# Collusion attack from hubs in the blockchain offline channel network

Subhasis Thakur[1][0000−0001−6579−724X] and John G. Breslin[2][0000−0001−5790−050X]

[1] National University of Ireland, Galway, Ireland subhasis.thakur@nuigalway.ie
[2] National University of Ireland, Galway, Ireland john.breslin@nuigalway.ie

**Abstract.** Offline channels can improve the scalability of blockchains by reducing the number of transactions in the blockchain. Offline channels provide Path-Based fund Transfer (PBT) service which allows a pair of peers without a mutual channel to transfer fund between them using paths in the channel network. In PBTs, peers allow a 3rd party to use their channel for fund transfer in exchange for a transfer fee. There are channels in the Bitcoin Lightning network which are designed to collect such PBT transfer fees. An analysis of Bitcoin's Lightning network revealed the existence of hubs or nodes with very high degree in the channel network. There are only 10 nodes who own more than 50% funds in the Lightning network. These nodes are designed to facilitate PBTs among peers with a low degree (number of channels) in exchange for transfer fees. The emergence of hubs in channel network created the possibility of collusion attack on the channel network where a group of hubs deliberately make few channels non-operational to prevent PBTs involving a selected set of hubs (victims of the collusion attack). In this paper, we model such collusion attack using cooperative game theory and using Banzhaf index we classify the vulnerability of the hubs from the collusion attacks. We propose a design principle of the channel network that can decrease the possibility of collusion attacks.

**Keywords:** Offline channels · Blockchain · Collusion · Banzhaf Index

## 1 Introduction

Scalability is a prominent issue in the blockchain. While Mastercard processes 50000 transactions per second, Bitcoin processes 7 and Ethereum processes 15 transactions per second. Offline channel [15] is a useful tool to improve the scalability of blockchains. A pair of peers only need to broadcast two transactions to open and close a channel between them. A channel (theoretically) supports an infinite number of transactions between them. Channels offer offline Path-Based fund Transfer (PBT) service [12]. A PBT uses a path in the offline channel network for fund transfer between two parties who do not have a channel. Examples of offline channel networks are Lightning Network for Bitcoin, the Raiden Network[2] for Ethereum and SilentWhispers [12] for credit networks.

Peers allow PBT execution through their channels in exchange for small transfer fee. Hence PBT can be a source of revenue. While ordinary peers with limited funds cannot establish a great number of channels for the purpose of generating revenue, there are financial entities (with access to significant funding) who can establish offline channels for the sole purpose of collecting the PBT transfer fees. In Bitcoin Lightning network we witnessed this phenomenon. There are only 10 nodes with control of more than 50% funds available in the Lightning network. These nodes have a very high degree. We refer to these high degree nodes as the hubs.

Hubs can improve the performance of the offline channels by reducing the PBT completion time and improving the success rate of PBT execution. But it brings a new form of collusion attack on the channel network. In a collusion attack, a collusion (a group of hubs) can make few channels non-operational to prevent PBTs among a set of targeted hubs. The targeted hubs are the victim of the collusion attack. The objective of this paper is to investigate how such a collusion attack can be executed in the channel network and develop a mechanism that can lower the possibility of such anattack. We have the following results in this paper:

1. We present a mathematical model of collusion attack on the channel network. We use cooperative game theory and coalitional power index (Banzhaf index) [4, 3] to model collusion attacks. We model a collusion as a coalition and Banzhaf index gives the estimation on the importance of a hub's participation in a collusion attack.
2. We present a model of the likelihood of collusion attack among the hubs in a channel network using Banzhaf indices.
3. We analyze the possibility collusion attack in the Bitcoin's Lightning network. We found that there are 62 nodes who can execute collusion attacks against 90% of their neighbors in the Lightning network.

The paper is organized as follows: In Section 2 we discuss related literature, in Section 3 we present the collusion attack problem, in Section 4 we present a method to evaluate the possibility of collusion attack in a channel network, in Section 5 we present a method to lower the possibility of collusion attack, in Section 6 we evaluate Bitcoin's Lightning network to evaluate possibility of collusion attacks and we conclude the paper in Section 7.

## 2 Related literature

In this paper, we study an attack model in the offline channel network. Offline channels are designed to improve the scalability of blockchains. Examples of such developments are as follows: Bitcoin Lightning network was proposed in [15] which allows peers to create and transfer funds among them without frequently updating the blockchain. Similar networks are proposed for Ethereum [2] and credit networks [12]. A privacy-preserving payment method in the credit network was proposed in [13]. Recent advances on the offline channel network

are focused on the development of routing protocols for offline channels. Examples of such routing protocols are as follows: A method for anonymous payment to improve privacy in PBT was developed in [9]. [10] proposed a decentralised routing algorithm for the channel network.

