sufficient computing power for this type of attack is directly complicated, it is possible to achieve it through other such attacks such as: Sybil attacks, DNS attacks, BGP hijacking and spatial partitioning, double spending attacks, Finney attacks, Selfish mining attacks, etc (Kausar et al., 2022). Also, it opens the doors to other attacks like DDoS. Figure 4 depicts an example of the double spending problem from (Pinzón and Rocha, 2016).

3.2 Method

For the development of a consensus algorithm capable of disfavoring the formation of mining pools, we proposed a dynamic mining restriction in geographical areas to the proof-of-work algorithm. Zones which are randomly generated based on the hash of the previous block, this allows nodes to verify if an incoming block comes from a node in a valid mining zone.

3.2.1 Generation of Mining Zones

Due to the need for each node in the network to have access to a canvas with valid mining areas. It was proposed to use a seed-based gradient noise random number generator, since using a seed in this type of method allows all nodes to generate the same canvas. This allows to generate a 2-dimensional canvas in which the axes represent the longitude and latitude of the planet. Using the hash of the previous block as a seed, it is possible for any participant in the network that wishes to mine to generate such a canvas and verify if its location (longitude and latitude) is in a valid mining area. If it is in a valid area, then the mining process would continue; otherwise it would be in standby mode.

In Algorithm 20, in the 1st and 2nd lines the configuration for the generation of random zones is declared. In the 3rd line, the seed is declared by adding the characters of the hash of the previous block. In the 4th line, an array is initialized with the dimensions corresponding to the available longitudes and latitudes (rounded). On line 5, the 2-dimensional random number generator is initialized with the OpenSimplex algorithm. In line 6 and 8 the offsets for the generation of random numbers are initialized. Between lines 7 and 19 the longitudes and latitudes to be evaluated are iterated. In line 10 the random number n is generated based on the offsets. Between lines 11 and 15, if n is greater than the gap, True is assigned to that zone, otherwise, False is assigned. In line 16 and 18 the offsets are increased respectively by the noiseScale. Finally, on line 20 the canvas is returned with the valid mining regions set to True. For the development of this mining zone generator, a gradient noise function called OpenSimplex was used. Specifically version 0.7.0 implemented in Rust. Up to 4 dimensions are allowed, however only 2 are needed.

```
   Data: Previous block hash
   Result: Boolean matrix with valid mining
           zones by longitude and latitude
1  $noiseScale \leftarrow 0.03$;
2  $gap \leftarrow 0.2$;
3  $seed \leftarrow$ SumChars($prevHash$);
4  $canvas \leftarrow$ Matrix(180,360);
5  $Generator \leftarrow$ OpenSimplex($seed$);
6  $xoff \leftarrow 0$;
7  for $x \leftarrow 0$ to 360 do
8      $yoff \leftarrow 0$;
9      for $y \leftarrow 0$ to 180 do
10         $n \leftarrow Generator([xoff, yoff])$;
11         if $n > gap$ then
12             $canvas[y,x] \leftarrow True$;
13         else
14             $canvas[y,x] \leftarrow False$;
15         end
16         $yoff \leftarrow yoff +$noiseScale;
17     end
18     $xoff \leftarrow xoff +$noiseScale;
19 end
20 return canvas;
```

Algorithm 1: Mining Zones Generator.

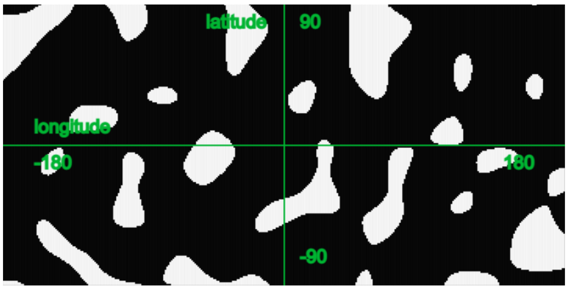


Figure 5: Canvas with blank valid mining areas generated with 42 as a seed.

Figure 5 depicts a canvas with blank valid mining areas generated with 42 as a seed. The use of a seed-based random number generator was chosen to take advantage of the hash of the previous blocks. However, it was also considered to generate gradient random numbers, since mining zones will be more similar to regions and give nodes more chances of being inside one.

