Applied Network Science

RESEARCH

Open Access

Decentralizing the lightning network: a score-based recommendation strategy for the autopilot system

Mohammad Saleh Mahdizadeh[1*†], Behnam Bahrak[2†] and Mohammad Sayad Haghighi[1†]

†Mohammad Saleh Mahdizadeh, Behnam Bahrak, and Mohammad Sayad Haghighi contributed equally to this work.

*Correspondence: mahdizadeh.s@ut.ac.ir

[1] School of Electrical and Computer Engineering, University of Tehran, North Kargar St., Tehran, Iran
[2] Tehran Institute for Advanced Studies, East Daneshvar St., Tehran, Iran

Abstract

The fundamental objective of the Lightning Network is to establish a decentralized platform for scaling the Bitcoin network and facilitating high-throughput micropayments. However, this network has gradually deviated from its decentralized topology since its operational inception, and its resources have quickly shifted towards centralization. The evolution of the network and the changes in its topology have been critically reviewed and criticized due to its increasing centralization. This study delves into the network's topology and the reasons behind its centralized evolution. We explain the incentives of various participating nodes in the network and propose a score-based strategy for the Lightning Autopilot system, which is responsible for automatically establishing new payment channels for the nodes joining the network. Our study demonstrates that utilizing the proposed strategy could significantly aid in reducing the network's centralization. This strategy is grounded in qualitative labeling of network nodes based on topological and protocol features, followed by the creation of a scoring and recommendation model. Results of the experiments indicate that in the evolved network using the proposed strategy, concentration indicators such as the Gini coefficient can decrease by up to 17%, and channels ownership of the top 1% of hubs decrease by 27% compared to other autopilot strategies. Moreover, through simulated targeted attacks on hubs and channels, it is shown that by adopting the proposed strategy, the network's resilience is increased compared to the existing autopilot strategies for evolved networks. The proposed method from this research can also be integrated into operational Lightning clients and potentially replace the current recommendation methods used in Lightning Autopilot.

Keywords: Lightning network, Blockchain, Bitcoin, Autopilot, Preferential attachment

Introduction

Lightning Network, as a layer two protocol built on top of the Bitcoin network, aims to facilitate payment processes through micropayments, addressing the inherent limitations of the Bitcoin payment network. This has been achieved by establishing a network of off-chain payment channels.

Springer Open

© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Payment channels in the Lightning network are peer-to-peer entities. The capability of multi-hop payments in this network has led payment channels to not only facilitate micropayments between the two participating nodes but also enable micropayments with parties that do not share a channel. This attribute contributes to the formation of a payment network comprising nodes and payment channels.

Over time and during its evolution, the Lightning Network has, due to various incentives for users and the cost-intensive nature of creating multiple channels, developed a topology similar to that of centralized networks. To such an extent that the current network structure can be considered a robust scale-free structure or even aligned with the non-distributed core-periphery model (Lin et al. 2022).

Numerous studies have criticized the current structure of the Lightning Network, highlighting its vulnerabilities to various attacks, particularly topological attacks (Rohrer et al. 2019). While researchers have explicitly acknowledged the possibility of attacks like route hijacking attacks (Tochner et al. 2019, 2020) due to the network's imbalanced resource distribution, other attacks also deserve attention. These attacks take advantage of the centralized topology of the Lightning Network, facilitating attackers in executing attacks or exacerbating the resulting damages. Among these, we can mention griefing attacks (Robinson 2019; Mizrahi and Zohar 2021; Pérez-Sola et al. 2020), time dilation attacks (Naumenko and Riard 2021), as well as some forms of privacy attacks (Malavolta et al. 2017; Erdin et al. 2021; Herrera-Joancomartí et al. 2019; Rohrer and Tschorsch 2020).

Controlling the topology of a complex network is an ambitious and challenging endeavor, often unattainable. This is primarily due to the inaccessibility, management complexity, and the lack of authority in dictating communication among network nodes in many intricate networks. In other words, the defining factor of a network's topology is its nodes, and managing their decisions introduces significant complexities. Fortunately, in the case of the Lightning Network, there is an influential mechanism called "Autopilot" (2.2) that provides the opportunity for intervention in the network's evolution, albeit implicitly, through a recommending authority.

The goal of this research is to propose a strategy for the Lightning autopilot tool in a way that aligns users' personal interests with guiding the network's topology towards greater distribution simultaneously. In other words, Lightning network users should be able to engage in multi-hop payments with a minimal number of steps across the entire network, choosing from various available paths with suitable capacity. Additionally, the network's topology should provide relatively distributed characteristics towards mitigating resource concentration in limited nodes. This research strives to employ the influential power of the Lightning client's autopilot tool in influencing node decisions on how to establish payment channels in the network. This would transform the network's topology from its current state, resembling a partially centralized network, to a distributed topology with properties akin to distributed networks during its evolutionary process (Fig. 1).

Continuing, in Sect. "Related work", we will delve into the existing research conducted in this domain. Moving on to Sect. "Preliminaries", under the preliminaries, we will elaborate on the available data and the topological characteristics of the network. Section "Preferential attachment criteria" will provide an in-depth discussion on the

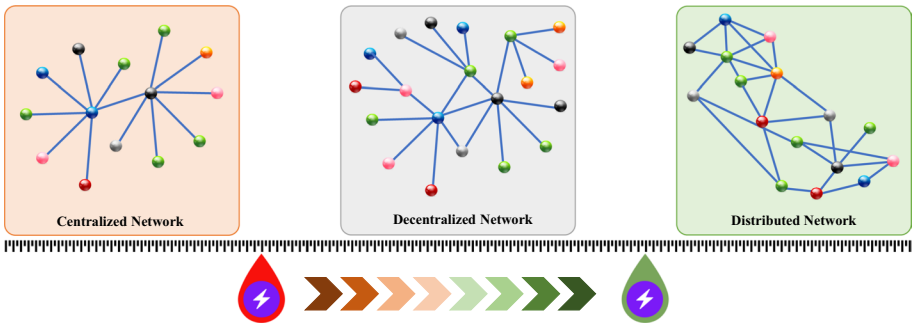

Fig. 1 The current research aims to facilitate the transition of the topology of the Lightning network from a decentralized network state with characteristics similar to centralized networks to a decentralized network state with characteristics akin to distributed networks

core research idea and the necessary preference attachment criteria within the Lightning Network. Section "Labeling" will address the process of labeling network nodes, while Sect. "The scoring model" will expound upon the formation of the scoring model. In Sect. "Empirical experiment", an empirical experiment will be conducted, followed by the presentation and analysis of the results in Sect. "Results and discussion". Lastly, Sect. "Conclusion and future works" will encompass the conclusion and offer insights for future research endeavors.

Related work

In this section, we will begin by reviewing previous research that has investigated the topological characteristics and the evolution of the Lightning Network. We will delve into articles that have examined potential topological vulnerabilities of the network. Furthermore, we will provide specific details about the Lightning Network's autopilot mechanism. We will also discuss studies and articles that have assessed the effectiveness of this module or critically analyzed its functioning.

Topological properties and evolution

The Lightning network is an evolving network, its growth being governed by the joining of new users to the network and the establishment of payment channels with existing nodes. Various studies have investigated this growth process and analyzed the topology emerging at each stage of this progression.

In Martinazzi (2019), Martinazzi delved into the evolution of the Lightning network, one year after its operational launch. Martinazzi's investigation was based on twelve snapshots of the Lightning network taken from February 2018 to January 2019. In this research, the Lightning network was modeled as a weighted, undirected network. Martinazzi demonstrated that over the course of the one-year study period, the network exhibited characteristics of scale-free networks, with an exponential parameter (γ) holding a value of approximately 2. A notable phenomenon highlighted by Martinazzi is the disassortativity present in the network. In other words, in the Lightning network, nodes with lower degrees tend to create more payment channels with nodes of higher degrees. From Martinazzi's perspective, the underlying cause for this behavior is the costliness of the channel creation process. This prompts network nodes to establish channels with

network hubs, enabling them to access other nodes at a lower cost through a fewer number of channels. Consequently, powerful hubs emerge within the network. Martinazzi notes that this phenomenon could be considered a threat to the Lightning network, as these hubs potentially possess the capability to control payment flows in the network and gather significant amounts of information. To support this intuition, Martinazzi assessed the network's robustness and demonstrated that a targeted attack based on the Betweenness centrality measure, by removing only 10 nodes from the network, can diminish up to 75% of the network's efficiency.

Guo et al. also conducted a systematic measurement on the topology of the Lightning network in a fifteen-month time span in Guo et al. (2019). After examining the degree distribution of Lightning network nodes in early April 2019 and providing evidence of the adherence of the degree distribution to a Powerlaw distribution (Newman 2005), they proceeded to evaluate the performance of the Lightning network. Their performance assessment comprised two categories: routing efficiency examination and network resilience against attacks. Their findings have demonstrated that 90% of the network nodes can access 90% of the other nodes solely through a maximum of 4 channels, highlighting the potential for optimal routing in the network. Furthermore, given that 90% of the network channels possess capacities less than 0.1 Bitcoin, payments exceeding 0.1 Bitcoin are unlikely to succeed with a high probability in the network. Concerning network resilience, they indicated that the removal of only 5% of the key network nodes would lead to the loss of 94% of the network's capacity. Over the course of the 15-month study period, this phenomenon has worsened, highlighting a more severe state of vulnerability through the network's evolution.

Rohrer and colleagues in Rohrer et al. (2019) have examined the topological properties of the Lightning network, analyzed its weaknesses, and introduced two topological attacks named channel exhaustion and node isolation. Their investigations have demonstrated that the topology of the Lightning network follows a scale-free and small-world structure, which renders the network susceptible to targeted attacks. Their findings have shown that an attacker can inflict severe damage on the network by targeting nodes based on centrality measures. Furthermore, an attacker can execute an attack by targeting nodes that host channels connecting participants in a min-cut with minimal resources. To prevent topological attacks, a systematic approach has not been suggested. They propose countermeasures to prevent node isolation, similar to what they refer to as the "flood-and-loot-attack" (Harris and Zohar 2020) mitigation method. This involves imposing limitations on the number and size of multi-hop micropayments on the client nodes' side. Additionally, they briefly mention that by implementing changes to the autopilot system of Lightning network clients, responsible for managing new channels, the topology of the Lightning network can be altered to exhibit increased resistance against such attacks.

In Seres et al. (2020), Seres and colleagues have also examined the topology and network resilience of the Lightning network against targeted attacks. After presenting the topological features of the network, they have shown that the degree distribution of the network follows a Powerlaw distribution with a parameter of $\gamma = 2.13$. This claim has been statistically proven by them using the Kolmogorov-Smirnov statistic (Massey Jr 1951). Furthermore, they have demonstrated that the Lightning network is resilient to

random attacks, but highly susceptible to targeted attacks. Specifically, by removing only the node with the highest degree, the network will be divided into 37 independent components without delay. They have stated that although there is a trade-off between a network's resilience to random and targeted attacks, strategies can be employed on the Lightning client side to control the joining of new nodes to the network.