Current research in the offline channel network for blockchains is focused on developing better routing protocols for balancing the channels, privacy preserving routing and fast routing protocols. But there is a lack of analysis on the collusion attack that hubs in a channel network can orchestrate. In this paper we analyze collusion attacks among the hubs in a blockchain peer to peer network. The collusion attack is similar to eclipse attack. [11] analysed eclipse attack [5, 16] on Bitcoin network and it proposed appropriate countermeasures. [14] analyzed a combination of selfish mining and eclipse attack on blockchain peer to peer network. In this paper, we analyze collusion attack among hubs in the blockchain network instead of analyzing eclipse attack on the entire peer to peer network. We perform such analysis as we observed centralization of the channel network. Our results can characterize the effect of centralisation in the channel network. We will use the Banzhaf index to characterize collusion attacks. [4] analyzed Banzhaf power indices for network flow games and [3] analyzed Banzhaf power indices for network connectivity games. These research have proved that it is NP-hard to compute the Banzhaf index.

## 3 Collusion attacks

First, we present an analysis of the Bitcoin's Lightning network to illustrate the existence of hubs in the channel network. Next, we present the model of collusion attack on the channel network.

### 3.1 Hubs in Bitcoin Lightning network

We use the Bitcoin Lightning network data [1] to explain the existence of hubs. The dataset has 2810 nodes and 22596 edges. The average degree of nodes is 16. If we consider nodes with a degree more than 50 as hubs then, there are 168 hubs. Collusion is a coalition among the hubs which can prevent PBTs for the remaining hubs. A collusion attack can be executed by creating a cut the channel network.

Collusion is a group of hubs in the channel network who aim to prevent PBTs between a pair of targeted hubs or victims of the collusion attack. We will describe the model of collusion using a neighbourhood of a chosen hub. The neighbourhood will be restricted by the maximum distance from the hub. This will allow us to evaluate the potential of a hub to orchestrate a collusion attack in its neighbourhood. In the next Section we will define such collusion and we will define the potential of a peer to organise collusion as it Banzhaf index. Banzhaf index measures the value of a hub in a coalition (collusion) as it evaluates if the coalition will remain successful (to execute a collusion attack) if this hub leaves the coalition.

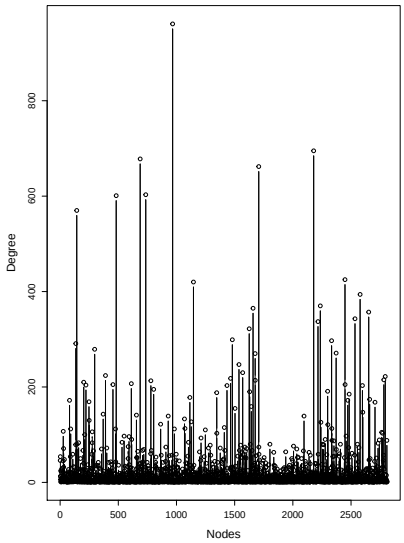


**Fig. 1.** Degree distribution of Bitcoin Lightning network data. It shows the existence of very high degree nodes.

### 3.2 Models of collusion attack

Let $G = (V, E)$ be a directed graph with $n$ nodes $V$ representing the hubs of the channel network and $m$ edges $E$ representing the channels among the hubs. Let $G^i$ be the subgraph induced by the vertices who are at most $k$ edges apart from $V_i \in V$ (including $V_i$) on the graph $G$ where $k$ is a positive integer less than diameter of $G$. $V^i$ will denote the set of nodes at most $k$ edges apart from $V_i$ or the set of vertices of the subgraph $G^i$.

We will define collusions w.r.t any specific node $V_i$ to use the subgraph $G^i$. A collusion is a subset of nodes $V^i$ such that:

1. It can produce a cut between a pair of hubs (or more pairs of hubs) in $G^i$. This pair of hubs is the victim of the collusion attack as PBTs between them will not be executed in $G^i$.
2. The hubs in the collusion have additional channels to allow the flow of tokens through them.

We formally define collusion as follows:

**Definition 1.** *In a hub network $G = (V, E)$, a collusion $C$ centred at $V_i$ is a subset of $V^i$ such that the following holds:*

1. *$V_i \in C$.*
2. *$|C| \leq \delta$ where $\delta$ is a positive integer.*
3. *Let $F \subset E$ be the set of edges originating from any $V_x \in C$.*

4. *There exists a pair of nodes* $(V_a, V_b) \in V^i - C$ *such that there cut* $F' \subset F$ *where the source is* $V_a$ *and sink is* $V_b$.
5. *There is a path in* $F - F'$ *that connects every node in* $C$ *to any node* $V_x \in V - V^i$.