3.2.2 Consensus Algorithm

For the proposed consensus algorithm, it is taken into account that it is capable of doing the following: generate unbiased mining zones, restrict mining in said zones and validate that the nodes comply with the restrictions. On line 1 of our consensus algorithm, we get g, which represents if a node is in a mining zone (based on the previous block hash and the node's IP). In line 3, if you are not in a valid mining zone and if you have not exceeded the tolerance limit in line 2, the mining process is stopped. In Algorithm 9, from lines 4 to 9, the traditional PoW algorithm is continued. The situation where a node tries to mine before the tolerance time is not in scope. Furthermore, it is not considered to reliably validate the location of a node due to the complexity involved in developing it.

3.2.3 Smart Contracts

Additionally, the blockchain was configured to support smart contracts. This is possible thanks to the fact that Substrate, the framework used, supports integration with this technology. Added Substrate Smart Contracts module. This module is in charge of being able to enable Smart Contracts in the Blockchain. For this, the Substrate library called *pallet-contracts* had to be added. Smart Contracts are developed using the Ink! language. This Rust-based language allows you to create Smart Contracts in a simplified way and using the advantages of Rust such as memory optimization. For the validation of the blockchain, a Smart

```
Data: Block info
Result: Nonce that solves the mining puzzle
$g \leftarrow$ IsOnMiningZone($prevHash, IP$);
if not $g$ and $currentTime < timestamp + tolerance$ then
    return NULL
$b \leftarrow$ SHA256HashFunction($txRoot, timestamp, prevHash, IP$);
$nonce \leftarrow 0$;
while SHA256HashFunction ($b, nonce$) $< targetDifficylty$ do
    $nonce \leftarrow nonce + 1$;
end
return $nonce$
```

Algorithm 2: Consensus algorithm for GeoBlockchain inpired by PoW from (Juriˇci´c et al., 2020).

Contract oriented to clinical records was developed. In this Smart Contracts patients are stored along with their medical records. Substrate has a tool to be able to develop Smart Contracts called Contracts UI that allows you to add the Smart Contract to a Blockchain with the pallet-contracts module. This UI made it easy for us to deploy the Smart Contract and test it during development.

Once the blockchain is configured with the modified consensus algorithm, it is compiled using *cargo build* command and the generated compilation is executed. This compilation lifts the genesis block of the blockchain, lifts the endpoints so that clients connect to the blockchain and can consume data through RPC API, and also enables the websocket that allows communication with a telemetry server.

4 EXPERIMENTS

In this section we will discuss the experiments our project has undergone, as well as what is needed to replicate said experiments and a discussion of the results obtained after this process.

4.1 Experimental Protocol

To recreate the process of building, deploying, and testing the blockchain used in our project, we start by describing what it took to accomplish that task.

4.1.1 Development Environment

Being a blockchain, the development environment is variable, 3 devices were used that ran Linux either directly or through WSL2. The blockchain binary weighs around 3 4 GB, however, a minimum of 16 GB of RAM was required for development.

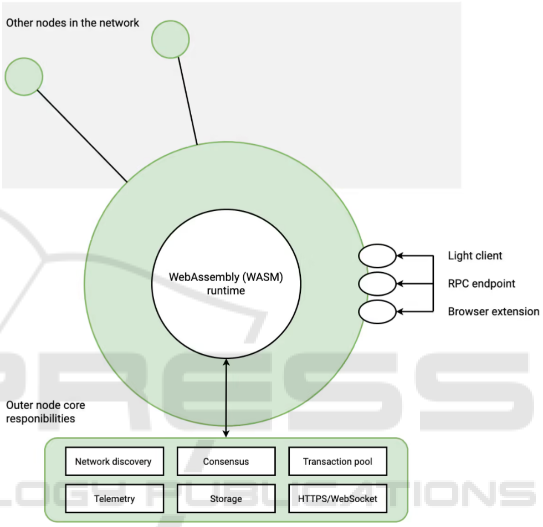


Figure 6: Substrate architecture from Substrate.io.

4.1.2 Blockchain

The blockchain was developed based on the Substrate SDK, a framework designed for the creation of blockchains. Substrate has a template called substrate-node-template which was used as the basis for developing the blockchain.

Figure 6 depicts the substrate architecture from Substrate.io. This default template has the PoS consensus algorithm. The PoW consensus algorithm implemented in substrate was modified to transform it into the one proposed for GeoBlockchain. In addition, a library with OpenSimplex support in rust was used for the generation of regions. The blockchain was configured with the library *sp-consensus-pow* and *sc-consensus-pow* from Substrate and updated to the latest version to ensure compatibility with the same framework and other libraries from Substrate and Rust. The update to the latest version was necessary because Substrate is a new framework which does not have much documentation and the previous pre-release versions deprecate quickly. After the configuration of the blockchain, a mining script was added for the nodes. The mining algorithm invokes the algorithm proposed in this research when calling the worker's submit method. This algorithm has been added as one more library to order the logic that we have from the independent blockchain to the proposed algorithm. This library is implemented in the main blockchain and is called every time the mining process is carried out.

