**Abstract**

We have written this paper on “Prevention of Pool Hopping using Smart Contracts, and we described the solutions for which the miners jump from one pool to the other which leads to network stability. So, we described everything in this paper from the point when miner enters the mining pool up to when the miner leaves the pool.

In the process of onboarding miners to the network, a critical step involves verifying their miner certification. Each miner possesses a unique miner address associated with their individual certificate, which is necessary for receiving rewards. The pool manager, while respecting the miner's privacy, accesses this certificate by requesting the miner's address. This certificate details the miner's history, particularly focusing on their Hop-Count, past smart contract (SC) upholds, and SC violations.

To assess a miner's risk to the mining pool, both the Hop-Count (α) and SC-violated (μ) components are evaluated. When both values are zero, the miner is considered safe for inclusion in the pool. However, if either or both values are greater than zero, indicating a potential risk due to hopping behavior, a smart contract is initiated. This contract requires miners with a high Hop-Count to submit coins as an escrow, acting as a penalty if they prematurely leave the mining pool.

The escrow serves as a deterrent, ensuring miners remain committed to completing their mining tasks. The terms of the smart contract mandate the escrow requirement, with the specific amount determined based on the miner's history of upholding or violating prior smart contracts. Miners with a reliable history may need a smaller escrow, reflecting their trustworthiness. This approach effectively discourages pool hopping behavior and promotes a fair and secure mining environment.

**Keywords:** Smart contract, proof-of-work, pool hopping, accountability, Hop count, SC-upheld, SC-violated

1. **Introduction**

A **blockchain** is a [distributed ledger](https://en.wikipedia.org/wiki/Distributed_ledger) with growing lists of [records](https://en.wikipedia.org/wiki/Record_(computer_science)) (*blocks*) that are securely linked together via [cryptographic hashes](https://en.wikipedia.org/wiki/Cryptographic_hash_function). Each block contains a cryptographic hash of the previous block, a [timestamp](https://en.wikipedia.org/wiki/Trusted_timestamping), and transaction data (generally represented as a [Merkle tree](https://en.wikipedia.org/wiki/Merkle_tree), where [data nodes](https://en.wikipedia.org/wiki/Node_(computer_science)) are represented by leaves). The timestamp proves that the transaction data existed when the block was created. Since each block contains information about the previous block, they effectively form a *chain* (compare [linked list](https://en.wikipedia.org/wiki/Linked_list) data structure), with each additional block linking to the ones before it. Consequently, blockchain transactions are irreversible in that, once they are recorded, the data in any given block cannot be altered retroactively without altering all subsequent blocks.

Blockchains are typically managed by a [peer-to-peer (P2P)](https://en.wikipedia.org/wiki/Peer-to-peer) computer network for use as a public [distributed ledger](https://en.wikipedia.org/wiki/Distributed_ledger), where nodes collectively adhere to a [consensus algorithm](https://en.wikipedia.org/wiki/Consensus_algorithm) [protocol](https://en.wikipedia.org/wiki/Communication_protocol) to add and validate new transaction blocks. Although blockchain records are not unalterable, since [blockchain forks](https://en.wikipedia.org/wiki/Fork_(blockchain)) are possible, blockchains may be considered [secure by design](https://en.wikipedia.org/wiki/Secure_by_design) and exemplify a distributed computing system with high [Byzantine fault tolerance](https://en.wikipedia.org/wiki/Byzantine_fault_tolerance).

Mining is the process that Bitcoin and several other cryptocurrencies use to generate new coins and verify new transactions. It involves vast, decentralized networks of computers around the world that verify and secure blockchains – the virtual ledgers that document cryptocurrency transactions. In return for contributing their processing power, computers on the network are rewarded with new coins.

Pool-hopping is the practice of mining in a pool only during the good times, and leaving during the bad times; by so doing, a pool-hopper can get more out of the pool than the value they contribute to it, increasing their rewards at the expense of other miners. Pool-hopping gets its name from the act of constantly hopping into and out of the pool (to either other pools or solo mining)

In this paper, we make the following contributions:

* We propose methods to avoid pool hopping.
* We discuss Pool hopping attacks and existing studies to mitigate these attacks are described in detail.
* The methods to prevent pool hopping attacks, we describe a smart contract-based pool hopping attack prevention model.
* A detailed numerical equation is provided to evaluate the miner risk to the mining pool and how to calculate the escrow amount a miner will submit as part of the smart contract agreement.
* Three key consideration factors, computational power, miner risk, and accountability, are analysed to demonstrate the effectiveness of our proposed model.
* A comparative analysis with existing solutions and the proposed model is presented.
* Future directions of the research are discussed.