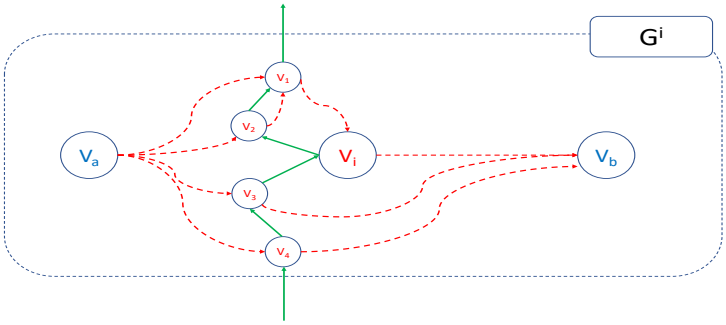


**Fig. 2.** Example of a collusion centred at $V_i$ that includes $V_i, V_1, V_2, V_3$ and $V_4$. The collusion produces a cut between $V_a$ and $V_b$ in $G^i$.

The explanation of the above notion of collusion attack is as follows:

1. We define a collusion w.r.t a node $V_i$. It helps us to define the set of collusions where $V_i$ can have significant contributions.
2. We restrict the size of collusions using the parameter $\delta$.
3. We restrict the size of a subgraph that collusion can control by the parameter $k$. If collusion can produce a cut between $V_a$ and $V_b$ in the subgraph $G^i$ then it means there is no path in $G$ with distance less than the distance between $V_a$ and $V_b$ in $G^i$. It means if the collusion blocks the paths between $V_a$ and $V_b$ in $G^i$ then cost of PBT transfer between $V_a$ and $V_b$ is increased by the PBT transfer fee of at least one more channel. Hence a collusion attack can at least increase the cost of PBTs between the victims even if the collusion could not completely prevent any PBTs among its victims.
4. Finally, collusion must have a path to the hubs outside the subgraph $G^i$ despite closing certain channels to execute the collusion attack. It is needed for executing the set of PBTs that the collusion allows.

**Definition 2.** *Weight of a collusion* $C \subset V^i$ *is the number of pairs of nodes for which the collusion can produce cuts.*

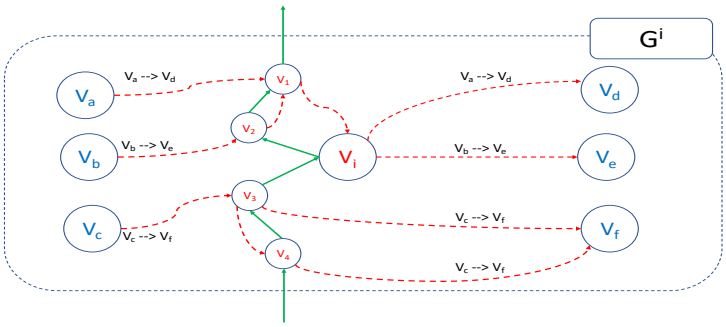


**Fig. 3.** The collusion is the set of hubs $V_1, V_2, V_3, V_4, V_i$ and the victim of the collusion are the hubs $V_a, V_b, V_c, V_d, V_e, V_f$. Weight of the collusion is 3 as it disconnects 3 pairs of hubs.

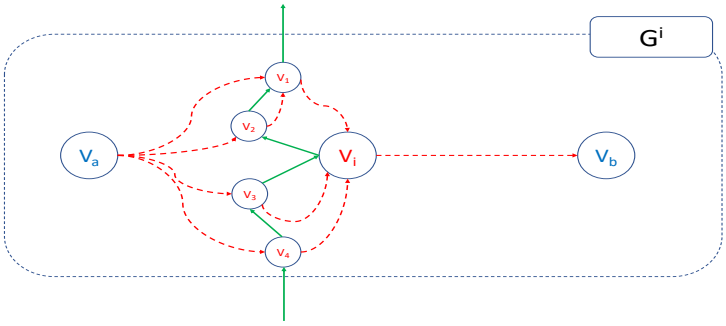


**Fig. 4.** The collusion is the set of hubs $V_1, V_2, V_3, V_4, V_i$ and the victim of the collusion are the hubs $V_a, V_b$. $V_i$ is a critical player as the collusion will fail if $V_i$ leaves.

Now we define collusion formation game as a cooperative game.