Other studies have also highlighted the vulnerability of the Lightning network to targeted attacks. Examples include references to Lee and Kim (2020) and Martinazzi and Flori (2020). Lin et al. (2020), have pointed out the centralized structure of the Lightning network, in which high-degree hubs appear to play the role of channel switches. Their measurement method in the mentioned study involved the use of the Gini coefficient (Dorfman 1979). They attribute this phenomenon to the inclination for cost-efficient routing by the paying nodes and the preference for higher fees from the network hubs. They also thoroughly elaborate on the Lightning network's extreme vulnerability to targeted attacks.

Also, Camilo et al. (2022), by examining the Lightning network over an eight-month period, have indicated that this network is facing a strong trend towards the centralization of connections and resources. They found that more than half of the network's capacity is controlled by only 0.38% of nodes, which results in a significant vulnerability of the network against targeted attacks. According to their findings, these observations stand in contradiction to the primary goal of a decentralized payment network.

In Seres and Benczúr (2021), the topology of the Lightning network has been investigated from an economic perspective. They have facilitated the examination of the economic aspect of the Lightning network by designing a traffic simulator for it, without requiring precise information about balances, capacities, and micropayments. They demonstrated that nodes in the Lightning network, when functioning as routing nodes or intermediaries for multi-hop payments, do not have economic incentives. They attributed this phenomenon to the minimal fee amounts in comparison to the resources of the network nodes. In other words, if Lightning network nodes intend to benefit reasonably from network fees, the fee amount will significantly increase, violating the fundamental philosophy of the Lightning network, which involves micropayments with minimal fees. They also revealed that the current topology of the Lightning network leads to a considerable portion of network micropayments (possessing anonymity features) potentially becoming de-anonymizable. Although this can be improved by introducing nodes with low fees into payment paths.

Zabka and colleagues have indicated in a concise article (Zabka et al. 2022) that in order to maintain the liquidity of the Lightning network and to prevent payment path-based attacks, the network should move towards a more decentralized topology. Based on this premise, they conducted an investigation into centrality in the Lightning network to analyze its level of centrality. They utilized a tool called "TimeMachine" to gather data on the network's topology over time and analyze it. They demonstrated that while the Lightning network is decentralized, a small number of nodes in the network route the majority of payments, leading to a skewness in the network. They also showed that during their studied period, from 2020 to 2022, the Gini Index experienced a 10% jump, indicating a significant increase in the network's centrality.

In Lin et al. (2022), a study on the weighted Lightning network has also been conducted. By examining the topology of the Lightning network over two consecutive years, they have found that the Nakamoto Coefficient indicates a declining trend in the number of nodes owning 51% of the network's links, highlighting a highly uneven distribution of node centrality. Further investigation by them has revealed that the network's topology is moving towards a pattern that aligns with the core-periphery model (Borgatti and Everett 2000), with the dimensions of its core decreasing. Additionally, they have indicated that the removal of network hubs could rapidly divide the network into independent segments, potentially creating vulnerability to attacks like split attacks.

Lightning autopilot

The autopilot system within the Lightning Network can be seen as a suitable tool for the implicit management and control of the network's topology. The autopilot system is an automatic mechanism implemented on the Lightning client side to propose, create, and manage payment channels for the user node. While not being an integral part of the Lightning protocol, different Lightning clients employ various implementations of the autopilot system, often in the form of plugins or extensions. For instance, the c-lightning client (ElementsProject 2016) uses the lib_autopilot library (Pickhardt 2019), while the LND client (LightningNetwork 2017) has its own custom implementation of the autopilot system.

Autopilot tools in Lightning clients initially employed the Barabási Albert network generation model (Barabási and Albert 1999). This model relied on preferential attachment based on node degrees to suggest and establish payment channels. This autopilot policy has been extensively debated, questioned, and criticized by developers (Pickhardt 2018) and has even undergone academic scrutiny (Wang et al. 2022). One of the challenges associated with the preferential attachment strategy based on node degrees is the creation of powerful hubs in the network, which gradually erodes decentralization. Another strategy adopted by autopilot systems in Lightning clients involves proposing the creation of payment channels with nodes having the highest level of centrality, predominantly betweenness centrality. The main reason for using this strategy is to optimally utilize network paths in multi-hop payments. In other words, connecting new nodes to highly betweenness-central nodes ensures the shortest possible distance to end-to-end nodes, resulting in lower costs for multi-hop payments. However, it's worth mentioning that the preferential attachment strategy based on node centrality not only reduces resource distribution in the network but also significantly increases inequality in the distribution of payment channel ownership and the network's vulnerability to targeted attacks compared to preferential attachment based on node degrees (Tsiotas 2020; Topirceanu et al. 2018).

The subject of the impact of preferential attachment on network topology has attracted the attention of Lange et al. in Lange et al. (2021). They have investigated the influence of preferential attachment strategies on network topology. These strategies are based on one-dimensional metrics such as degree, betweenness centrality and K-Center. The investigated topological properties in their study include the Gini coefficient and the network diameter throughout their evolutionary process. Moreover, transaction success rates and network fees in the protocol have been analyzed. They concluded that a

balance should be struck between network efficiency and distribution, but achieving both is not feasible. They have also alluded to the possibility of employing composite metrics for preferential attachment, but further details of this idea are not provided.

Preliminaries

In this section, we will begin by presenting an overview of the available dataset, followed by a comprehensive analysis of the network's topological characteristics. We will provide the necessary evidence to elucidate the nature of the network's structure, and subsequently, statistical proofs will be presented.

Dataset description

In the Lightning Network, each node needs to maintain a version of the Lightning Network itself. This is essential for routing processes within the Lightning Network by the nodes, as the sending node must explicitly determine a route consisting of payment channels from itself to the destination. To achieve this, the Lightning Network has implemented a gossip protocol (Demers et al. 1987). Gossip protocols are used in peer-to-peer networks to share information among nodes. In the Lightning Network, network nodes establish encrypted peer-to-peer communications with each other and share received gossip protocol information among themselves. When two nodes in the network want to create a payment channel, they send the information about that payment channel to their neighboring nodes using the gossip protocol, and in turn, this information will be disseminated throughout the network.

Various entities engage in the routine collection, organization, and analysis of Lightning Network data. One of their services involves providing information about Lightning Network nodes and payment channels, or in other words, the topology of the Lightning Network. In addition to these information-providing services for the Lightning Network, each node individually within the network can autonomously conceive the Lightning Network graph on its side after an appropriate period of time. The process of connecting to the network, gathering gossip messages, and constructing the network graph within a Lightning node is facilitated by Lightning clients. Notable and popular Lightning Network clients include LND (LightningNetwork 2017), Eclair (ACINQ 2016), and C-lightning (ElementsProject 2016).

The network used in this study was a snapshot of the Lightning Network taken on November 27, 2021, collected by an LND client. The LND client provides a command-line interface called *lncli*, which connects to the gRPC service (Google 2015) offered by the respective client. The gRPC service of the LND client presents network graph information through a Unary RPC service named DescribeGraph. This information encompasses nodes and network channels along with their attributes. The dataset employed in this study was sourced from the repository by Rohrer and Elias (2021). Initially created for research conducted in Rohrer et al. (2019) by Rohrer, the repository has continued to gather network snapshots following the publication of the paper.

The information utilized in this study for each node of the network includes the following:

- **pub_key**: The public key of the network node.
- **alias**: The alias of the network node.

The information used in this study for each payment channel in the network comprises:

- **channel_id**: The identifier of the payment channel.
- **node1_pub / node2_pub**: The public keys of the nodes on both ends of the payment channel.
- **capacity**: The capacity of the payment channel.
- **fee_base_msat**: The base fee for each of the nodes at the ends of the payment channel.
- **fee_rate_milli_msat**: The fee rate for each of the nodes at the ends of the payment channel.

We define the Lightning Network as a directed network, denoted by $G = (V, E)$, where V represents the nodes of the Lightning Network (network addresses), and E stands for the payment channels between the network nodes. Furthermore, for each edge, such as e, specific attributes are assigned to that payment channel. In the following sections, some attributes for network nodes will also be derived using the attributes of edges (Sect. "Preferential attachment criteria").

Topological analysis

The network utilized in this study is a snapshot of the Lightning Network as of November 27, 2021. This network is connected and comprises a single component. The number of nodes in this network is 18133, and the number of payment channels is 76725.

The average degree of nodes in this network is 8.46. However, due to the extreme skewness in the distribution of node degrees, this value alone cannot serve as a reliable metric. The density of this network is 0.00046, indicating a sparse network.

The diameter of this network is 12, and the average shortest paths in the network is 3.65. Although this calculated value is independent of the channel capacities, it signifies the high accessibility of nodes and the absence of multi-hop payments with excessive hops in the network.

The number of bridges in this network amounts to 8560 edges. Given the existence of numerous nodes with a degree of 1, a significant portion of the calculated bridges are edges associated with single-degree nodes. The number of bridges in the network, excluding the bridges associated with single-degree nodes, is 174. This indicates the presence of a considerable number of vital payment channels for maintaining network connectivity.

The local clustering coefficient of a node indicates the level of connectivity among its neighboring nodes. This value signifies the similarity between the subgraph formed by a node's neighbors and a clique composed of those neighboring nodes. The distribution of local clustering coefficient values in the examined Lightning Network reveals the presence of a considerable number of nodes with high local clustering coefficient values (Fig. 2). Intuitively, nodes with a local clustering coefficient of 1 in the network are mostly nodes with low degrees, connected to hubs in the network that are interconnected themselves. This phenomenon is predominantly observed in networks with

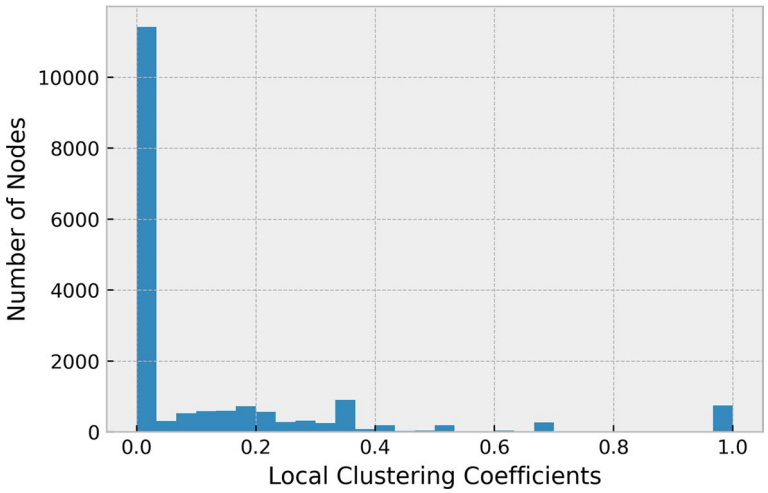

Fig. 2 Distribution of nodes local clustering coefficients

scale-free characteristics. Additionally, the clustering coefficient value, which is equal to the average value of local clustering coefficients of network nodes, is 0.1231. This value, excluding zero values, would be 0.3255.

The assortativity coefficient of degrees in a network indicates the tendency of nodes to connect to nodes with similar degrees. This characteristic ranges between 1 and -1, where a positive value signifies the propensity of high-degree nodes to connect to other high-degree nodes, and a negative value indicates the tendency of high-degree nodes to connect with low-degree nodes. In the Lightning Network context of this study, the assortativity coefficient of degrees is −0.195. This value reflects the linkage of low-degree nodes to high-degree nodes, a feature commonly found in scale-free networks.