1. **Literature survey**

A blockchain is a decentralized digital ledger that records transactions across many computers. These transactions are grouped into blocks and each block is linked to the previous one, forming a chain of blocks (hence the name “blockchain”). In the context of cryptocurrencies like Bitcoin, transactions are verified by a network of users called “miners” who use their computational power to solve complex mathematical problems. When a miner successfully solves a problem and verifies a block of transactions, they are rewarded with a certain amount of cryptocurrency. Miners can join together to form “mining pools” where they combine their computational power to increase their chances of successfully verifying a block and earning the reward. However, some miners engage in “pool hopping” where they switch between pools to maximize their rewards. This can be harmful to the mining pool as it deprives it of its collective hash power and leaves it unable to mine the block successfully.

*Different solutions proposed to prevent pool-hopping:*

The paper “Smart Contract-Based Pool Hopping Attack Prevention for Blockchain Networks” proposes a solution to this problem by using smart contracts [1]. A smart contract is a self-executing contract with the terms of the agreement between buyer and seller being directly written into lines of code. In this case, the smart contract would require miners to submit coins in the form of an escrow and risk losing them if they abandon the pool before completing mining of the block. This would discourage miners from engaging in pool hopping.

The solution proposed in the paper "Pool-Hopping in Proof-of-Stake Cryptocurrencies" is called "randomized consensus." The authors argue that the current solutions used to prevent pool hopping in proof-of-stake (PoS) cryptocurrencies are inadequate. They propose a new scheme to solve this problem. Randomized consensus works by randomly selecting validators from the pool to validate each block [2]. This means that no single validator can predict when they will be chosen to validate a block, making it more difficult for validators to engage in pool hopping. Additionally, the authors propose a reward scheme that incentivizes validators to stay in the pool and behave honestly. Overall, the authors argue that randomized consensus is a more effective solution to the problem of pool hopping in PoS cryptocurrencies than current solutions [2][3]. The scheme has been shown to be effective in simulations and could be implemented in practice to improve the security and stability of PoS blockchains.

The solution proposed in the paper "Game-Theoretic Analysis of Pool Hopping in Proof-of-Stake Blockchain Systems" is a new game-theoretic model to incentivize miners to behave honestly and discourage pool hopping. The authors first analyse the problem of pool hopping using game theory, and argue that it is a rational strategy for miners to adopt [4]. They then propose a new model that includes two mechanisms: a penalty for pool hopping and a reward for long-term commitment to the pool. The penalty for pool hopping is designed to discourage miners from leaving the pool to join another one. The reward for long-term commitment is designed to incentivize miners to stay in the pool and behave honestly over a longer period of time [5]. Overall, the authors argue that their game-theoretic model can effectively prevent pool hopping in proof-of-stake blockchain systems. The model has been shown to be effective in simulations and could be implemented in practice to improve the security and stability of PoS blockchains.

The paper "Pool-Hopping Attacks in Proof-of-Work Mining Pools" examines the problem of pool hopping in proof-of-work (PoW) mining pools and proposes a new scheme to prevent this problem. The authors analyse the factors that contribute to pool hopping, such as the pool's payment scheme and the miner's hash rate. The authors propose a new scheme called "adaptive pool hopping detection" to prevent pool hopping in PoW mining pools. The scheme works by monitoring the hash rate of miners in the pool and detecting sudden changes that could indicate pool hopping [6]. If a sudden change is detected, the scheme imposes a penalty on the miner, such as lowering their payout or excluding them from the pool. This penalty is designed to discourage pool hopping and incentivize miners to remain in the pool. Overall, the authors argue that their adaptive pool hopping detection scheme can effectively prevent pool hopping in PoW mining pools [6]. The scheme has been shown to be effective in simulations and could be implemented in practice to improve the security and stability of PoW blockchains.

The paper "Mitigating Pool Hopping in Multi-Pool Mining" proposes a new mechanism called "stratum mining" to prevent pool hopping in the context of multi-pool mining. Multi-pool mining refers to the practice of miners switching between different pools that mine different cryptocurrencies to maximize their profits. The authors argue that the current solutions used to prevent pool hopping are not effective in this context because they do not consider the dynamics of multi-pool mining. The authors propose stratum mining as a solution to this problem. Stratum mining involves the pool setting the difficulty target for each miner based on their hash rate. This difficulty target is adjusted periodically to prevent pool hopping. If a miner's hash rate drops below a certain threshold, their difficulty target is lowered, making it easier for them to mine. This makes it less attractive for miners to hop between pools because their hash rate is tied to their difficulty target [6]. Overall, the authors argue that stratum mining is a more effective solution to the problem of pool hopping in multi-pool mining. The mechanism has been shown to be effective in simulations and could be implemented in practice to improve the security and stability of multi-pool mining.

