Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

Crawling the Bitcoin Client Ecosystem

Semester Thesis

Hyun-Min Chang
changh@ethz.ch

Distributed Computing Group
Computer Engineering and Networks Laboratory
ETH Zürich

Supervisors:
Dr. Lucianna Kiffer, Lioba Heimbach
Prof. Dr. Roger Wattenhofer

January 9, 2024

Acknowledgements

First and foremost, my deepest thanks go to my supervisors, Dr. Luciana Kiffer and Lioba Heimbach, for their exceptional guidance and support in the development of this thesis. Their patience, encouragement, and enthusiasm have been pivotal in motivating me, especially during technical challenges I faced with the machine running the crawler.

I am also grateful to Prof. Dr. Roger Wattenhofer and the Computer Engineering and Networks Laboratory for providing the opportunity to conduct this study.

Lastly, my heartfelt appreciation extends to my family and friends for their unwavering belief in me and constant encouragement, helping me remain focused on my goals.

Abstract

In this semester thesis, I explore the complex dynamics of Bitcoin and its forks, including Bitcoin-Cash, Dogecoin, Litecoin, and Bitcoin-SV. I developed a custom crawler using Go (Golang) to analyse these cryptocurrency networks, focusing on peer activities.

The research uncovers significant geographical and organisational dispersion of network peers, revealing concentrations in the US and Germany for countries as well as "Hetzner Online GmbH", "Amazon.com, Inc.", and "OVH SAS" for organisations. Comparative analysis with Blockchair data confirmed the effectiveness of my crawler.

The study highlights areas for enhancement in data collection, underscoring the potential for refining the crawler's capabilities. This aspect opens up opportunities for further advancements and optimisations in data gathering techniques. Additionally, it highlights the necessity of more finely tuned parameters for the crawler to enhance data gathering results. These findings lay the groundwork for further research and improvements in understanding the evolving nature of cryptocurrency networks.

Contents

Chapter 1

Introduction

1.1 Background and Context on Cryptocurrency Networks and P2P Architecture

Cryptocurrency networks, epitomised by Bitcoin, are realised using distributed systems that critically depend on the performance and security of their interconnecting network. These networks operate based on a peer-to-peer (P2P) model, which facilitates the distribution of system data while maintaining a decentralised structure. Unlike traditional communication networks, cryptocurrency networks often come with unique requirements, such as anonymity and different traffic mixes, including frequent broadcasts of transactions and blockchain states [1] [2].

P2P cryptocurrency networks support the communication between entities by allowing network members to synchronise their system state view and disseminate peer information. This decentralised approach lacks a standard format, leading to diverse implementations across various cryptocurrencies [1]. Furthermore, the incentive structures within these networks, essential for maintaining the distributed ledger, differ significantly from those in conventional P2P networks. For instance, in Bitcoin, miners are incentivised through block rewards and transaction fees, influencing their behaviour in transaction propagation [2].

1.2 Challenges in Crawling Cryptocurrency Networks

Designing crawlers for cryptocurrency networks presents unique challenges. The network topology of cryptocurrencies like Bitcoin, which employs a flat random graph topology, significantly differs from traditional networks.

Performance implications are also a key consideration. The distributed nature of these networks can introduce bottlenecks, impacting scalability. For example, in Bitcoin, the propagation speed of blocks affects the network's ability to function as a decentralised system. If blocks are produced faster than a node can receive them, it compromises the node's ability to verify transactions and blocks, potentially leading to exclusion from the network [2].

1.3 Research Objectives

The primary objective of this thesis is to develop and evaluate a custom crawler for the Bitcoin network, using the btcsuite's btcd/wire package [3]. This involves:

- Understanding the unique characteristics and challenges of the Bitcoin P2P network.
- Designing and implementing a crawler that can efficiently navigate and gather data from this network.
- Comparing the performance and effectiveness of the custom crawler with existing public crawlers.