The degree distribution of the investigated Lightning Network presents a distribution with high skewness. In other words, the network incorporates nodes that control a considerable number of network channels. It's noteworthy that one percent of nodes with the highest degree in the network are connected to 34% of the payment channels. This skewness is so pronounced that the top 10 hubs of the network own 10% of the network's payment channels.

To investigate the scale-free nature of the Lightning Network, we employ the methodology outlined in Clauset et al. (2009). Accordingly, using the Kolmogorov Smirnov test (Massey Jr 1951) and calculating the minimum distance for ascending sorted degree values, an initial degree value (x_{min}) is chosen. Subsequently, the Powerlaw distribution function is fitted onto the values beyond this point. The process of computing values and fitting the Powerlaw function in this study was facilitated by the powerlaw package (Alstott et al. 2014).

Figure 3 depicts the Powerlaw curve fitted onto the distribution of node degrees in the Lightning Network. In this fit, the minimum Kolmogorov Smirnov distance is calculated to be $D = 0.021$ for $x_{min} = 27$. Additionally, the curve is fitted with a gamma parameter of $\gamma = 2.26$. The fitted curve, following the methodology from Clauset et al. (2009), using the Goodness of Fit test, and compared against ten thousand proposed

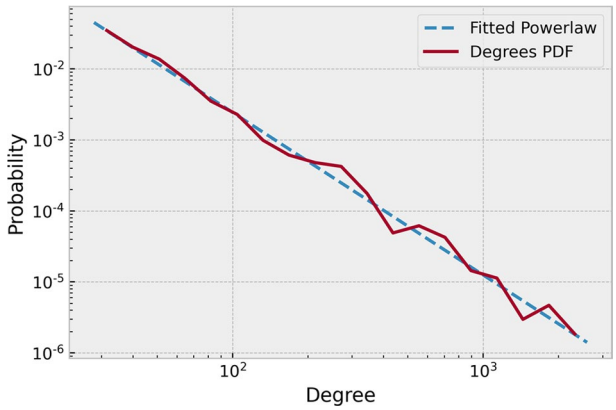

Fig. 3 The Powerlaw curve fitted onto the degree distribution with a parameter of $\gamma = 2.28$

artificial datasets, yields a p-value of 0.2114. Given this value surpasses 0.1 (as proposed by Clauset et al. 2009), the null hypothesis of the network not being scale-free will not be rejected.

Based on the following observations, it can be claimed that the Lightning Network is a scale-free network:

1. The network features powerful hubs that possess significant ownership of network connections.
2. The Powerlaw curve fits statistically onto the degree distribution. Additionally, the parameter gamma of the fitted curve equals 2.26, falling within the acceptable range of (2, 3).
3. Considering an average shortest path value of 3.65 and comparing this network to similar scale-free networks in terms of quantity, it can be asserted that the ultra-small world property holds in the Lightning Network.
4. The distribution of clustering coefficient values concerning node degrees in the Lightning Network is skewed, confirming its scale-free nature.
5. The assortativity coefficient in the Lightning Network is negative and significantly deviates from zero (neutral). This negative value also indicates the network's scale-free nature.

Another metric that can be employed to assess data heterogeneity is the Gini coefficient (Sen et al. 1997). This measure is commonly used in economics to gauge inequality levels. In this current study, we will employ this metric to evaluate the inequality of degree distribution among nodes in the Lightning Network. The Gini coefficient ranges between 0 and 1, representing absolute equality and absolute inequality in the examined data, respectively. The computed Gini coefficient value for the degree distribution in the Lightning Network is 0.7646.

Preferential attachment criteria

A new node joining the Lightning Network needs to establish payment channels with other network nodes. These channels enable the node to conduct off-chain multi-hop payments through neighboring nodes. The decision of a node regarding which set of nodes to establish payment channels with will result in changes to the topology of the Lightning Network.

From the perspective of a new node, the created payment channels should facilitate payment conditions. The objectives of a new node operating in the network can be summarized as follows:

- The new node expects its payments within the network to be subject to minimal fees.
- The new node aims to have the shortest possible paths to other nodes in the network, minimizing the number of hops required for its payments.
- The new node anticipates successful payment experiences within the network. In other words, it desires the presence of available well-behaved and non-adversarial nodes along the payment routes and in its neighborhood. Additionally, a larger number of potential paths to the payment destination provides more options for selecting a payment route, increasing the likelihood of successful transactions.
- The new node expects the capacity of its payment routes from the source to the destination to be maximized. This allows the node to transfer larger amounts without the need to split them or incur additional costs.

To address the goal fulfillment of new nodes joining the Lightning Network, it is necessary to examine network features that can have an impact on addressing or creating the stated requirements. The criteria that can be effective in achieving these goals encompass a wide range of protocol characteristics and network topology in the Lightning network. Some sources, such as Lightning Labs (2019), have introduced criteria for node scoring in the Lightning Network and have even implemented them in practice. The criteria that can effectively contribute to the fulfillment of the stated goals are as follows:

1. Channel Lifespan: Nodes with older and more established channels are more likely to exhibit reliable and non-adversarial behavior. To assess this criterion, it is necessary to monitor the channel status of nodes in the network over a long period of time.
2. Availability History: Nodes that have higher availability and remain consistently active will be more trusted. This criterion can also be estimated through continuous network monitoring to verify the nodes' active status.
3. Total Channel Capacity: Nodes with higher total payment channel capacities are more reliable options for establishing channels and performing routing. A high channel capacity of a node indicates significant investment in the Lightning Network, which creates an incentive for the node to behave favorably.
4. Average Channel Capacity: The total channel capacity alone cannot enhance the quality of a node, as a malicious node may create numerous low-capacity channels with low costs to match the total capacity of a well-behaved node with fewer chan-

nels but higher capacities. Therefore, considering the average capacity of channels (i.e., capacity per channel) can be an effective criterion for node scoring.

5. Number of Channels: In general, a node with a higher number of channels can be a suitable option for routing in the network. Connecting to such a node can shorten payment paths. These nodes, known as hubs, increase network connectivity and reduce the average shortest path in the network.
6. Quality of Neighbor Nodes: A node with high-quality neighboring nodes is more likely to possess high-quality itself. This can be a result of a node needing to connect to nodes with high quality in routing to improve its own efficiency. This holds true for quality metrics such as high capacity, low fees, and similar factors. This quality criterion can be aligned with the definition of Eigenvector centrality (Bonacich 1987). In other words, the Eigenvector centrality values of nodes can be considered as effective features in assessing the quality of Lightning Network nodes.
7. Proximity to Payment Destination Nodes: Nodes that are closer to payment destinations will be considered higher quality in terms of reducing multi-hop payment costs. Nodes that are more frequently traversed in payment paths are better options for establishing links. This criterion aligns with the definition of Betweenness centrality (Freeman 1977). In other words, nodes with higher Betweenness centrality are more suitable for establishing payment channels with them.
8. Total Channel Fees: Nodes with lower total channel fees can be suitable options for establishing links or routing. This becomes particularly important when a node needs to route its own payments through longer paths or with higher amounts. In the Lightning Network, fees are divided into "base fee" and "fee rate." The base fee is a constant amount calculated as the initial fee for any payment, followed by the fee rate calculated based on the payment amount. A low fee rate is desirable for larger payment amounts, and a low base fee is desirable for smaller payment amounts.
9. Average Channel Fees: A node with a high number of payment channels and minimal fees can have a total fee equal to a node with fewer channels but higher fees. Therefore, in addition to considering the total channel fees of a node, the average channel fees (i.e., fees per channel) should also be considered as an effective criterion for assessing the node's quality.

It is expected that a Lightning client, after running and receiving the network's topology, provides channel creation and micropayments services. The mentioned criteria for fulfilling the objectives of new nodes in the Lightning network's preferential attachment process can be provided to clients by third-party centers in the network. However, due to the centralized nature of this approach, it may potentially introduce vulnerability. In this study, a proposed strategy is presented where a Lightning client can independently identify suitable nodes for joining, without relying on third-party centers. Therefore, the use of criteria that require long-term network monitoring is disregarded. These criteria include (1) the lifespan of network nodes' channels and (2) the accessibility history of network nodes. In conclusion, the criteria in Table 1 will be utilized as scoring criteria for network nodes in the Lightning network.

Table 1 Proposed lightning nodes quality scoring criteria

Criterion	Feature name
Total channel capacity	total_capacity
Average channel capacity	capacity_per_channel
Number of channels	degree
Quality of neighbor nodes	eigenvector_centrality
Proximity to payment destination nodes	betweenness_centrality
Total channel fee bases	total_fee_base
Average channel fee bases	fee_base_per_channel
Total channel fee rates	total_fee_rate
Average channel fee rates	fee_rate_per_channel

Labeling

The Lightning Network data does not consists of labeled qualitative data about nodes. In fact, the decision to label each node depends on the problem's content and desired objective. In the current study, it is necessary to evaluate and label network nodes based on their quality and reliability in the process of preferential attachment of new nodes into the network. This labeling should be done considering the nine proposed criteria in Sect. "Preferential attachment criteria".

Given the large number of nodes in the Lightning Network (18,000 nodes), they need to be divided into groups with common and similar characteristics for batch labeling. By examining the preferential attachment criteria and examining the characteristics of a subset of network nodes, it is believed that clustering nodes based on these features may be possible.

The Lightning Network is composed of multiple and diverse nodes, each engaging in network activities with a specific goal. For example, some nodes in the Lightning Network aim to generate income from payment fees. These nodes strive to increase their centrality in the network and have a strong inclination to establish payment channels with smaller and newer nodes in the network. Additionally, there are some nodes in the network that host a limited number of payment channels but with very high capacities. These nodes have set low fee rates for their payment channels but show little interest in connecting with other nodes.

An example of collective behavior of nodes in the Lightning Network is the Zion Social Network (Rezvani 2021). Zion is a decentralized social network based on the Lightning Network that allows its users to engage in content production without the need for a centralized intermediary and without incurring fees. It provides a social environment without advertisements, censorship, and surveillance. This social network implements an economic model based on content payment, creating the necessary incentives for its users to participate in network activities. Each user in this social network is either an owner of a Lightning full-node or has rented a Lightning full-node.