The paper "Pool-Hopping and Block-Withholding in Bitcoin Mining" proposes a new mechanism called "non-outsourceable puzzles" to incentivize miners to contribute honestly to the blockchain and prevent pool hopping and block withholding. The authors analyse the problem of pool hopping and block withholding in Bitcoin mining, which refers to miners joining a mining pool and then not contributing their resources honestly, leading to lower rewards for the other miners in the pool. The authors propose non-outsourceable puzzles as a solution to this problem [7]. These are puzzles that cannot be outsourced to other parties, so a miner must solve them on their own. By requiring miners to solve these puzzles, the mechanism incentivizes them to contribute honestly to the blockchain. Non-outsourceable puzzles also prevent pool hopping because they make it more difficult for miners to quickly switch between different pools. The mechanism ensures that a miner must invest a certain amount of computational power to solve the puzzle, making it harder to switch between pools and receive rewards [8]. Overall, the authors argue that non-outsourceable puzzles are a more effective solution to the problem of pool hopping and block withholding in Bitcoin mining. The mechanism has been shown to be effective in simulations and could be implemented in practice to improve the security and stability of Bitcoin mining.

*Existing systems*

**Slush’s method** is designed to resist pool-hopping. Here instead of basing participants’ reward on a simple count of the number of shares they have submitted during the round, each share submitted credits the participant with a certain quantity of score, and the block reward at round end is distributed among participants in proportion to their score. The score given for each share depends on the amount of time that has elapsed since the round started. The more time has passed, the higher the score [9]. Because in the proportional method shares submitted early are inherently worth more than late shares, this scoring counters the effect. The scoring function used is exponential, s = exp(T/C) where s is the score given for a share submitted at time T, and C is some constant.

The parameter C controls how fast the score awarded for every share grows – the lower it is, the faster the growth. Equivalently, it controls how fast the score of a given share decays relatively to new shares. If C is low, the decay is fast – this means that every share has a high chance to decay completely before a block is found, thus receiving no payment, while if a block is quickly found, the payment given will be very high because it will not be split between many other shares [9]. The result is that a low value of C significantly increases the variance of the received payments, while reducing the vulnerability to pool-hopping.

The main drawback of this method is that a steady state is only reached sometime after a round starts. At the beginning of a round, there aren’t many previous shares with which a potential reward is to be shared, and thus it is more profitable to mine early in the round [10]. The magnitude of the effect is not as large as in proportional pools, but the method is still not completely hopping-proof.

**Geometric method** is a hopping-proof reward system inspired by slush’s method. It, too, involves exponentially decaying score, but addresses the weaknesses of slush’s method and puts the system on a solid mathematical framework. In this method, there are two kinds of fee, a fixed fee and a variable fee. The fixed fee is a constant amount taken from the reward of every block [11]. The variable fee takes the form of a score granted automatically to the pool operator at the start of every round, and decays just like the score of participants. Therefore, the shorter the round, the higher the variable fee. This is designed to create a steady-state where the score granted for every new share, relatively to already existing score and the score of future shares, is always the same, thus there is no advantage to mining early or late in the round.

In all the above discussed solutions for pool hopping, there are numerous drawbacks. Some of them are:

* Fraudulent miners are no penalised or fined.
* No incentives given for honest miners.
* Previous history of miner’s actions is not tracked.
* Solutions proposed are theoretical and cannot be implemented in real-world.
* There is a risk of centralization.
* Some are resource intensive and may increase transaction costs.

1. **Proposed solution**

The following four steps make up the proposed smart contract-based pool hopping attack prevention model:

1. The miner first submits a request to join a mining pool.
2. The pool manager asks the miner to give their address, which they have linked to the block address they have saved locally in a database.
3. The pool manager locates the miner certificate from the blockchain network using the block address.
4. Finally, the miner will be required to agree to the terms of a smart contract requiring them to submit coins as escrow in order to safeguard the mining pool if they have repeatedly switched mining pools and have breached a high number of prior smart contracts.