**Definition 3.** *A collusion formation game produces a set of collusions denoted as the set* $\{(C, V_i)\}$ *where* $C \subset V^i$ *is the collusion centred at* $V_i$ *as the result of cooperation among the members of each collusion. The value of a collusion is defined by the function* $\theta$ *as follows:*

$$\theta(C) = \begin{cases} 1 & \text{if } C \text{ can produce a cut for a pair of hubs } (V_a, V_b) \in V^i - C \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

Now we define the importance of a hub in a collusion.

**Definition 4.** *In a collusion* $(C, V^i)$*, a hub* $V_x \in C$ *is a critical player if* $\theta(C) = 1$ *and* $\theta(C - V_x) = 0$ *indicating that the collusion becomes unsuccessful if* $V_x$ *leaves the collusion. The number of collusions centred at* $V_i$ *where* $V_i$ *is a critical player is denoted by* $\nabla_i$*.*

Now we define the Banzhaf index of a hub for a collusion formation game as follows:

**Definition 5.** *Banzhaf index of the player* $V_i$ *in the collusion formation game is*

$$\beta_i = \frac{\nabla_i}{\sum_{V_x \in V^i} \nabla_x} \tag{2}$$

Note that we restrict the definition of the power of a hub within the subgraph in which it forms collusion. This is because the same subgraph is valid where the hub will be a victim of another collusion attack. Next, we will discuss the algorithm to compute the Banzhaf index.

## 4 Potential of collusion attacks

In this Section, we discuss a method to evaluate the possibility of executing collusion attack in a channel network. First, we will discuss the algorithm to compute Banzhaf index for collusion attack as defined in the previous Section. It should be noted that the computation complexity of computing Banzhaf index is NP-hard [17]. In this paper, we will use Algorithm 1 to estimate Banzhaf indices. The explanation of Algorithm1 is as follows:

1. In a subgraph $G^i$ centerd at $V_i$, we compute the number of collusions (subsets of nodes in $V_i$ with maximum cardinality $k$) where $V_i$ is a critical player.
2. It should be noted that if the number of nodes in $G^i$ is $x$ then the number of collusions where $V_i$ is a member is

$$\frac{x!}{(x-k)!k!} - \frac{(x-1)!}{(x-1-k)!k!} \tag{3}$$

$$\frac{(x-1)!}{(x-1-k)!k!}[\frac{x}{x-k} - 1] \tag{4}$$

**Algorithm 1:** Computation of Banzhaf index

**Data:** Hub network as $G = (V, E)$
**Result:** Banzhaf indices of $V$ as $\{\beta_i\}$
**begin**
  $Score \leftarrow$ a vector of length $n$
  **for** *Each* $V_i \in V$ **do**
    $G^i \leftarrow$ induced subgraph on $G$ by nodes within distance $k$ from $V_i$
    $d_1 \leftarrow$ degree of $V_i$ in $G^i$
    $d_2 \leftarrow$ is the diameter of $G^i$
    $Groups \leftarrow$ a $d_1 \times d_2$
    $N \leftarrow$ neighbours of $V_i$ in $G^i$
    $j \leftarrow 1$
    **for** *Each* $V_x \in N$ **do**
      $Groups[j,] \leftarrow$ outcome of a random walk of length $k_1$
      Remove edges in the path $Groups[j,]$ from $G^i$
      $j++$
    **for** *j in [1: size of set Groups]* **do**
      $C \leftarrow j$'th row of the matrix $Groups$
      $H \leftarrow$ created by deleting edges from these vertices with vertices outside $C$
      $H' \leftarrow$ created by deleting edges from the vertices $C - V_i$ with vertices outside $C$
      **if** *Is.connected(H) == FALSE & Is.connected(H') == TRUE* **then**
        $Score[i] \leftarrow Score[i] + 1$
  **for** *Each* $V_i$ **do**
    $\beta_i = \frac{Score[i]}{\sum_{V_x \in V^i} Score[x]}$

The number of possible collusions which includes $V_i$ is very large. It is computationally difficult to test all such collusions to check if $V_i$ is a critical player. Hence instead of checking all collusions we only check the collusions created by a set of random walks from $V_i$.

3. For each node $V_i$ we create $x$ random walks where $x$ is the degree of $V_i$ in the graph $V^i$.
4. For the set of vertices in each such random walk,
   **Case 1** We compute the if the deletion of the edges from the set of vertices in each random walk (treated as collusion) to the remaining vertices of $V^i$ (victims of the collusion attack) disconnects the graph $G^i$.
   **Case 2** Next, we compute if such disconnection of the graph is possible without $V_i$.
5. $V_i$ is a critical player if Case 1 is true and Case 2 is false. Using such information we compute the Banzhaf indices for all hubs.