1.4 Related Work

Significant research has been conducted on the unique aspects of cryptocurrency P2P networks. This includes detailed studies on network topology, neighbour discovery, block and transaction propagation, and security vulnerabilities in these decentralised systems [1] [2]. However, despite these advances, many fundamental aspects of cryptocurrency networks, especially concerning node distribution and activity patterns, remain underexplored.

Building on this body of work, this thesis extends the research I conducted in my bachelor's thesis, titled "Measuring Cryptocurrency Networks". The objective of that study was to acquire a comprehensive understanding of node activity across various cryptocurrencies [4]. By gathering and analysing data from multiple node explorers and API sources, it provided valuable insights into the distribution and activity trends within these networks, setting a foundation for the in-depth exploration undertaken in this current research.

Chapter 2

Methodology

2.1 Crawler Design

2.1.1 Choice of Programming Language: Go (Golang)

The crawler is implemented in Go (Golang), chosen for its exceptional concurrency and networking capabilities. Go's native support for handling multiple network connections in parallel, along with its efficient performance, is ideal for the demanding requirements of network crawling, particularly in the dynamic environment of cryptocurrency networks. Go is famous for its easy concurrency, making it a popular choice for cloud-native projects and network applications, which are heavily reliant on concurrency features like goroutines and channels [5] [6].

2.1.2 Architecture and Functionality

The crawler's architecture is modular, with each component designed for specific roles in the network crawling process:

1. **Network Data Acquisition**
   (`crawl_btcd.go`, `crawl_ltcd.go`): These modules are tasked with acquiring peer lists from known nodes in the network. Upon establishing a successful connection with a node, the crawler uses the 'getaddr' message to retrieve the node's peer list. It then connects to these new nodes and repeats the process, thereby expanding the network discovery. This iterative process continues until no new nodes are discovered, ensuring comprehensive coverage of the network's topology.

2. **Peer Connectivity and Status Check**
   (`checkPeers_btcd.go`, `checkPeers_ltcd.go`): These modules are primarily used post-crawl to verify if the nodes discovered during the crawl are still active and responsive.

3. **Network Responsiveness Assessment**
(`pingPeers.go`): This module evaluates the responsiveness of network nodes using TCP ping, providing insights into network availability.

4. **Support Functions and Utilities**
(`universal_functions.go`): Comprising utility functions that support the crawler's operation, this module aids in tasks like IP address parsing and file handling.

5. **User-Configurable Settings**
(`crawler_main.go`): The crawler is designed to be flexible, allowing runtime customisation of parameters such as network selection (`currency`), IP version (`ipversion`), timeout (`timeout`), retry strategy (`retry`), and level of concurrent operations (`concurrency`). It also supports various operation modes (`mode`) and adjustable verbosity for debugging (`verbosity`).

2.1.3 Operation Modes

The crawler supports various modes, each tailored for different crawling goals:

- **Crawl Mode**
(`crawl_btcd.go`, `crawl_ltcd.go`): Active connection mode for data gathering from network nodes.

- **Check Mode**
(`checkPeers_btcd.go`, `checkPeers_ltcd.go`): Verifies the reachability of peers in the network.

- **Ping Mode**
(`pingPeers.go`): Uses TCP pinging to evaluate node response times and latency.

2.2 Data Collection

Commencing on November 24, 2023, the data collection process was meticulously automated using a crontab setup on a Linux machine at ETH Zurich. Initially, the crawler was programmed to activate in crawl mode for all currencies daily at 5 am, supplemented by hourly executions in check and ping modes. However, a series of events impacted data quality. On December 12, a critical patch noticeably reduced reachable nodes, potentially due to a security update affecting connections, as detailed in the Results chapter (Chapter 5). Additionally, no data was collected on December 21 and 22 due to unforeseen issues, and the subsequent Ubuntu update on the host machine on December 22 caused the crontab to malfunction. Furthermore, due to a mistake, no pinging data was collected from December 4, 2023, onwards. Consequently, pinging data post-December 4 and crawling data post-December 22 was deemed unreliable and excluded from the analysis.