Finally, the pool manager will update the mining certificate and return or take hold of the submitted coins based on the miner’s behaviour. The proposed model preserves the symmetry of the mining pool by preventing miners from leaving and withdrawing their computational power. Existing methods present detection strategies; however, they do not present a robust solution that helps prevent any future pool hopping attacks.

The pool managers' ability to locate a miner's certificate using the miner's address in a ledger is the cornerstone for this model's operation. The miner's address is where they get block rewards, to use the popular Ethereum-based blockchain wallet Ethpool as an example. This is the address that they use to send bitcoin to other users or to earn rewards for successfully mining a block. Each miner's certificate has a different miner's address, and no two certificates have the same miner's address.

Based on the miner's conduct after leaving a mining pool, the pool manager will update the certificate. The revised certificate will be sent to the blockchain network, and the pool manager's local database will store the block address along with the updated certificate. The miner's certificate includes the following information: the address of the miner, the number of hops, any prior smart contracts upheld, and any prior smart contracts broken.

The pool manager requests the miner's miner address when the miner applies to join a new mining pool. The pool manager searches the local database using the address to find the hash value of the block on the blockchain network that contains the miner certificate. The pool manager will create a brand-new smart contract based on the information in the miner certificate.

The certificate will be updated by the pool manager, who will then submit it as a blockchain network transaction. The local database is associated with the miner's address and stores the block address holding the miner certificate. This makes sure that nobody else can change the information on the miner's certificate.

A pool administrator may permit a miner who frequently switches between numerous mining pools to join their network of pools. For this to work, the miner must consent to the parameters of a smart contract in which they will deposit coins as escrow. The miner must stick with the mining pool network according to the terms of the contract until the block is mined. The miner receives their submitted coins back after the block has been successfully mined, together with any benefits from mining the pool. The miner will forfeit all of their coins if they break the contract. The coins are subsequently dispersed to all of the network's active miners.

When a miner requests to join the mining pool, the pool manager is required to evaluate the risk factors of allowing the miner to join. The manager will determine the trustworthiness of the miner to remain as part of the mining process until block mining is complete. This module is based on a three-step process, which is as follows:

1. The miner will ask the pool management to connect to the network. If the miner has a history of mining, a pre-issued miner certificate that details the miner's past conduct will be available. The pool manager will nonetheless issue a fresh miner certificate if the miner is a novice miner. In order to locate the miner's certificate from the ledger, the pool manager will ask the miner to provide their miner address. Each miner's address is specific to their individual miner certificate. As this address is necessary for the miner to get the rewards for successfully mining a block, a request to it does not breach the miner's right to privacy.
2. Verify miner certification: Using the miner's address to find the block address, the pool manager consults the ledger kept in the local database. The pool manager finds the miner's certificate stored as data on the blockchain network using the block address. The pool manager will primarily look at the miner's Hop-Count, past SC-upheld counts, and SC-violated counts. The pool manager evaluates whether a miner is safe or poses a risk to the entire mining pool depending on the value of the Hop-Count and SC-violated.
3. Assess the risk to miners. The miner's potential risk to the mining pool will be assessed by the pool management. The manager will count two components, SC-violated (**μ**) and Hop-Count(**α**), respectively. The miner is deemed safe and joins the mining pool if both values are zero. The miner is rated at a risk to the mining pool if the miner certificate shows as one or above. The component SC-Upheld(**β**) is should be ideally greater, implying that miner is safe and reliable.

Miner risk is calculated as shown below:

Let **M** be the Risk factor of miner

Miner’s risk increases with increasing number of Hop-Count i.e,

M ∝ α

Miner’s risk increases with increasing number of SC-violated i.e,

M ∝ μ

* M ∝ (α + μ)

ln(M) = k (α + μ) (Here, k is proportionality constant)

* M = e k(α + μ)

Value of M should lie within the range (0,1)

Thus, k = - K

* Mrisk = 1 - e-K(α + μ) (Equation 1)

Miner risk is evaluated based on two conditions: (1) the miner is safe and (2) the miner is a risk to the mining pool. Two different case studies are presented to explain both conditions. To satisfy the first condition, we assume that the miner is joining the mining pool for the first time. The α and β values are assumed zero, and Mrisk is calculated as zero and is safe to join the mining pool.

To fulfil the second condition, we assume that the miner has previously abandoned the mining pool at least once. The α and β values are one; thus, Mrisk is non zero. The miner is evaluated as a high risk to the mining pool.