Now we define the potential of collusion attack in the channel network as follows:

**Definition 6.** *The potential of collusion attack in a channel network can be estimated by the standard deviation of Banzhaf indices of hubs. High standard deviation indicates that there are few hubs who can easily execute collusion attack while the remaining hubs are unlikely to execute collusion attack.*

## 5 Method to reduce possibility of collusion attacks

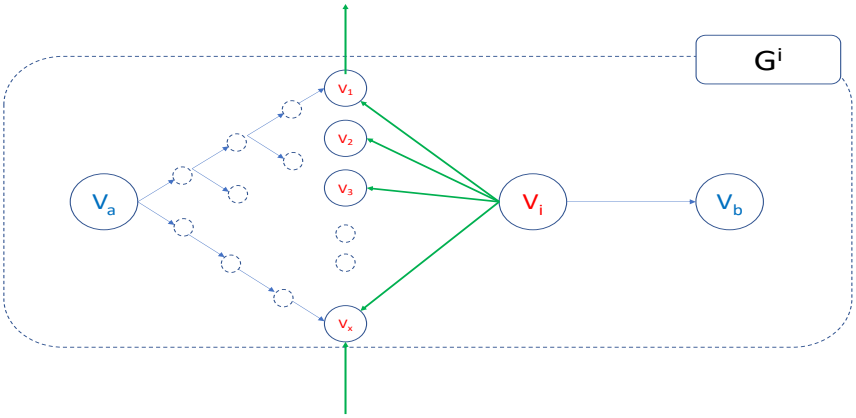


**Fig. 5.** Relation between degree of an attacker and probability of successful attack: it shows the worst case scenario for $V_i$ as a critical player to execute a collusion attack against $V_a$ and $V_b$.

**Theorem 1.** *The probability that a hub can successfully execute a collusion attack increases as its degree increases.*

*Proof.* Let the collusion $C$ is the set $V_i \cup (V_1, V_2, \ldots, V_x)$ and it wants to execute a collusion attack against the pair of hubs $(V_a, V_b)$ in the subgraph $G^i$. The attack scenario is illustrated in Figure 5. Let $V_a$ is at a distance $k-1$ from $V_i$ and $V_b$ is adjacent to $V_i$. In this scenario, we will estimate the size of collusion needed to execute an attack against the pair of hubs $(V_a, V_b)$ by $V_i$. As shown in Figure 5 the set of collusion is $V_1, \ldots, V_x$. In the worst case, the number of such collusion is the number of leaf nodes of a tree from $V_a$ with depth $k-2$. Hence the number of nodes is

$$X = 1 + d + d^2 + \cdots + d^{k-2} = \frac{d^{k-2} - 1}{d - 1} \tag{5}$$

where $d$ is the average degree of the hub network. Hence in the worst case $V_i$ needs cooperation from $\frac{d^{k-2}-1}{d-1}$ hubs to attack the pair $(V_a, V_b)$. Note that, $V_i$ can have $d_i$ neighbours (degree of $V_i$ in $G^i$). The relation between degree of $V_i$ and the probability that $V_i$ can attack on the pair $(V_a, V_b)$ is as follows:

1. $V_i$ may execute the attack if $d_i < \frac{d^{k-2}-1}{d-1}$. It means if $V_i$ has sufficient number of neighbours to form collusion then it can orchestrate such an attack.
2. The probability that $V_i$ has can execute the attack depends on the probability that each node $V_1$ to $V_x$ has $V_i$ as its neighbour. The probability that the node $V_1$ is a neighbour of $V_i$ is $d_i \frac{d-1}{d^k-1}$ where $\frac{d^k-1}{d-1}$ is the estimated number of edges in $G^i$.
3. Hence the probability that $V_i$ can execute the attack is $[d_i \frac{d-1}{d^k-1}]^X$
4. Thus the probability that $V_i$ can successfully attack the pair of hubs $(V_a, V_b)$ increases with the degree of $V_i$.

**Theorem 2.** *If hubs of the hub network have a uniform degree then, Banzhaf indices are approximately equal.*

*Proof.* Note that Banzhaf index of a hub depends on the number of collusions where it is a critical player. As proved in the previous theorem, higher the degree higher the probability that a hub can successfully execute a collusion attack. It proves that if the degree of nodes is equal then they will be equally likely to execute successful collusion attacks. Hence their Banzhaf indices will be approximately equal.