4.1.3 Testing Environment

Docker was used to test the blockchain, creating virtual nodes simulating a computer. The containers can be configured as it would be with a real computer in such a way that each one can have assigned amount of resources, IP address, etc. The use of containers was necessary to be able to simulate a blockchain due to lack of resources to be able to have several computers and carry out a real simulation. After generating the blockchain binaries, an image was created that was uploaded to different devices to simulate a network. Substrate has a telemetry application where connected nodes, blocks, etc. are displayed. When orchestrating the attack in this application, it is shown that the dishonest nodes are the ones that take the lead in the blockchain, even propagating the blocks that they are mining. Showing that the dishonest nodes are the ones that have priority because they have greater computational power than the honest ones. Thanks to this, dishonest nodes have a greater chance of mining the block belonging to a mining area. Honest nodes at times also manage to propagate blocks, however, the trend is the blocks of dishonest nodes.

4.1.4 Source Code

Our source code for building the blockchain is publicly available at https://github.com/magpex13/geoblockchain.

4.2 Results

Eight nodes were deployed for the orchestration of the 51% attack. For this, 4 containers were used for the honest nodes and 4 for the dishonest ones. To simulate a 51% attack, the computational power of the nodes was distributed 80% to the dishonest ones and 20% to the honest ones. For this, the percentage of CPU per container was configured in such a way that the dishonest ones have greater computational power.

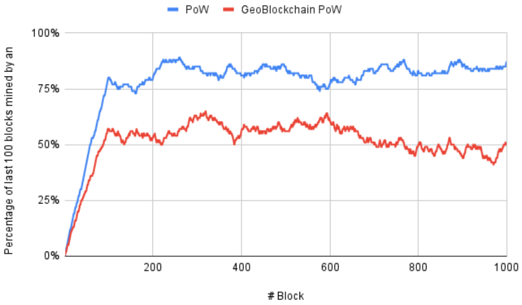


Figure 7: Percentage of the last 100 nodes that were mined by an attacker.

Table 1: Average of the last 100 blocks mined by dishonest nodes.

Consensus	Percentage of blocks mined by dishonest nodes
PoW	94.10%
GeoBlockchain PoW	64.70%
Difference	29.40%

Figure 7 depicts the percentage of the last 100 nodes that were mined by an attacker (PoW in blue and ours in red). Considering that the probability that a node mines a block in PoW depends on its computational power, for our experiment this situation was conditioned 19.63% of the time. Since, generating mining areas with a seed of 42 to 420, on average that percentage of valid areas is obtained. In addition, the computational power of the attacking nodes was set to 80%, while the remaining 20% went to the honest nodes. The results shown in the Table 1 indicate that, indeed, the probability of a 51% attack can decrease depending on the different configurations of the mining zones generator. On average, PoW had a 94.10% probability of attacks of 51%, compared to ours, which obtained 64.7%. A difference of 29.40%. As can be seen, the probability of a 51% attack on a PoW blockchain is much higher.

4.3 Discussion

As presented above, the results of our experiments indicate that the probability of an attack varies with different settings of our algorithm. In our opinion, this is a very important aspect to take into account when designing a blockchain. However, it is important to keep in mind that in case no nodes are found in valid mining zones (80.37% of the times) the block will be mined by the one that managed to distribute it faster after the tolerance time. Our proposal has been designed to achieve a higher degree of security in the network, since a malicious node would need to be able to predict the area of a valid mine in advance, and be able to deliver the block to the network faster than the other nodes. The probability of a successful attack is proportional to the number of miners that are in the same mining zone at the same time, since in our system the mining zone is generated randomly. Therefore, the probability of an attack is greater the greater the number of miners in valid zones from which they cannot be predicted. We propose to decrease the size of the mining area to decrease the probability of an attack, however we consider it important to keep in mind that if the size of the mining areas is too small, the block will take longer to be distributed.

5 CONCLUSION

We conclude that adding geographical validations to the PoW algorithm can add a 51% layer of protection against attacks. Despite not prioritizing the most optimal configurations, 29% protection was achieved with our consensus algorithm compared to PoW using our metric of the average percentage of the last 100 blocks mined by attacking nodes.