By examining the Lightning mainnet, the user nodes and control nodes of the Zion social network can be identified as distinct star and cluster structures. The components of this network can be distinguished in the Lightning network by following node aliases such as "zion-*" or "*-m" for users and control nodes labeled as "Morpheus_1," "Trinity_1," "Neo_1," "NEO_2," and "ZION_TRINITY_2." Fig. 4 illustrates the placement of

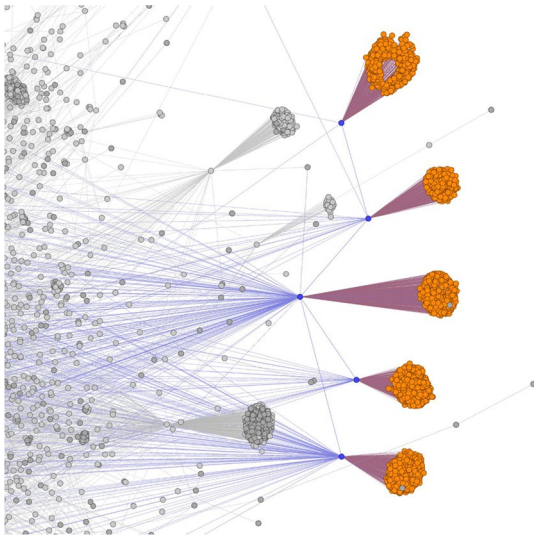

Fig. 4 Placement of Zion social network nodes in the Lightning network topology. The orange nodes represent user nodes, and the blue nodes represent control nodes of the Zion network

Table 2 Insights into the preferential attachment criteria for control and user nodes of the Zion network

Metric	Zion control nodes	Zion user nodes
total_capacity	High	50000 or 70000
capacity_per_channel	High	50000 or 70000
degree	High	1
eigenvector_centrality	–	–
betweenness_centrality	High	0
total_fee_base	0	0 or 1000
fee_base_per_channel	0	0 or 1000
total_fee_rate	High	0 or 1
fee_rate_per_channel	High	0 or 1

some nodes of the Zion social network in the Lightning network's topology. The orange nodes represent user nodes, while the blue nodes represent the control nodes of this network. The number of Zion social network nodes examined in this research within the Lightning network exceeds 2,000 (equivalent to 12.5% of the network nodes), all of which are located within the specified topological pattern.

By examining the preferential attachment criteria for Zion network nodes, it is evident that the control nodes of this network in the Lightning network have high degrees, high capacities, zero base fees, and non-zero betweenness centrality values. The user nodes of the Zion network also have degree 1, capacities of either 50,000 satoshis or 70,000 satoshis, zero or 1,000 milli-satoshis base fees, zero or 1 milli-satoshis fee rates, and zero betweenness centrality values. In other words, all control nodes of the Zion network exhibit similar criteria values, and the user nodes of this network follow the same pattern. Table 2 illustrates the derived criteria from the Zion network nodes. Considering that the social network nodes in the Lightning network

do not accept any nodes outside the social network and it is not possible to establish payment channels with any of the Zion nodes, all Zion nodes can be considered as a negative label in establishing communication channels with new nodes under the preferential attachment process.

To identify groups and label nodes with similar behaviors, it is necessary to employ an unsupervised machine learning approach. In this study, after analyzing and transforming node features, we will reduce and make the data dimensions independent, followed by performing clustering operations. Finally, we will examine the resulting clusters and assign appropriate labels to each cluster.

Features preparation

The available dataset consists of nine different features with various dimensions and types. By examining the distribution of these features, it is evident that all of them follow a skewed distribution. Therefore, to address this issue, a logarithmic transformation will be applied to the features, transferring them to a new space where the skewness is minimized. After applying the logarithmic transformation to the features, it is necessary to perform data scaling operations. Given that the distribution of the data properties in this research deviates from the normal distribution and sometimes exhibits skewed distributions, the min-max scaling method has been utilized for scaling.

Principal Component Analysis (PCA) is one of the most well-known machine learning methods for dimensionality reduction and decorrelation of features (Jolliffe 2002). When the explanatory variables of a phenomenon are correlated with each other, the presence of redundant information among the variables can introduce complexity in future analysis and modeling. PCA is a technique that linearly (or nonlinearly in the case of nonlinear PCA) maps a set of explanatory variables to a set of uncorrelated variables. The purpose of this mapping is to achieve independence among the explanatory variables and enable dimensionality reduction of the data. In this study, after the initial preparation of node features in the network, PCA was employed to eliminate the correlation among the existing variables. Given the small number of data features (9 in total), dimensionality reduction was not pursued to avoid losing information present in the variables and to maintain consistency in the performance of the model with this number of features.

Clustering

After the preparation of features, it is necessary to cluster and identify data groups in the space of the components generated by PCA analysis. Considering the available data and their type, rather than aiming to find distinct clusters, the objective is to identify data aggregation centers to discover similar behavioral patterns. In other words, the purpose of clustering in this research is to categorize nodes based on their feature patterns for labeling. Therefore, clustering methods that emphasize cluster independence, such as agglomerative clustering (Gowda and Krishna 1978), are not desirable. For this purpose, the unsupervised K-means clustering method has been used (Hartigan and Wong 1979). This method not only focuses on finding data aggregation centers but also allows control over the number of clusters.

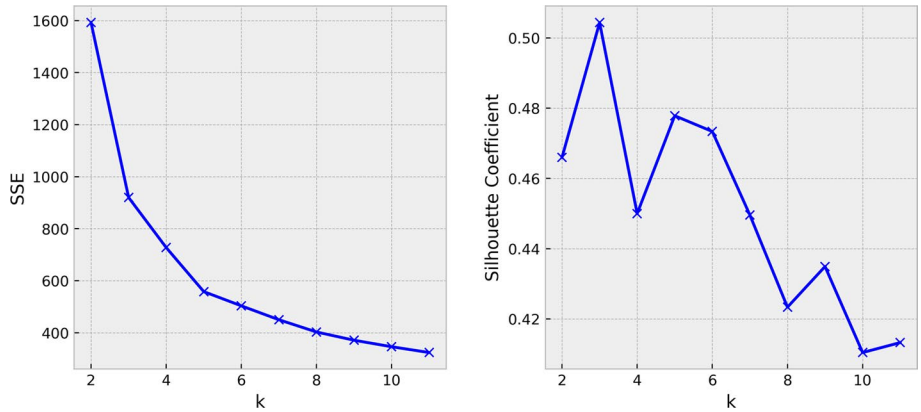

Fig. 5 Values of SSE and Silhouette Coefficient for performing clustering using the K-means method with different numbers of clusters (k) ranging from 2 to 11

To determine the optimal number of clusters, qualitative clustering criteria can be used. The sum of squared errors (SSE) is an indicator of the total squared Euclidean distance between each point and its nearest cluster centroid. The K-means clustering method also aims to minimize this value. One method for determining the optimal number of clusters is finding the knee-point (or elbow-point) in the SSE values plotted against the number of clusters. It is evident that increasing the number of clusters in the K-means algorithm will always lead to a decrease in SSE, as a higher number of centroids reduces the distances between points and their nearest centroids compared to a lower

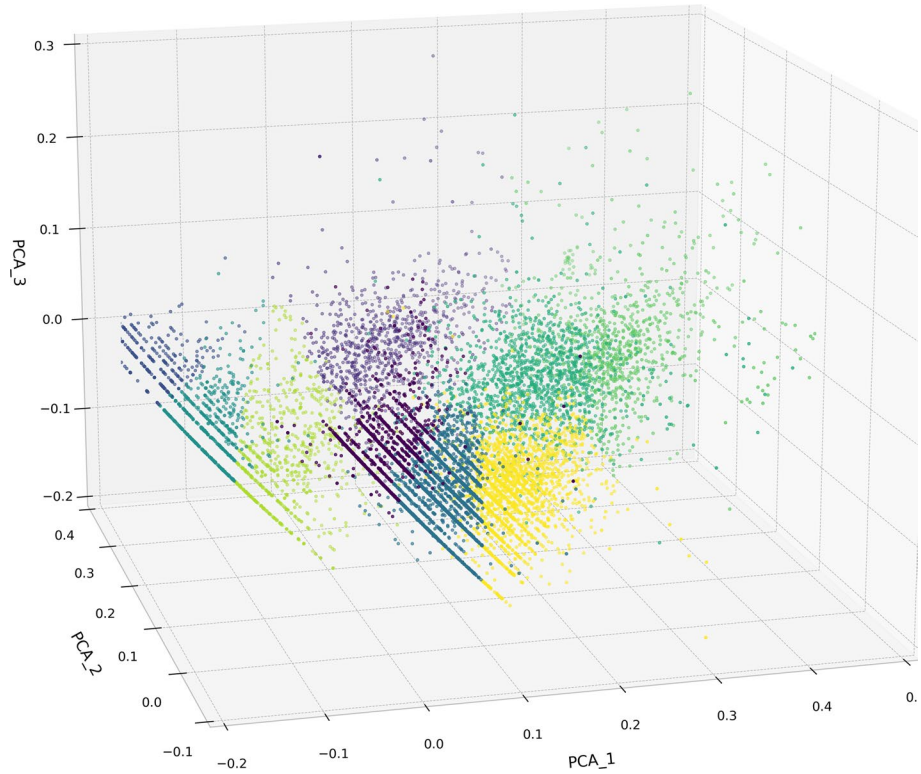

Fig. 6 Clusters obtained using the K-means method on the spatial distribution of the three principal components from PCA analysis

number of centroids. The desired points on this curve are where the SSE values start decreasing linearly. Another method for determining the optimal number of clusters is using the Silhouette Coefficient (Rousseeuw 1987). This coefficient is a value in the range of $[-1, 1]$ that indicates the quality of the performed clustering; a value of 1 indicates well-separated clusters, a value of 0 indicates overlapping clusters, and a value of -1 indicates inappropriate clustering.

Figure 5 shows the SSE and Silhouette Coefficient values for clustering performed with the K-means method using different numbers of clusters (ranging from 2 to 11). The knee-points in the SSE plot correspond to 3 or 5 clusters, which are consistent with local maxima in the Silhouette Coefficient plot. Therefore, these numbers of clusters can be considered as the optimal number of clusters for clustering using the K-means method. In this research and in this section, the goal of clustering is to create independent groups for the labeling process. Hence, while a higher number of clusters may lead to similarities between different cluster properties, it will also increase the accuracy of the labeling process. Based on this, the next local maximum in the Silhouette Coefficient plot, considering the negligible difference in the coefficient values at this point compared to the previous values, can be utilized (9 clusters). Figure 6 illustrates the resulting 9 clusters obtained by applying the K-means method to the spatial distribution of the three principal components derived from PCA analysis with the highest explained variability.

Qualitative labeling

Figure 7 illustrates clusters of nodes with common characteristics on the Lightning Network. As expected, nodes with similar topological behavior and positions are classified into similar clusters.

For example, all the user nodes of the Zion network are categorized into clusters number 2 and 7 (indicated by blue and green nodes in Fig. 7). Additionally, all the control nodes of the Zion network are placed in cluster number 5. Although, from a topological perspective, the control nodes of the Zion network have a spatial resemblance to the hub nodes, such as "ACINQ," "1 ML.com node ALPHA," "CoinGate," and "WalletOfSatoshi.com," which are located in cluster number 8 (represented by bold nodes in Fig. 7). The reason behind this lies in the nine selected features during the clustering process, which includes non-topological properties as well, leading to a proper distinction between the cluster of Zion network control nodes and the cluster of the Lightning network hub nodes.

In order to better identify suitable clusters for positive labels (including desirable conditions for preferential attachment) and negative labels (including undesirable conditions for preferential attachment), we will investigate the 9-dimensional features between the clusters. Due to the skewed distribution of these features, we will also use the "logarithmic mean" measure for comparing them. Table 3 presents these values for each of the features and each cluster of nodes in the Lightning Network using a linear binning method with three levels: "low," "medium," and "high," represented by styles like "bold," "italics," and "bolditalics," respectively. This facilitates distinguishing between different clusters based on the level of their features, making it easier for labeling purposes.