Chapter 3

Implementation

3.1 Crawler Scripts

The initial phase in the crawler‘s development involved identifying a suitable GitHub repository. The ‘btcdsuite‘, specifically the ‘btcd/wire‘ package, was selected. ‘btcd‘ represents an alternative full node Bitcoin implementation written in Go [7]. The ‘wire‘ package implements the Bitcoin wire protocol and offers a comprehensive test suite with 100% coverage, ensuring functionality [3]. Designed for standalone use, it is ideal for projects interfacing at the Bitcoin protocol level.

The subsequent step was to establish a connection with a Bitcoin peer, achieved by utilising the ‘wire.WriteMessage()‘ and ‘wire.ReadMessage()‘ functions from the ‘wire‘ package. These functions were key in sending a ‘version‘ message and receiving a ‘verack‘ message, initiating the handshake mechanism crucial in the Bitcoin protocol for peer communication [8].

Upon successful initial connection, the next task involved sending a ‘getaddr‘ message after the version handshake, followed by receiving an ‘addr‘ message, which contains the peer list. The process encountered intermittent success, leading to the implementation of a retry loop. This enhancement significantly improved the odds of successfully acquiring an ‘addr‘ message and was also applied to the handshake process to increase stability.

Once the basic functions of the crawler were operational, the next crucial phase involved creating a process to recursively query peers for their peer lists using ‘getaddr‘, and then to query the peers from these lists for their peers. This iterative process continued until no new peers were discovered, ensuring a thorough network traversal.

The crawler, capable of functioning with both IPv4 and IPv6, was adapted to work with a range of cryptocurrencies including Bitcoin, Bitcoin Cash, Bitcoin SV, Litecoin, and Dogecoin. For Litecoin, a specific fork of ‘btcd/wire‘ known as ‘ltcd/wire‘ was employed [9]. Attempts to extend the crawler‘s functionality to Zcash and Dash were made; however, the Dash fork ‘dashd-go‘ was found to be in a non-functional state, and no feasible solutions were identified for Zcash [10].

For the seed node, initial attempts utilised a publicly available seeder node, which proved effective for Bitcoin. However, for other currencies like Dogecoin, Litecoin, and Bitcoin Cash, such publicly available seeders were not found. Consequently, the Blockchair API was used to obtain a random selection of 100 nodes for each of these currencies, serving as the seed nodes [11]. To maintain consistency, the same method was applied to Bitcoin. For Bitcoin SV, the WhatsOnChain service was employed to source the initial node list [12].

The data collected from all nodes were meticulously organised and saved. All gathered nodes were stored in a 'peerLists' folder in JSON format, with files named according to the convention:

'<currency>_IPv<ipversion> YYYY.MM.DD.json'.

Additionally, separate files were created in an 'active_peerLists' folder, which exclusively contained information about the peers that responded. This dual-folder structure enabled efficient categorisation and retrieval of data, distinguishing between all discovered nodes and those actively participating in the network.

3.2 Pinging and Protocol Verification Scripts

The 'pingPeers.go' script performs TCP pings on nodes listed in the 'active_peerLists' folder, selecting files corresponding to the current date for each cryptocurrency. It then records the responsiveness of each node as JSON files, and saves them in the 'pinged_peerLists' folder.

Concurrently, the 'checkPeers_ltcd.go' and 'checkPeers_btcd.go' scripts engage in protocol-level verification. They attempt a version handshake with peers from the 'active_peerLists' and document whether each peer responds. The outcomes of these handshakes are saved as JSON files in the 'responded_peerLists' folder, offering a comprehensive view of peer activity and network reliability.

3.3 Enhancements and Feature Additions

As the crawler's core functionality became robust, focus shifted to enhancing efficiency and usability. A significant improvement was the introduction of concurrency, dramatically reducing the runtime and enabling more efficient data processing.