We propose that uniform Banzhaf indices may prevent collusion attacks in the hub network. This claim is based on the following observations:

1. In order to detect collusion attack, a hub must observe where its PBT requests are denied in the network. If the network is synchronous then it is a trivial problem. But in an asynchronous network, such detection problem is non-trivial. The collusion detection problem can be formulated as the problem of finding black holes in the network. Block holes are the nodes in a

network who destroy mobile agent visiting the node. The collusion detection problem can be formulated as a block hole search problem where mobile agents are network probes. The complexity of this search problem is NP-hard [7, 6]. But several approximation algorithms exist for both synchronous and asynchronous networks[8].
2. If the Banzhaf indices are approximately equal then it means if hub $V_a$ can attack hub $V_b$ then $V_b$ can also execute a collusion attack against $V_a$.
3. Hence equal Banzhaf indices will bring an 'equilibrium' in the sense that if $V_a$ attacks $V_b$ then $V_a$ can reciprocate such action. Hence it will prevent the hubs from orchestrating collusion attacks.

## 6 Evaluation with Bitcoin Lightning network

In this paper, we discussed a model of collusion attack in the offline channels for blockchains. We proved that (a) hubs will have approximately equal Banzhaf indices if their degrees are the same and (b) if hubs have equal Banzhaf indices then they are less likely to initiate a collusion attack. We measure the uniformity of Banzhaf indices as its standard deviation. In this Section, we perform an experimental evaluation of Algorithm 1 and we measure the Banzhaf index of nodes in the Bitcoin Lighting network. We have the following objectives in this experimental evaluation:

1. Prove the correctness of Algorithm 1 which measures the Banzhaf index.
2. Measure the Banzhaf indices of hubs in the Lightning network.
3. Explore the correlation between the uniformity of Banzhaf indices and diameter of a channel network.

We use Bitcoin Lightning network data [1] to analyze collusion attacks. [1] provided an API to access the Lightning network data. The downloaded data is in JSON format and RJSONIO package was used to process the data. The data contains (a) information about each node, i.e., public key and (b) network structure as the edge list. The data was accessed on 1st March 2019. It should be noted that the current size of Lightning network is slightly larger. The data contains the network structure of the Lightning network and it has the following properties:

| # Nodes | # Edges | Avg. Degree | Min Degree | Max Degree |
|---|---|---|---|---|
| 2810 | 22596 | 16 | 1 | 961 |

In the experimental evaluation, we execute Algorithm 1 using the above data as the input. First, we will evaluate the accuracy of Algorithm 1. We have proved that a peer's Banzhaf index depends on its degree. Greater the degree higher the Banzhaf index. We create a hub network by selecting nodes with a degree

in the ranges $(30, 100)$ from the Lightning network data. In this network, we execute Algorithm 1 to estimate the Banzhaf indices of the nodes. The result of such estimation is shown in Figure 6. It shows that as degree of hubs decreases the Banzhaf index also decreases. Figure 6 provides empirical evidence for the accuracy of Algorithm 1.

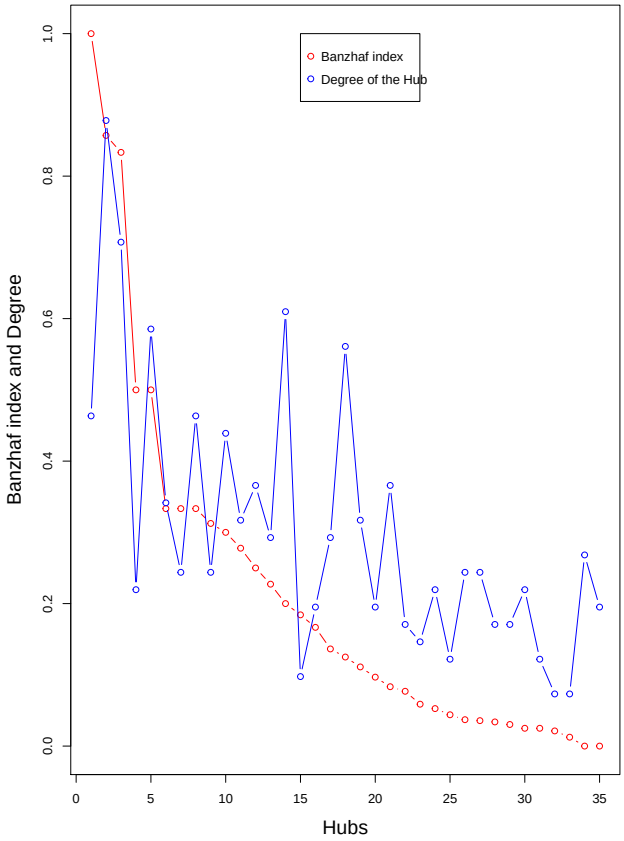


**Fig. 6.** The Figure shows the relationship between the Banzhaf index and degree of the nodes(normalized to the range [0,1]).