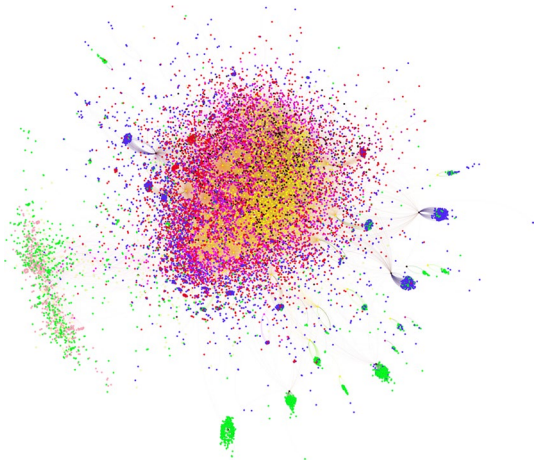

Fig. 7 Clusters of groups of nodes with common characteristics on the Lightning Network (results from Sect. "Clustering")

Table 3 The logarithmic mean of node features for clusters in the Lightning Network

Metric (Mean Log)	Cluster0	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5	Cluster6	Cluster7	Cluster8
total_capacity	**8.9614**	***16.2170***	**11.3402**	*14.2657*	***15.8520***	***16.8999***	*13.7407*	**11.2011**	***18.8869***
capacity_per_channel	**8.2357**	***14.2473***	*11.0977*	***13.2712***	***13.9965***	***14.3370***	***13.0648***	*11.0613*	***15.0746***
degree	**1.1830**	*2.1365*	**0.8448**	**1.3935**	*2.0388*	*2.6779*	**1.1283**	**0.7844**	***3.8415***
eigenvector_centrality	**1.05e−5**	**3.89e−3**	**6.25e−4**	**1.14e−3**	**3.57e−3**	*8.11e−3*	**1.40e−3**	**2.04e−4**	***1.94e−2***
betweenness_centrality	**5.40e−5**	**1.06e−4**	**5.99e−6**	**1.17e−4**	**5.56e−5**	**5.62e−4**	**1.45e−5**	**5.38e−6**	***1.79e−3***
total_fee_base	**0**	***8.6813***	***7.1112***	**0.2083**	***8.4946***	**0.3154**	***7.4847***	**0.0178**	***10.2157***
fee_base_per_channel	**0**	***6.7134***	***6.8689***	**0.1387**	***6.6418***	**0.1173**	***6.8093***	**0.0148**	***6.4317***
total_fee_rate	**0**	**2.3359**	**0.9299**	**1.1562**	***7.0485***	***7.4905***	**1.2265**	**0.2709**	***8.6475***
fee_rate_per_channel	**0**	**0.8802**	**0.7790**	**0.7978**	***5.2020***	***4.9666***	**0.7944**	**0.2595**	***4.9002***

The levels "low," "medium," and "high" are represented by styles "bold," "italics," and "bolditalics," respectively

In the following and during a blacklist policy, we proceed to assign a negative label to undesirable clusters, assuming initially that all clusters are preferable for preferential attachment. This policy transforms the preferential attachment process into a non-discriminatory procedure, increasing the chances for nodes to be selected as preferential attachment destinations for new nodes in the Lightning network.

As mentioned earlier in this section, nodes in the Zion network are those with a negative label for preferential attachment; they will only interact within the Zion network and won't establish channels with new nodes in the Lightning Network. Hence,

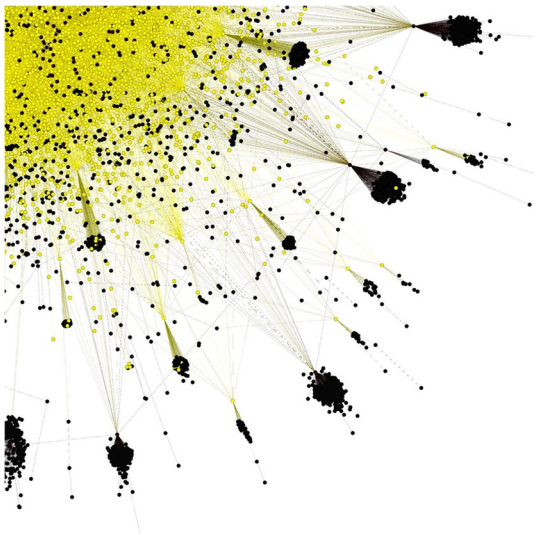

Fig. 8 Star-shaped colonies in the Lightning Network, described by clusters 2, 7, and 5, highlighted in black

the user and control clusters of nodes in the Zion network (clusters 2, 7, and 5) are considered as negative labels. These clusters not only identify social nodes in the Zion network but also reveal star structures on the network's periphery, representing isolated colonies - structures with dependent nodes primarily having low-fee payment channels and hubs (star centers) responsible for supporting the dependent nodes. These structures resemble patterns observed in the Zion network, indicating a lack of inclination of these nodes to establish new payment channels with newly joined anonymous nodes. Examples of these structures can be seen in Fig. 8.

The nodes present in cluster 0 are nodes with a small number of connections at the network periphery, having low degrees, low capacity, and all with a fee rate of 0. By examining the placement of these nodes in the network topology, it becomes evident that they are created with the purpose of establishing private micropayment channels and personal ownership, as indicated by their limited number of payment channels and the absence of fees received. This implies their reluctance to actively participate in the Lightning Network and operate in the shadow. Therefore, these nodes will also be identified with a negative preferential attachment label. Figure 9 illustrates the nodes of cluster 0 at the network periphery (highlighted in black).

After clustering, the number of nodes labeled as positive will be 8906, and the number of nodes labeled as negative will be 9227.

The scoring model

After labeling the Lightning Network nodes using an unsupervised learning method, the desirability of selecting each node as a target for the preferential attachment process is determined. While the desirability has been specified for existing nodes in the network and for a specific snapshot, the network's evolving process may cause the node features to change. Additionally, newly joined nodes in the network can also be involved as target nodes in the preferential attachment process based on their own features. Hence, there is a need to design a predictive model for the desirability of preferential attachment based

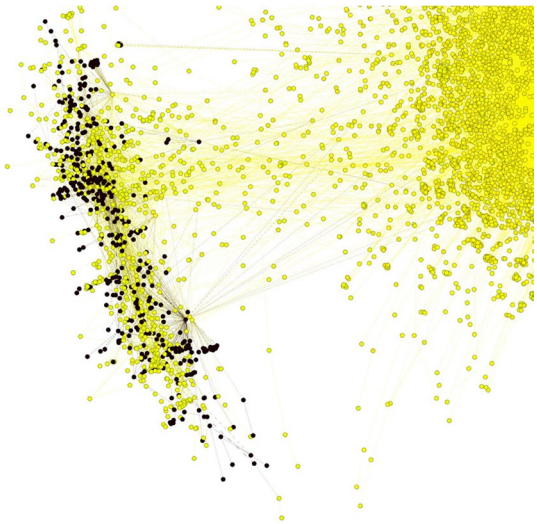

Fig. 9 Nodes with a small number of connections in cluster 0 at the network periphery, highlighted in black

on the labels of the current network nodes. This model takes a node with its features as input and provides the level of desirability for that node as output.

In the current research, the desirability in the preferential attachment process is not considered as a deterministic element. In other words, when a node intends to join the Lightning Network, the selection of target nodes to establish payment channels with them can be determined with a probability coefficient of their desirability. Nodes with higher desirability will have a higher probability of being selected, while nodes with lower desirability will have a lower probability. This desirability will be represented by a continuous score and will not be limited to zero or one.

Model formation and training

In this research, similar to credit-scoring methods used in insurance and banking systems, the logistic regression model is employed (Cox 1958). Logistic regression (or logit model) is a model based on the logarithm of odds, which evaluates the probability of a positive or negative outcome of the dependent variable. Various methods exist for calculating coefficients (β parameter) in this model, with the most well-known being the maximum likelihood method.

The logistic regression model is used for modeling data with a categorical dependent variable. After forming the model and determining the coefficients, new data can be fed to the model, and after calculating the probability of each outcome by the model, based on a specified threshold, the classification of each input data can be determined. This threshold is commonly set to 0.5. Due to the need to determine the desirability values of nodes in this research, the model will be used to calculate the logit value of the input nodes. Unlike probability, the logit (e.g., log-odds) can take any real value. This allows this value to serve as an unscaled score and a selection coefficient in the preferential attachment process for a node.

Table 4 Performance evaluation metrics of the logistic-regression model on the test dataset

Accuracy	96.46%
Precision	96.64%
Recall	96.14%
F1-Score	96.39%

To form the logistic regression model, we first randomly select half of the node data and exclude them from the training process as the test data set. These 50% of the data will be used as the basis for evaluating the model at the end. The selection of training data is done randomly but with a stratified strategy. This strategy ensures a similar distribution of labels in both the original data and the selected data.

After selecting the training data, the logistic regression model is learned using the k-Fold cross-validation method with $k = 10$. In this process, the model is trained 10 times. The learning process can be summarized in the following steps:

- Shuffle the data randomly.
- Divide the data into $k = 10$ groups, ensuring a similar distribution of labels in the groups.
- For each data group:
  - Select the group as the test data.
  - Select the other groups as the training data.
  - Form the model on the training data and evaluate the model based on the test data.
  - Keep track of the evaluated performance metric of the model.
- Estimate the overall performance based on the evaluation values of each step.

The evaluated metric during the cross-validation process is Accuracy. This metric indicates the proportion of correctly predicted values relative to the total available data. The average evaluated Accuracy of the model is 96.40%, with a standard deviation of 0.59%.

The performance evaluation metrics of the model on the test dataset include Accuracy, Precision, Recall, and F1-Score. These values can be observed in Table 4.

Scores calculation

By forming the logistic regression model, for each node in the network with specific features, the value of logit (logarithm of odds) can be calculated. Assuming we represent the coefficients of the formed logistic model with the vector $\beta = [\beta_n, \beta_{n-1}, ..., \beta_1]$ and the model's intercept with β_0, and let X be the data matrix such that each row contains the features of a node and the columns represent the features of the nodes (an $n \times m$ matrix where n is the number of nodes in the network and m is the number of their features), we will have:

$$L = X.\beta^T + \beta_0$$

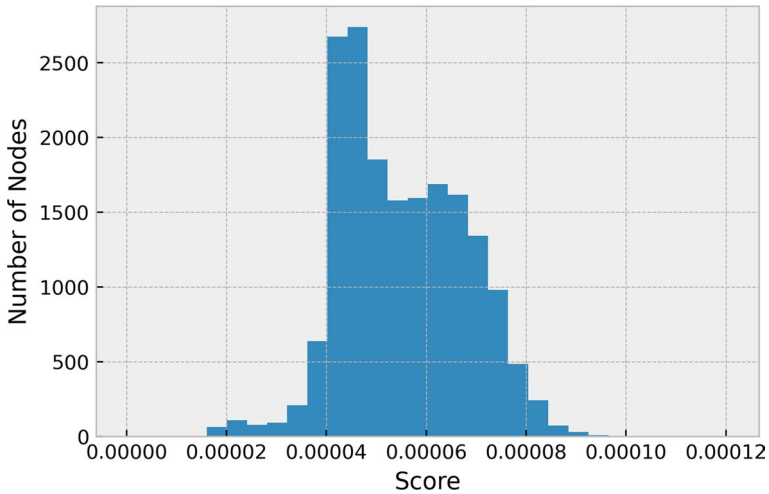

Fig. 10 The distribution of computed scores for network nodes for use in the preferential attachment process

where L is a vector with a length of n and represents the logit values of the network nodes.