Further, the crawler's flexibility was enhanced by implementing command-line parameters, allowing for dynamic configuration. Instead of multiple scripts for each currency, functionality was consolidated into a single, versatile script. Parameters such as '-currency', '-ipversion', '-concurrency', '-retry', '-timeout', and '-verbosity' were added, providing control over operational aspects like the target cryptocurrency, IP version, concurrency level, connection attempts, response timeout, and debugging verbosity.

Additionally, the capability to specify the crawler‘s mode was introduced. This allowed for different operations like ‘-mode=crawl‘ for crawling, ‘-mode=ping‘ for pinging, and ‘-mode=check‘ for protocol-level peer response checking, further enhancing the crawler‘s utility in diverse scenarios.

Chapter 4

Data Processing

4.1 Data Structure and Preliminary Cleaning

The data was systematically organised with key-value pairs in the 'active_peerLists' folder, where each peer's IP address was marked as 'True' for responsive nodes. Similar formatting was applied for the pinging and in-protocol checking results, with 'False' indicating non-responsive peers. Minimal cleaning was required, mainly involving the removal of extraneous characters like unnecessary braces.

4.2 Data Visualisation Using Jupyter Notebooks

The Jupyter Notebook environment was employed for its dynamic data analysis and immediate visualisation capabilities.

4.2.1 Geolocation and ASN Data Retrieval

To enrich the IP address data with geographical and organisational information, the 'ipinfo.io' API was utilised [13]. The 'geo_distribution' function was used to for IP address details in a locally stored file, 'ip_list.json'. If the information was not found locally, the function then made an API request to 'ipinfo.io'. This process ensured efficient data retrieval by minimising API calls. The gathered country, ASN, and company information was then used for detailed geographical and organisational analysis.

4.2.2 Data Preparation and Cleaning

Functions like 'check_missing_entries' and 'fill_missing_entries' were used to ensure data completeness. Additional cleanup involved removing extraneous characters from the dataset.

4.2.3 Geographical Distribution Analysis

The ‘geo_distribution‘ function, along with GeoPandas and pandas, facilitated the plotting of peers’ geographical distribution. ‘plot_geo_distribution‘ and similar functions transformed raw data into maps, pie charts, and other visualisations, highlighting peer activity by country, Autonomous System Number (ASN), and company.

4.2.4 Temporal Analysis

Temporal changes in network activity were examined using ‘plot_from_total_dict‘, which plotted active peer counts over time, providing insights into network dynamics.

4.2.5 Combinatorial Analysis

The script used ‘plot_ip_combinations‘ to analyse IP address overlap across different cryptocurrencies, revealing patterns in multi-network peer usage.

4.2.6 Pie Chart Visualisations

For an illustrative breakdown of peer activity by geography, ASN, and company, the script utilises pie chart visualisations. Functions like ‘plot_geo_distribution_pie‘, ‘plot_geo_distribution_asn_pie‘, and ‘plot_geo_distribution_company_pie‘ create pie charts to depict the distribution of peer activities. These functions first calculate the average peer activity within specified time frames and then represent these averages visually. The custom function ‘make_autopct‘ enhances the pie charts by displaying both the percentage and the actual value of each segment, offering a clear and informative visual representation of the data. Important to note is that only data from 2023.11.24 until 2023.12.11 was used, as including the data from later dates might skew the results, due to the patch mentioned in Section 2.2.