Next, we explore the relation between the diameter of a channel network and the uniformity of Banzhaf indices. We generate 17 hub networks from the Lightning network by selecting nodes with minimum degree 30 and maximum

degree $50, 55 \ldots 135$ respectively. We increase the maximum degree of hub network to increase the diameter of the network (a subgraph of the hub network) as we want to explore the relationship between the Banzhaf index and the network diameter. We use Algorithm 1 to compute Banzhaf indices of all nodes in each such hub network. We observe (shown in Figure 7) that as we increase the maximum degree of the subgraph generated from the hubs, the diameter of the graph becomes low. As the diameter becomes low it indicates graph converges towards a complete graph. Hence it becomes difficult to generate cuts in such a graph. We keep $k$ (diameter of the collusion graph) as 2 for all datasets. We want to analyze the possibility of collusion within a hub's immediate neighborhood, hence we use diameter 2 because the average diameter of these graphs is 4.5. The computed Banzhaf indices show that power indices become more uniform as the graph evolves towards a complete graph. It means it difficult to execute collusion attacks in channel network if the diameter of the graph becomes small.

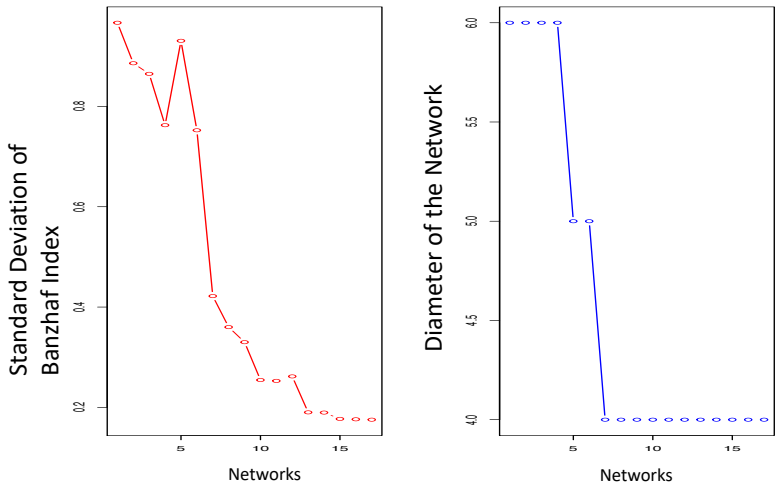


**Fig. 7.** Relation between the Banzhaf index and diameter of the network. The uniformity of Banzhaf increases as the diameter of the network decreases.

Next, we analyze the vulnerability of Bitcoin Lightning network against collusion attack. We use the following metric to measure such vulnerability per node as :

$$\text{vulnerability w.r.t node } v_i = \frac{\#\text{ of neighbours with Banzhaf Index less than } v_i}{\text{Size of neighbourhood of } v_i}. \tag{6}$$

We found that there are 62 nodes (shown in Figure 8) with vulnerability metric at least .9. This means there are 62 nodes who can execute collusion attacks against 90% of their neighbors in the Lightning network.

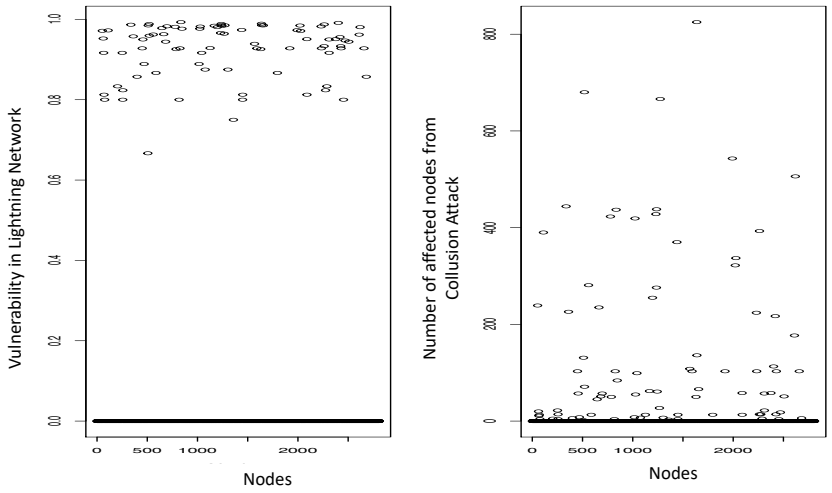


**Fig. 8.** Vulnerability metric for Lightning network. The left hand figure shows the vulnerability metric for each node in Lightning network and the right hand figure shows the number of neighbours with less Banzhaf index for each node in the Lightning network.