The terms of the smart contract are decided for the miner, who is evaluated as a risk to the mining pool because of a high Hop-Count value. The miner is required to submit coins as an escrow, which acts as penalty if the miner leaves the mining pool without completing the mining of the block. The miner risks losing the coins if they leave the mining pool early. This is based on a three-step process, which is as follows:

1. Miner risk: If the Hop-Count of the miner is determined to be at one or above, then they are a risk to the mining pool. The miner exhibits behaviour that could potentially hop the mining pool when it is not suitable for them and return when the rewards are high. The pool manager will issue a smart contract, based on which the miner must agree to a set of terms, to be allowed to join the mining pool.
2. Terms of the smart contract: The necessity for escrow is the main prerequisite for a smart contract between the mining pool and the new miner. A miner with a high Hop-Count value must enter a specific number of coins into the smart contract in order to effectively defend against pool hopping assaults. The possibility of losing the money serves as a deterrent for the miner to leave or quit the mining pool. The pool manager will keep track of how many prior smart contracts the miner has upheld and violated. Even though the miner has a history of having a high Hop Count, their behaviour from the past mining pools shows that they are a trustworthy miner, as seen by the number of prior honoured smart contracts.
3. Calculating the escrow value: The miner deposits coins as a form of security deposit, forfeiting the coins deposited in escrow if the miner leaves the mining pool. A miner must submit coins as an escrow if they have a Hop-Count and SC-violated value of one or higher. Our suggested solution for caution deposit isas following:

* D =Min. Deposit +Mrisk \* (Max. Deposit) (Equation 2)

Escrow value is based on three conditions:

(1) the miner has no risk

(2) the miner has considerable risk and

(3) the miner is highly risky

In first condition, the miner might have joined the mining pool for the first time or may be an honest miner with no fraudulent history and has a Hop-Count value (α) and SC-violated value (β) value of zero. Using the above Equation (2), we determine that the miner is only required to pay the minimum number of coins as escrow deposit.

In the second scenario, we assume that the miner has abandoned the mining pool once or more previously. The Hop-Count value (α) and SC-violated value (β) are both greater than or equal one. Using Equation (1), Mrisk is calculated to be somewhere in the range between 0 and 1, and the miner is evaluated as a risky miner. Using Equation (2), we determine the escrow amount the miner is required to pay, which is minimum deposit along with the calculated number of coins.

In third condition, we suppose the values of α and μ is maximum. Using Equation (1), Mrisk is calculated to be the highest and its value closer to 1. Mathematically, using Equation (2), the miner is required to pay the sum of both minimum and maximum number of coins as escrow deposit. This increased escrow deposit prevents the miner from leaving the pool, discourages any malpractices and safeguards the mining process.

1. **Evaluation methodology**

In this section, the proposed model using a practical case study is based on two different scenarios: when a miner finishes mining of the block and when the miner abandons the mining pool. The two different scenarios are evaluated according to the proposed methodology. Loss of computational power is simulated by comparing existing studies with our proposed model.

The miner often leaves the mining pool once the block has been fully mined. A miner may, however, leave early and launch a pool hopping assault. Based on the miner's actions, the pool manager will update the certificate. If the agreed-upon terms are met or not in this module, it will be determined by the smart contract that was issued. When the block is successfully mined, the miner receives the coins placed in escrow, and the Hop-Count value is decremented by one on the new miner certificate. The miner forfeits all funds placed in escrow if they leave before mining the block, and their Hop-Count value is raised by one. This module is based on a three-step process, which is as follows:

Mining of block: The miner in agreement to the terms of the smart contract joins and becomes a member of the mining pool. The miner will continue to mine the block and be part of the reward sharing system adopted by the pool. Upon successful completion of the mining, all miners receive the reward for solving the cryptographic puzzle of the block.

When mining is finished, the pool manager will decide whether or not to return the coins to the miner who placed them in escrow. At this point, it is determined whether the smart contract was fulfilled or violated. The miner will receive every coin put into escrow if the smart contract's conditions are met. The miner will forfeit all coins offered in escrow if the conditions are not met. To compensate for the loss of processing capacity caused by the pool hopping assault, these coins will be distributed equally to all other miners in the mining pool. The financial repercussions of breaching the contract are an essential precaution to deter the miner from leaving the mining pool. Miners leaving the pool for their financial gains and losing coins to the mining pool breaks that objective.