We have analyzed the 51% attack on a blockchain network and its possible countermeasures. The attack is a serious problem in the world of cryptocurrencies, as it allows the attacker to manipulate transactions and even block them, which could lead to double spending. This shows how important it is to have protection against this type of attack and the advantages that the use of geographical validations offers. Considering that the verification of locations was simulated due to the complexity of its implementation, it is possible to develop a system that, through a network, allows knowing the geographical location of a node with solutions such as Octant. A framework with which the geographical position of a node can be determined with great confidence, simply by measuring its latency with reference points (Wong et al., 2007).

In a future work we will seek to implement a geographic location validator, since our consensus algorithm continues with PoW as there are no valid nodes in the mining zones, and try it with different kinds of data such as healthcare data (Arroyo-Mariños et al., 2021) or Wood supply chain (Cueva-Sánchez et al., 2020). With this validator it is possible to prioritize the distance from a node to a mining area, which promotes lower energy expenditure.

REFERENCES

Arroyo-Mariños, J. C., Mejia-Valle, K. M., and Ugarte, W. (2021). Technological model for the protection of genetic information using blockchain technology in the private health sector. In *ICT4AWE*.

Biryukov, A. and Feher, D. (2020). Recon: Sybil-resistant consensus from reputation. *Perva. Mob. Comput.*, 61.

Cueva-Sánchez, J. J., Coyco-Ordemar, A. J., and Ugarte, W. (2020). A blockchain-based technological solution to ensure data transparency of the wood supply chain. In *IEEE ANDESCON*.

Ferdous, M. S., Chowdhury, M. J. M., and Hoque, M. A. (2021). A survey of consensus algorithms in public blockchain systems for crypto-currencies. *J. Netw. Comput. Appl.*, 182.

Juričić, V., Radošević, M., and Fuzul, E. (2020). Optimizing the resource consumption of blockchain technology in business systems. *Busi. Syst. Res. J.*, 11(3).

Kausar, F., Senan, F. M., Asif, H. M., and Raahemifar, K. (2022). 6g technology and taxonomy of attacks on blockchain technology. *Alexandria Eng. J.*, 61(6).

Liu, Z., Tang, S., Chow, S. S. M., Liu, Z., and Long, Y. (2019). Fork-free hybrid consensus with flexible proof-of-activity. *Future Gener. Comput. Syst.*, 96.

Nakamoto, S. (2009). Bitcoin: A peer-to-peer electronic cash system.

Pinzón, C. and Rocha, C. (2016). Double-spend attack models with time advantange for bitcoin. *Electronic Notes in Theoretical Computer Science*, 329.

Platt, M. and McBurney, P. (2021). Sybil attacks on identity-augmented proof-of-stake. *Computer Networks*, 199.

Qiao, L., Dang, S., Shihada, B., Alouini, M.-S., Nowak, R., and Lv, Z. (2021). Can blockchain link the future? *Digital Communications and Networks*.

Shrestha, R. and Nam, S. Y. (2019). Regional blockchain for vehicular networks to prevent 51% attacks. *IEEE Access*, 7.

Song, H., Zhu, N., Xue, R., He, J., Zhang, K., and Wang, J. (2021). Proof-of-contribution consensus mechanism for blockchain and its application in intellectual property protection. *Inf. Process. Manag.*, 58(3).

Truby, J., Brown, R. D., Dahdal, A., and Ibrahim, I. (2022). Blockchain, climate damage, and death: Policy interventions to reduce the carbon emissions, mortality, and net-zero implications of non-fungible tokens and bitcoin. *Energy Research & Social Science*, 88.

Wendl, M., Doan, M. H., and Sassen, R. (2023). The environmental impact of cryptocurrencies using proof of work and proof of stake consensus algorithms: A systematic review. *J. of Env. Management*, 326.

Wong, B., Stoyanov, I., and Sirer, E. G. (2007). Octant: A comprehensive framework for the geolocalization of internet hosts. In *NSDI*. USENIX.

Xue, T., Yuan, Y., Ahmed, Z., Moniz, K., Cao, G., and Wang, C. (2018). Proof of contribution: A modification of proof of work to increase mining efficiency. In *IEEE COMPSAC*.

Yan, S. (2022). Analysis on blockchain consensus mechanism based on proof of work and proof of stake. *CoRR*, abs/2209.11545.

Zheng, Z., Xie, S., Dai, H., Chen, X., and Wang, H. (2018). Blockchain challenges and opportunities: a survey. *Int. J. Web Grid Serv.*, 14(4).