Chapter 5

Results

5.1 Peer Activity over Time

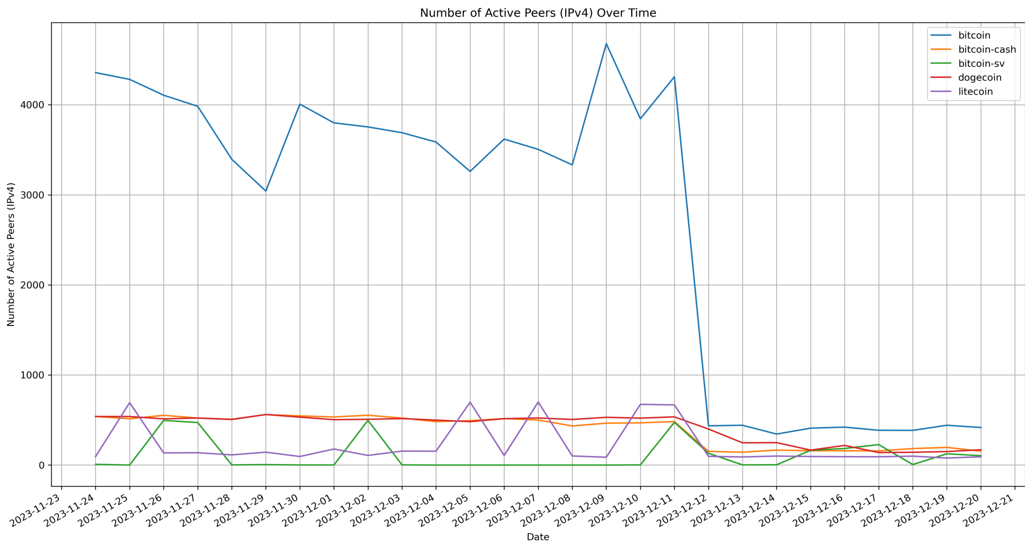
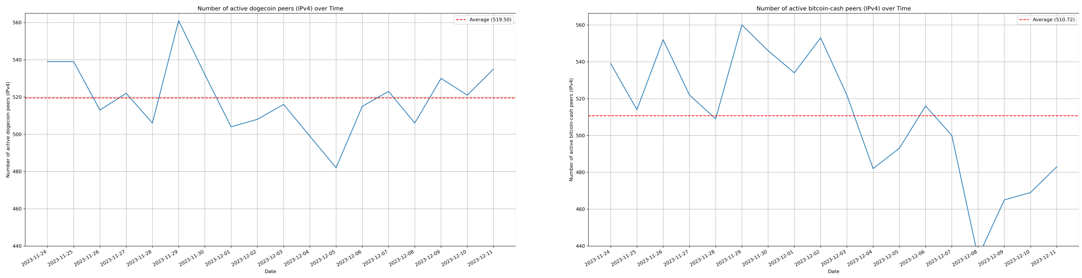


Figure 5.1: The line graph displays the number of active peers over the study period.

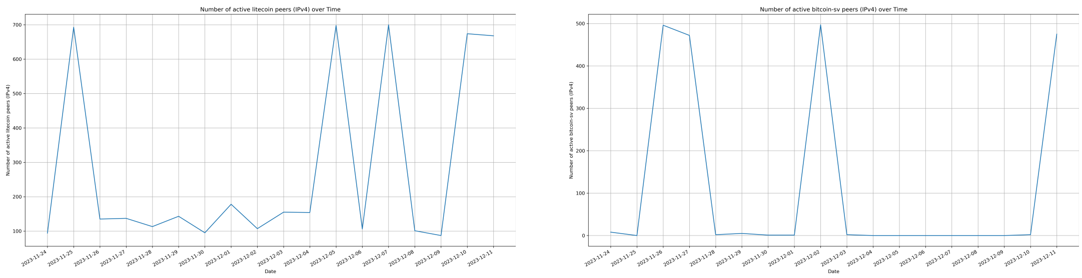
As depicted in Figure 5.1, Bitcoin has by far the highest amount of active peers. Dogecoin and Bitcoin-Cash seem to have relatively consistent and similar levels of activity, whereas Litecoin and Bitcoin-SV both seem to have spikes in activity. The sharp decline in the number of peers between the 11th and 12th of December is due to unforeseen problems with the machine previously mentioned patch in Section 2.2. As a result from now on the plots will only show data from the 24th of November till the 11th of December.