To calculate the final scores of the nodes, it is sufficient to compute the probabilities of the values in the L vector using the following method:

$$S = \frac{L - \min_i\{L\}_{i=1}^{n}}{\sum_{i=1}^{n} L}$$

where the vector S is a vector of length n, indicating the likelihood of each node's score relative to the scores of other nodes. The sum of the values in vector S will be equal to 1, in other words, the computed scores vector of the nodes is a probability vector.

Figure 10 illustrates the distribution of scores for the network nodes. Based on the features, clustering has been performed, and a model has been constructed, resulting in a non-skewed distribution of these scores. These scores will be used as probability coefficients in the process of Lightning Network's preferential attachment, estimating and combining the features of each node towards its desirability for preferential attachment.

Empirical experiment

The preferential attachment process used in this research is defined by drawing inspiration from the Albert and Barabási (2002) and employing the calculated scores as follows:

"The Lightning Network starts in the initial state s_0 with a total of n_0 nodes. At time t_i, a node with identifier i is added to the network. This node establishes e_i payment channels with existing nodes. The probability of establishing each of these payment channels with any of the network nodes is proportional to their scores at time t_i." In other words:

$$p_j = S_{j_{t_i}}$$

where p_j represents the probability of establishing a payment channel k with node j in the network, and $S_{j_{t_i}}$ is the calculated score of node j (Sect. "Scores calculation") at time

t_i. Additionally, there is a constraint preventing the establishment of duplicate payment channels when a node joins the network.

The stated preferential attachment process empowers new nodes in the network to choose the desired number of payment channels they want to establish. From another perspective, when a new Lightning client node joins the network, it will be provided with a sorted list of suitable nodes for establishing new payment channels with them. The scores of nodes are calculated in real-time upon request, and then the client is presented with a prioritized list of nodes (vector S) with the highest scores. Afterward, the user can select the desired nodes to create payment channels with and join the network accordingly. All of these processes can be facilitated by the Lightning client's Autopilot module, which, depending on the features provided to the user, can implicitly or explicitly guide them in selecting suitable targets.

We will then simulate the preferential attachment strategy with calculated scores. For this purpose, the existing snapshot of the Lightning Network (Sect. "Dataset description") is used as the initial state of the network (s_0). The simulation process involves attaching 5000 new nodes to the network. To determine the non-topological propertiess of these new nodes and their payment channels upon joining the network, we use the Bootstrapping strategy (Efron and Tibshirani 1994). As mentioned in previous sections, Lightning Network nodes are grouped with similar behavioral characteristics; hence, randomly and independently selecting all features for a new node would be unrealistic. Therefore, Bootstrapping is employed in this simulation for this purpose. Additionally, using a constant value for the degree of the new nodes in the network would be unrealistic and would lead to the exclusion of nodes with low degrees. In general, the simulation process for each step of preferential attachment can be described as follows:

1. Extract and compute the features of network nodes.
2. Calculate the preferential attachment scores of network nodes using the learning model.
3. Randomly select a reference node from the network for its features.
4. Based on the number of payment channels of the reference node, choose destination nodes from the list of network nodes with probabilities based on their scores.
5. Create a new node.
6. For each payment channel of the reference node, create a new payment channel with each destination node from the new node with the following features:
    - Capacity equal to the capacity of the selected channel of the reference node.
    - Base fee on the side of the new node equal to the base fee of the selected channel of the reference node.
    - Fee rate on the side of the new node equal to the fee rate of the selected channel of the reference node.
    - Base fee on the side of the destination node equal to the base fee of a randomly selected edge from the destination node.
    - Fee rate on the side of the destination node equal to the fee rate of the same randomly selected edge from the destination node.

Table 5 Topological properties of the simulated network

Property	$\mathcal{G}_{degree}$	$\mathcal{G}_{betweenness}$	$\mathcal{G}_{score}$	$\mathcal{G}$
Number of Nodes	23133	23133	23133	18133
Number of Channels	123255	128373	120249	76725
Average Degree	10.65	11.09	10.39	8.46
Density	0.00046	0.00047	0.00044	0.00046
Diameter	11	6	8	12
Average Shortest Path	3.4188	3.3149	3.6496	3.65
Number of Bridges	10814	10736	1321	8560
Clustering Coefficient	0.1096	0.1650	0.0507	0.1231
Assortativity	−0.1824	−0.2445	−0.1302	−0.195
Top 1% Ownership	34.38%	40.08%	26.91%	34.97%
Powerlaw Gamma	2.1545	2.0413	2.4001	2.26
Gini Coefficient	0.7833	0.8040	0.6330	0.7646

For comparison and evaluation of the results, in addition to performing the simulation steps for preferential attachment based on the calculated scores in this study, simulations have been conducted for two other strategies: (1) preferential attachment based on node degrees, and (2) preferential attachment based on Betweenness centrality. These strategies are among the approaches used in autopilot tools for Lightning clients, which have been discussed in Sect. "Lightning autopilot". The difference in simulation steps for these two strategies compared to the described steps lies in step (2) of the preferential attachment simulation process, where the node scores are calculated based on node degrees according to formula 1 and based on Betweenness centrality according to formula 2.

$$s_{\text{degree}_i} = \frac{\text{deg}_i}{\sum_{k=1}^{n} \text{deg}_k} \tag{1}$$

$$s_{\text{betweenness}_i} = \frac{\text{betweenness}_i}{\sum_{k=1}^{n} \text{betweenness}_k} \tag{2}$$

Results and discussion

Table 5 illustrates the topological characteristics of the networks resulting from each of the preferential attachment strategies after attaching 5000 new nodes. The resulting networks include: $\mathcal{G}_{degree}$ (preferential attachment based on node degrees), $\mathcal{G}_{betweenness}$ (preferential attachment based on Betweenness centrality scores of nodes), and $\mathcal{G}_{score}$ (preferential attachment based on the calculated scores of nodes). Additionally, the Lightning network in its initial state (s_0) has been examined and denoted as $\mathcal{G}$ for feature comparison.

As evident in Table 5, the network $\mathcal{G}_{score}$ exhibits more balanced topological characteristics compared to networks $\mathcal{G}_{degree}$ and $\mathcal{G}_{betweenness}$, as well as the initial network $\mathcal{G}$. In other words, the network formed using the preferential attachment strategy with calculated scores represents a network with a more homogeneous distribution of degrees among its nodes, which not only avoids increasing the skewness observed in the initial

network but also leads to a more decentralized allocation of network resources compared to the current preferential attachment strategies in Lightning's Autopilot.

As the first investigated property, it should be noted that the value of the Powerlaw distribution parameter (gamma) in the network $\mathcal{G}_{score}$ is higher than the initial network and the other two networks. This signifies an increased steepness in the Powerlaw curve on the logarithmic axes of the degree distribution, implying a more balanced division of degrees among the intermediate nodes of the network. Additionally, it is worth mentioning that the gamma values in the preferential attachment strategies based on node degrees and Betweenness centrality not only remain low compared to the original network (which itself possesses a high concentration) but also tend to approach critical values, i.e., star-like centralized structures. This phenomenon has also been observed and explained in previous works such as Tsiotas (2020) and Topirceanu et al. (2018).

The examination of other topological properties of the compared networks aligns with the discovered evidence. The Gini coefficient in the network $\mathcal{G}_{score}$ is reduced by 17% compared to the original network $\mathcal{G}$, indicating a more equitable distribution of degrees among the nodes when employing the preferential attachment strategy based on calculated scores. In contrast, the other two strategies used in Lightning's Autopilot result in a 2.5% increase in the Gini coefficient in the degree-based strategy and a 5% increase in the Betweenness centrality-based strategy. In other words, the current strategies used in Lightning's Autopilot lead to a growing inequality in resource ownership by the hubs.

Furthermore, the investigation of top 1% ownership among the four existing networks also illustrates a more equitable distribution of degrees in the simulated network using the proposed strategy in the current research. The degree ownership in the network $\mathcal{G}_{score}$ reduces the power of 1% high-degree hubs by 20% compared to $\mathcal{G}$. Conversely, in the current two strategies of Lightning's Autopilot, this ownership either increases or remains unchanged. Figure 11 represents the cumulative percentage of node degrees relative to the total network degrees for nodes ranked in descending order based on their degree and for the four examined networks. The decrease in degree ownership by the hubs is evident in the network simulated with preferential attachment strategy based on calculated scores.

In further investigations, the number of bridges in the resulting networks using the mentioned strategies can be highlighted. The increase in the number of network bridges implies greater vulnerability to targeted attacks, particularly those aimed at payment channels. The removal of any of the network bridges leads to an increase in network components and its disconnection, thereby rendering the current Lightning network more susceptible to attacks or the deactivation and locking of payment channels. The preferential attachment strategy based on calculated scores notably reduces the number of network bridges. If nodes with a degree of 1 are not considered, the count of bridges decreases from 174 in the original network to 13 in the porposed evolved network. Regarding the total count of all network bridges, even with the addition of 5000 new nodes to the network, a reduction of 86% has been observed. This is in contrast to the current autopilot strategies where the number of bridges is expected to increase significantly.

Other values of network properties under examination, such as clustering coefficients or assortativity metrics, signify the moderation of inequalities in the simulated

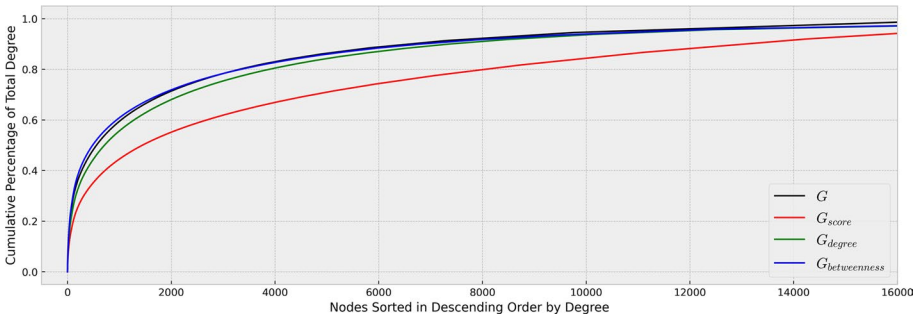

Fig. 11 Cumulative percentage of node degrees relative to the total network degrees for nodes ranked in descending order based on their degree and for the four examined networks

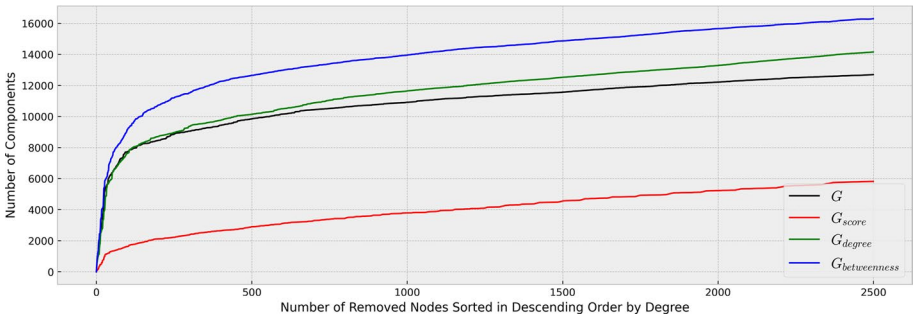

Fig. 12 Variations in the number of network components during the process of hub removal across the four examined networks

network using the proposed autopilot strategy in the current research. It is important to note that although the average shortest paths in the proposed autopilot strategy have shown a slight increase compared to the two current strategies, this value has remained unchanged in relation to the average shortest path in the original Lightning network graph, even with the increase in the number of nodes in the network. In other words, transitioning to the autopilot strategy using calculated scores for preferential attachment will not impose considerable cost on the process of multi-path payment routing in the Lightning network.