Update miner certificate: The pool manager will update the miner certificate based on the outcome of the smart contract. The smart contract is updated based on two conditions: (1) the miner committed a pool hopping attack (Matt)and abandoned the mining pool, and (2) the miner is safe (Msafe) and completed the mining process. (Matt) and (Msafe) are used to help explain the updating process of the miner certificate. The final values of α, β, and μ are important and are updated in the miner’s certificate to help evaluate the miner risk (Mrisk) using Equation (1) and escrow amount using Equation (2). If the miner abandoned the mining pool, the Hop-Count (α) and the SC-violated (β) value increases by one and the SC-upheld (μ) value decreases by one.

The usage of smart contracts and miner certificates is encouraged by the proposed smart contract-based pool hopping attack prevention paradigm in order to prevent pool hopping attacks, which regularly target mining pools. In order for various pool administrators to access them and take the required precautions to protect their mining pools from such assaults, miner certificate ledgers make sure that a record of each miner is preserved at all times. All honest miners will receive money in the case of an attack thanks to the implementation of smart contracts and an escrow of funds provided by pool offenders. There are numerous pool hopping detection techniques, however none of them are as effective against pool hopping as the above proposed model.

1. **Conclusions**

The study discusses the issue of efficiently guarding against pool hopping attacks and introduces a smart contract-based pool hopping attack defence for blockchain networks.

Our research is novel in three ways:

(1) Miners' risk is assessed before they join the mining pool.

(2) The terms for issuing a smart contract and determining the escrow amount are presented.

(3) The unique idea of a miner certificate to keep track of miners' records and its updating is presented with a mathematical equation.

By giving the submitted escrow coins evenly to the current pool members, the suggested model imposes a severe penalty on miners who quit the mining pool.

1. **Future work**

The suggested methodology has two drawbacks:

(1) Incorrect updating of the miner certificate as a result of a disagreement between the miner and the pool manager.

(2) Database security of the local database can be jeopardized as it is maintained only by the pool manager.

In subsequent research, we will make sure that miner certificates are updated with the approval of all active miners in the mining pool in order to overcome the limitations and safeguard the database.

Future work will address the limitations of our proposed method by mitigating possible conflicts between the miner and the pool manager. Verification of miner data submitted to the miner’s certificate in the blockchain network is possible by allowing all existing members of the mining pool to approve the data before the pool manager updates the miner’s certificate. The proposed research allows future research to focus on automating the proposed model, where the miner’s risk and escrow amount are evaluated automatically, and miner certificates are updated without pool manager intervention.

1. **References:**

* Kiayias, A., Panagiotakos, G., & Tsiopoulos, L. Pool-Hopping in Proof-of-Stake Cryptocurrencies. In International Conference on Financial Cryptography and Data Security (2018). Springer.
* Delgado-Segura, S., Pérez-Solà, C., & Herrera-Joancomartí, J.. Game-Theoretic Analysis of Pool Hopping in Proof-of-Stake Blockchain Systems. IEEE Transactions on Information Forensics and Security, 2018.
* Nayak, K., Kumar, A., & Miller, A. Pool-Hopping Attacks in Proof-of-Work Mining Pools. In Proceedings of ACM SIGSAC Conference on Computer and Communications Security (2016).
* Kim, D. H., & Lee, H. K. Mitigating Pool Hopping in Multi-Pool Mining. Journal of Information Processing Systems, 2016.
* Sapirshtein, A., Sompolinsky, Y., & Zohar, A. Pool-Hopping and Block-Withholding in Bitcoin Mining. In International Conference on Financial Cryptography and Data Security (2016).
* Singh, Sushil Kumar, Mikail Mohammed Salim, Minjeong Cho, Jeonghun Cha, Yi Pan, and Jong Hyuk Park. "Smart Contract-Based Pool Hopping Attack Prevention for Blockchain Networks", 2019
* Eugenio Cortesi , Francesco Bruschi , Stefano Secci , Sami Taktak  .“A new approach for Bitcoin pool-hopping detection” Received 29 May 2021, Revised 6 December 2021, Accepted 27 December 2021, Available online 14 January 2022, Version of Record 19 January 2022.
* Nitin K. [Tyagi, Mukta Goyal & Adarsh Kumar](https://link.springer.com/chapter/10.1007/978-981-16-8546-0_11), “Game Theory-Based Proof of Stake Mining in Blockchain for Sustainable Energy Efficiency”, 2017
* Hongwei Shi, Shengling Wang, Qin Hu, Xiuzhen Cheng, Junshan Zhang & Jiguo Yu. “Hopping-Proof and Fee-Free Pooled Mining in Blockchain”

# 