## 7 Conclusion

In this paper, we analyzed collision attacks among the hubs of offline channel networks. We have defined the potential for collusion attacks using Banzhaf indices. We have shown the correlation between uniformity of degree of the hub network and Banzhaf indices. Using experiments on Bitcoin's Lightning network we have shown that as the hub network evolves towards a complete graph it becomes more difficult to create cuts in the graph with a fixed number of edges and hence it increases the uniformity of power indices.

## 8 Acknowledgement

This publication has emanated from research supported in part by a research grant from Science Foundation Ireland (SFI) and the Department of Agriculture, Food and the Marine on behalf of the Government of Ireland under Grant

Number SFI 16/RC/3835 (VistaMilk), co-funded by the European Regional Development Fund.

## References


1. Bitcoin lightning network graph. https://graph.lndexplorer.com/api/graph (2019 (accessed March 1, 2019))
2. Raiden network. http://raiden.network/ (2019 (accessed March 1, 2019))
3. Aziz, H., Lachish, O., Paterson, M., Savani, R.: Power indices in spanning connectivity games. In: Goldberg, A.V., Zhou, Y. (eds.) Algorithmic Aspects in Information and Management. pp. 55–67. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
4. Bachrach, Y., Rosenschein, J.S.: Computing the banzhaf power index in network flow games. In: AAMAS (2007)
5. Castro, M., Druschel, P., Ganesh, A., Rowstron, A., Wallach, D.S.: Secure routing for structured peer-to-peer overlay networks. SIGOPS Oper. Syst. Rev. **36**(SI), 299–314 (Dec 2002). https://doi.org/10.1145/844128.844156, http://doi.acm.org/10.1145/844128.844156
6. Czyzowicz, J., Kowalski, D., Markou, E., Pelc, A.: Searching for a black hole in tree networks. In: Proceedings of the 8th International Conference on Principles of Distributed Systems. pp. 67–80. OPODIS'04, Springer-Verlag, Berlin, Heidelberg (2005)
7. Czyzowicz, J., Kowalski, D.R., Markou, E., Pelc, A.: Complexity of searching for a black hole. Fundam. Inform. **71**(2-3), 229–242 (2006), http://content.iospress.com/articles/fundamenta-informaticae/fi71-2-3-05
8. Dobrev, S., Flocchini, P., Prencipe, G., Santoro, N.: Mobile search for a black hole in an anonymous ring. Algorithmica **48**(1), 67–90 (Mar 2007)
9. Green, M., Miers, I.: Bolt: Anonymous payment channels for decentralized currencies. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 473–489. CCS '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3133956.3134093
10. Grunspan, C., Pérez-Marco, R.: Ant routing algorithm for the lightning network. CoRR **abs/1807.00151** (2018), http://arxiv.org/abs/1807.00151
11. Heilman, E., Kendler, A., Zohar, A., Goldberg, S.: Eclipse attacks on bitcoin's peer-to-peer network. In: Proceedings of the 24th USENIX Conference on Security Symposium. pp. 129–144. SEC'15, USENIX Association, Berkeley, CA, USA (2015), http://dl.acm.org/citation.cfm?id=2831143.2831152
12. Malavolta, G., Moreno-Sanchez, P., Kate, A., Maffei, M.: Silentwhispers: Enforcing security and privacy in decentralized credit networks. IACR Cryptology ePrint Archive **2016**, 1054 (2016)
13. Moreno-Sanchez, P., Kate, A., Maffei, M., Pecina, K.: Privacy preserving payments in credit networks: Enabling trust with privacy in online marketplaces. In: NDSS (2015)
14. Nayak, K., Kumar, S., Miller, A., Shi, E.: Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In: 2016 IEEE European Symposium on Security and Privacy (EuroS P). pp. 305–320 (March 2016). https://doi.org/10.1109/EuroSP.2016.32
15. Poon, J., Dryja, T.: The Bitcoin Lightning Network:Scalable Off-Chain Instant Payments https://lightning.network/lightning-network-paper.pdf

16. Singh, A., Ngan, T.W., Druschel, P., Wallach, D.S.: Eclipse attacks on overlay networks: Threats and defenses. Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications pp. 1–12 (2006)
17. Werman, S., Zohar, A.: Avoiding deadlocks in payment channel networks. In: Garcia-Alfaro, J., Herrera-Joancomartí, J., Livraga, G., Rios, R. (eds.) Data Privacy Management, Cryptocurrencies and Blockchain Technology. pp. 175–187. Springer International Publishing, Cham (2018)