In the realm of network resilience against targeted attacks, three distinct attacks have been conducted on the topology of both simulated networks and the Lightning network. The first attack involves removing 2500 hubs from the network, prioritizing those with the highest degree. Figure 12 illustrates the changes in the number of network components during the process of hub removal. A resilient network against topological attacks such as targeted attacks should be able to withstand an increase in its components during the attack process. Based on the obtained results, the simulated network with preferential attachment based on calculated scores demonstrated the highest resilience against targeted attacks on the core hubs compared to other networks. As previously mentioned, the highest vulnerability in this regard belongs to the evolved network based on betweenness preferential attachment, although the resilience of the network created using degree based preferential attachment autopilot strategy also significantly differs from the proposed strategy in the current study. Another assessable criterion during

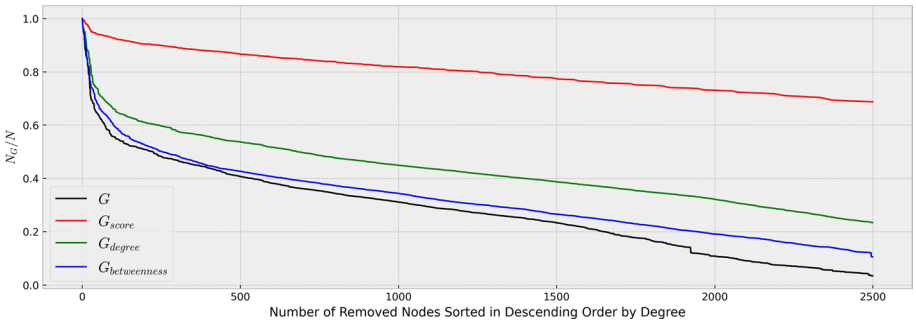

Fig. 13 Variations in the number of nodes of the largest network component relative to the total number of network nodes during the process of hub removal across the four examined networks

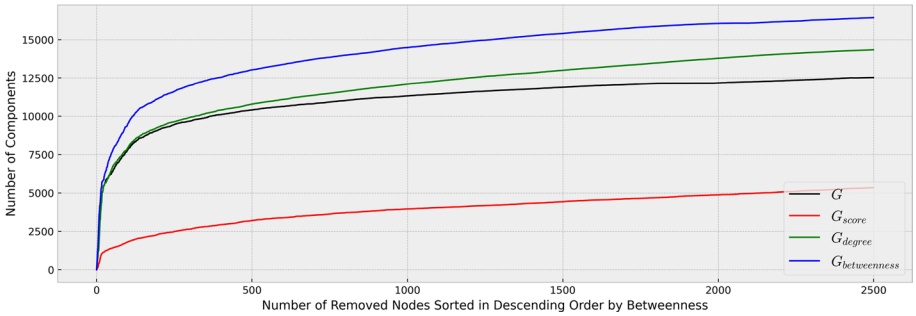

Fig. 14 Changes in the number of network components during the process of removing nodes with high centrality from the four examined networks

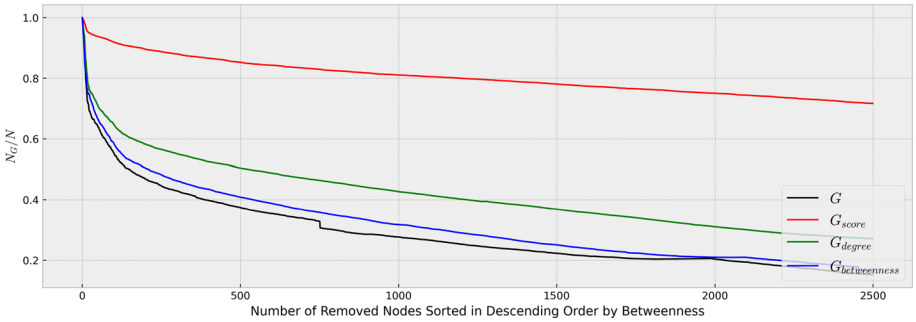

Fig. 15 Changes in the number of nodes of the largest component in relation to the total number of network nodes during the process of removing nodes with high centrality from the four examined networks

attacks is the ratio of the size of the largest component to the total network size, denoted as N_G/N. This criterion indicates the strength of the largest network component and its reduction signifies the weakening of the largest cohesive unit in the network. Figure 13 illustrates the changes in this criterion in the face of targeted attacks for the four existing networks, confirming previous analyses.

Another targeted attack examined involves targeting nodes with the highest Betweenness centrality in the network. Figure 14 illustrates the variations in the number of network components during the process of removing highly central nodes from the investigated networks, and Fig. 15 shows the changes in the N_G/N value during this

Fig. 16 Changes in the network's component count during the process of removing high centrality payment channels from the four examined networks

Fig. 17 Variations in the number of nodes of the largest network component relative to the total number of network nodes during the process of removing high centrality payment channels from the four examined networks

attack. In this attack scenario, the network created through the preferential attachment process based on Betweenness centrality has demonstrated the highest vulnerability. It's worth noting that the greater reduction in N_G/N in the main network is due to its smaller node count compared to the reference networks. In this type of attack as well, the network formed through the preferential attachment strategy with calculated node scores has exhibited the most effective resistance strategy.

Another simulated attack on the existing networks involves targeting payment channels within the Lightning network based on their Betweenness centrality. Betweenness centrality of a payment channel is the sum of the fraction of all-pairs shortest paths that pass through that channel. In other words, attacking a payment channel with high Betweenness centrality would increase the number of intermediate steps for multi-hop payments or potentially disrupt the network's connectivity. During the performed attack, a total of 5000 payment channels were removed from the network in order of their high centrality. Figures 16 and 17 respectively illustrate the changes in the number of components and N_G/N during the targeted attack on the payment channels. The results demonstrate the resilience of the simulated network with the adoption of a preference-based calculated score against disintegrity during the attack on payment channels.

While targeted attacks are desirable for attackers due to their strategic nature and cost-effectiveness, examining network resilience against random attacks can also serve as a benchmark for network evaluation. During the conducted random attack on the examined

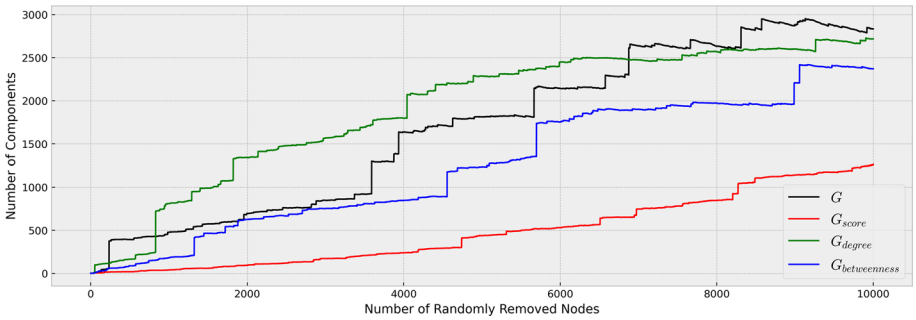

Fig. 18 Variations in the Number of Network Components during the Process of Random Node Removal across Four Investigated Networks

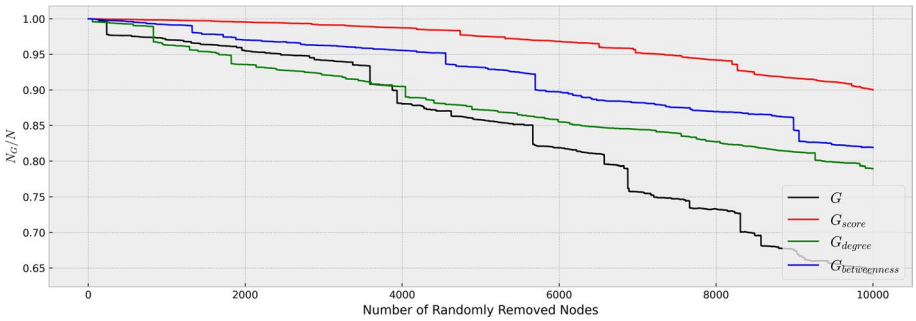

Fig. 19 Fluctuations in the Number of Nodes of the Largest Component of the Network Relative to the Total Number of Network Nodes during the Process of Random Node Removal across Four Investigated Networks

networks, 10,000 nodes were randomly removed from the network. Figures 18 and 19 illustrate the variations in the number of components and N_G/N during the process of random node removal from the four examined networks. Although it is expected that centralized networks would exhibit greater resistance against random attacks compared to distributed networks, with an increasing number of nodes removed from the constructed centralized networks (by preferential attachment using Betweenness centrality and degree centrality), these networks will also show a high susceptibility to random attacks. This phenomenon has been investigated in studies such as Crucitti et al. (2003), Bollobás and Riordan (2004), and Gallos et al. (2005). It should be noted that in the simulated centralized networks, the distribution of node degrees among the top percentiles is so skewed that even during a random attack process, the placement of network hubs among the attacked nodes is likely. The high centrality of these hub nodes in the network leads to significant network degradation upon the removal of any of these hubs, which increases network vulnerability compared to more distributed networks in the face of this type of attack.

Conclusion and future works

The aim of the conducted research has been to propose a strategy for enhancing the evolution process of the Lightning Network by modifying the criteria of preferential attachment. This strategy diverges from the current preferential attachment strategies of the Lightning Network's autopilot system, which primarily rely on metrics such as node Betweenness

centrality or their degrees. This departure has been prompted by the observation that the network tends to centralize over time under the influence of these strategies.

After evaluating the topological characteristics of the Lightning Network, the shared traits of network nodes were examined. Through an unsupervised process, node attributes were preprocessed, clustered, and then labeled. This labeling involved identifying nodes in the network that are suitable for establishing payment channels with newly onboarded nodes. Subsequently, a supervised process involved the introduction of a logistic regression model to classify network nodes based on the quality of each node. This calculated score was computed using the logarithm of odds ratios of features. The resultant calculated score represents the desirability of each node in the preferential attachment process suggested by the Lightning Autopilot system.

Finally, employing the proposed method, the network's evolution process was simulated and compared against the current strategies of the Lightning Autopilot system. The network resulting from the evolution based on the proposed preferential attachment strategy using calculated scores has exhibited topological characteristics closely resembling those of distributed networks. In comparison to similarly evolved networks employing current autopilot strategies, this network has demonstrated a significant redistribution of resources across the network. The Gini coefficient of the network has shown a reduction of up to 23%, and resources have been redistributed in a manner that the top 1% of hubs have had their channels reduced by up to 27%. This comparison also revealed the resilience of the resulted network against targeted attacks on nodes and payment channels, as well as random attacks in a multi-objective context. The enhanced distribution of the network generated through the proposed preferential attachment strategy acts as a deterrent or incurs higher costs for attacks such as griefing attacks, time dilation attacks (making eclipse attacks more difficult), route hijacking attack, various topological attacks, and even some privacy attacks.

So far, numerous studies have addressed the critical issue of centralization in the Lightning Network's topology. Many of these studies have consistently engaged in monitoring and analyzing the network's topology, highlighting its vulnerabilities and potential susceptibility due to the concentration of network resources. However, it should be noted that the number of studies proposing solutions or strategies to mitigate the concentration of the Lightning Network's topology is limited. Without close collaboration between Lightning Network developers and researchers in this field, the network may soon face serious risks and security threats.

The current research approaches this issue from both heuristic and empirical perspectives. Clearly, delving into this matter could also attract attention from an analytical and proof-based standpoint using network science methodologies. The analytical proof of the topological characteristics of the network, especially when involving a machine learning model as a score calculator in the preferential attachment process, can be a complex endeavor that could be pursued in further stages of this research.

It is anticipated that alternative machine learning methods in the process of clustering and modeling of this research could potentially enhance the results. However, the primary mission of this study has been to provide a more abstract presentation of the proposed idea. Future research endeavors could explore other clustering methods

or different machine learning models, coupled with fine-tuning of parameters and hyperparameters, aiming to improve performance and enhance the outcomes.

Although the simulation conducted in the current study draws inspiration from the bootstrapping process, it's possible to enhance the simulation process by introducing additional parameters that bring it closer to the real-world network evolution process. Moreover, the utilization of specialized Lightning Network simulation tools such as CLoTH (Conoscenti et al. 2018) can offer a more accurate simulation process along with a wider range of network attributes, such as payment flows within the network.

The focal point of the current research lies in the presence of the autopilot module in the Lightning Network. A prospective research proposal in this field could involve the presentation of methods for decentralizing the Lightning Network without relying on the autopilot module. Such methods could be based on social manipulation techniques to alter the incentive patterns of participating nodes in the network. These studies would need to incorporate methodologies from social and even psychological sciences to provide structured approaches for manipulating the incentives of network nodes effectively.

Acknowledgements
Not applicable.

Author contributions
All authors have contributed equally to the paper. All authors read and approved the final manuscript.

Funding
The authors did not receive support or any funding from any organization for the submitted work.

Availability of data and materials
The dataset employed in this study was sourced from the repository by Rohrer and Elias (2021). This dataset is accessible online via https://git.tu-berlin.de/rohrer/discharged-pc-data/-/tree/master/snapshots.

Declarations

Competing interests
The authors declare that they have no competing interests.

Received: 10 September 2023 Accepted: 22 October 2023
Published online: 30 October 2023

References

ACINQ (2016) Eclair. https://github.com/ACINQ/eclair. GitHub
Albert R, Barabási A-L (2002) Statistical mechanics of complex networks. Rev Mod Phys 74(1):47
Alstott J, Bullmore E, Plenz D (2014) powerlaw: a python package for analysis of heavy-tailed distributions. PLoS ONE 9(1):85777
Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512
Bollobás B, Riordan O (2004) Robustness and vulnerability of scale-free random graphs. Int. Math. 1(1):1–35
Bonacich P (1987) Power and centrality: a family of measures. Am J Sociol 92(5):1170–1182
Borgatti SP, Everett MG (2000) Models of core/periphery structures. Social Netw 21(4):375–395
Camilo GF, Rebello GAF, Souza LAC, Potop-Butucaru M, Amorim MD, Campista MEM, Costa LHM (2022) Topological evolution analysis of payment channels in the lightning network. In: 2022 IEEE Latin-American conference on communications (LATINCOM), IEEE, pp. 1–6
Clauset A, Shalizi CR, Newman ME (2009) Power-law distributions in empirical data. SIAM Rev 51(4):661–703
Conoscenti M, Vetrò A, De Martin JC, Spini F (2018) The cloth simulator for htlc payment networks with introductory lightning network performance results. Information 9(9):223
Cox DR (1958) The regression analysis of binary sequences. J Roy Stat Soc: Ser B (Methodol) 20(2):215–232
Crucitti P, Latora V, Marchiori M, Rapisarda A (2003) Efficiency of scale-free networks: error and attack tolerance. Physica A 320:622–642
Demers A, Greene D, Hauser C, Irish W, Larson J, Shenker S, Sturgis H, Swinehart D, Terry D (1987) Epidemic algorithms for replicated database maintenance. In: Proceedings of the Sixth Annual ACM Symposium on Principles of Distributed Computing, pp. 1–12

Dorfman R (1979) A formula for the gini coefficient. The review of economics and statistics, 146–149
Efron B, Tibshirani RJ (1994) An introduction to the bootstrap. CRC Press, Cambridge
Elements Project (2016) Core lightning. GitHub https://github.com/ElementsProject/lightning
Erdin E, Mercan S, Akkaya K (2021) An evaluation of cryptocurrency payment channel networks and their privacy implications. arXiv preprint arXiv:2102.02659
Freeman LC (1977) A set of measures of centrality based on betweenness. Sociometry 40:35–41
Gallos LK, Cohen R, Argyrakis P, Bunde A, Havlin S (2005) Stability and topology of scale-free networks under attack and defense strategies. Phys Rev Lett 94(18):188701
Google (2015) gRPC: A high performance, open source universal RPC framework. https://grpc.io/. gRPC
Gowda KC, Krishna G (1978) Agglomerative clustering using the concept of mutual nearest neighbourhood. Patt Recogn 10(2):105–112
Guo Y, Tong J, Feng C (2019) A measurement study of bitcoin lightning network. In: 2019 IEEE International conference on blockchain (Blockchain), IEEE, pp. 202–211
Harris J, Zohar A (2020) Flood & loot: A systemic attack on the lightning network. In: Proceedings of the 2nd ACM conference on advances in financial technologies, pp. 202–213
Hartigan JA, Wong MA (1979) Algorithm as 136: a k-means clustering algorithm. J Royal Stat Soci. Series C (Appl Stat) 28(1):100–108
Herrera-Joancomartí J, Navarro-Arribas G, Ranchal-Pedrosa A, Pérez-Solà C, Garcia-Alfaro J (2019) On the difficulty of hiding the balance of lightning network channels. In: Proceedings of the 2019 ACM asia conference on computer and communications security, pp. 602–612
Jolliffe IT (2002) Principal component analysis for special types of data. Springer, Berlin
Lange K, Rohrer E, Tschorsch F (2021) On the impact of attachment strategies for payment channel networks. In: 2021 IEEE International conference on blockchain and cryptocurrency (ICBC), IEEE, pp. 1–9
Lee S, Kim H (2020) On the robustness of lightning network in bitcoin. Pervas Mob Comput 61:101108
Lightning Labs (2019) The node operator's guide to the lightning galaxy, Part 2: node scoring and pathfinding. https://lightning.engineering/posts/2019-11-07-routing-guide-2/. Lightning Labs
LightningNetwork (2017) Lightning Network Deamon. https://github.com/lightningnetwork/lnd. GitHub
Lin J-H, Primicerio K, Squartini T, Decker C, Tessone CJ (2020) Lightning network: a second path towards centralisation of the bitcoin economy. New J Phys 22(8):083022
Lin J-H, Marchese E, Tessone CJ, Squartini T (2022) The weighted bitcoin lightning network. Chaos, Solitons & Fractals 164:112620
Malavolta G, Moreno-Sanchez P, Kate A, Maffei M, Ravi S (2017) Concurrency and privacy with payment-channel networks. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, pp. 455–471
Martinazzi S (2019) The evolution of lightning network's topology during its first year and the influence over its core values. arXiv preprint arXiv:1902.07307
Martinazzi S, Flori A (2020) The evolving topology of the lightning network: centralization, efficiency, robustness, synchronization, and anonymity. PLoS ONE 15(1):0225966
Massey FJ Jr (1951) The kolmogorov-smirnov test for goodness of fit. J Am Stat Assoc 46(253):68–78
Mizrahi A, Zohar A (2021) Congestion attacks in payment channel networks. In: International conference on financial cryptography and data security, Springer, pp. 170–188
Naumenko G, Riard A (2021) Time-dilation attacks on lightning network# 2. Cryptoeconomic Systems
Newman ME (2005) Power laws, pareto distributions and zipf's law. Contempor Phys 46(5):323–351
Pérez-Sola C, Ranchal-Pedrosa A, Herrera-Joancomartí J, Navarro-Arribas G, Garcia-Alfaro J (2020) Lockdown: Balance availability attack against lightning network channels. In: International conference on financial cryptography and data security, Springer, pp. 245–263
Pickhardt R (2018) is the Barabasi Albert Model a reasonable choice for the autopilot? https://github.com/lightningnetwork/lnd/issues/677. Github
Pickhardt R (2019) lightning-network-autopilot. https://github.com/renepickhardt/lightning-network-autopilot/. GitHub
Rezvani J (2021) Zion - The Social Network built on Bitcoin. https://www.zion.fyi/. Zion
Robinson D (2019) Htlcs considered harmful. In: Proceeding Stanford Blockchain Conference https://cbr.stanford.edu/sbc19/
Rohrer E (2021) discharged-pc-data. https://git.tu-berlin.de/rohrer/discharged-pc-data. Git
Rohrer E, Malliaris J, Tschorsch F (2019) Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks. In: 2019 IEEE European symposium on security and privacy workshops (EuroS &PW), IEEE, pp. 347–356
Rohrer E, Tschorsch F (2020) Counting down thunder: timing attacks on privacy in payment channel networks. In: Proceedings of the 2nd ACM conference on advances in financial technologies, pp. 214–227
Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65
Sen A, Sen MA, Foster JE, Amartya S, Foster JE, et al. (1997) On Economic Inequality. Oxford university press, Oxford
Seres IA, Benczúr AA (2021) A cryptoeconomic traffic analysis of bitcoin's lightning network. Cryptoeconomic Systems
Seres IA, Gulyás L, Nagy DA, Burcsi P (2020) Topological analysis of bitcoin's lightning network. In: Mathematical Research for Blockchain Economy, Springer, Berlin, pp. 1–12
Tochner S, Schmid S, Zohar A (2019) Hijacking routes in payment channel networks: A predictability tradeoff. arXiv preprint arXiv:1909.06890
Tochner S, Zohar A, Schmid S (2020) Route hijacking and dos in off-chain networks. In: Proceedings of the 2nd ACM conference on advances in financial technologies, pp. 228–240
Topirceanu A, Udrescu M, Marculescu R (2018) Weighted betweenness preferential attachment: a new mechanism explaining social network formation and evolution. Sci Rep 8(1):1–14

Tsiotas D (2020) Detecting differences in the topology of scale-free networks grown under time-dynamic topological fitness. Sci Rep 10(1):1–16
Wang Z, Zhang R, Sun Y, Ding H, Lv Q (2022) Can lightning network's autopilot function use ba model as the underlying network? Front Phys 9:794160
Zabka P, Foerster K-T, Decker C, Schmid S (2022) Short paper: A centrality analysis of the lightning network. In: International conference on financial cryptography and data security, Springer, pp. 374–385

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com