(a) Activity of Dogecoin (b) Activity of Bitcoin-Cash

Figure 5.2: Comparative peer activity between Dogecoin and Bitcoin-Cash.

The red dashed line in Figures 5.2a and 5.2b represents the average number of active peers up until 2023.12.11. It is apparent from these visualizations that Dogecoin has a slightly higher amount of active peers compared to Bitcoin-Cash.



(a) Activity of Litecoin (b) Activity of Bitcoin-SV

Figure 5.3: Comparative peer activity between Litecoin and Bitcoin-SV.

Figure 5.3a shows Litecoin's peer activity with periodic spikes, whereas Bitcoin-SV, as seen in Figure 5.3b, displays erratic spikes interspersed with periods of no or almost no recorded activity. This inconsistency suggests possible connection challenges, which might not be isolated to Bitcoin-SV but could affect Litecoin as well. It implies that the crawler's parameters, optimised for Bitcoin's network characteristics, may require adjustment. A more tailored approach with varied 'retry' and 'timeout' settings might improve the effectiveness of the crawler across different cryptocurrency networks.

5.2 Pie Chart Distributions of Peers

The pie charts in this section provide an overview of peer distribution across different categories like countries, ASNs, and organisations. Each chart displays the top 10 entities in its category, along with an additional 'Other' segment. The 'Other' segment includes all remaining entities not in the top 10. For clarity, the absolute number of different countries, ASNs, or organisations included in 'Other' is enclosed in parentheses. This approach offers a clearer understanding of the scale and diversity of entities involved in each category.

5.2.1 Geographic Distribution of Cryptocurrency Peers

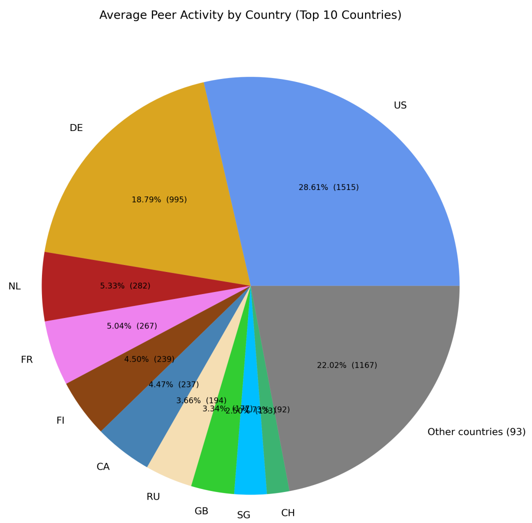


Figure 5.4: Peer activity distributed across the top 10 countries (all currencies combined)

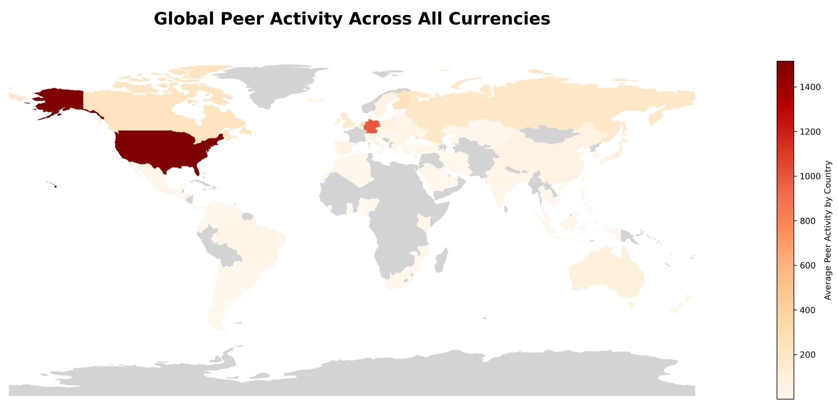
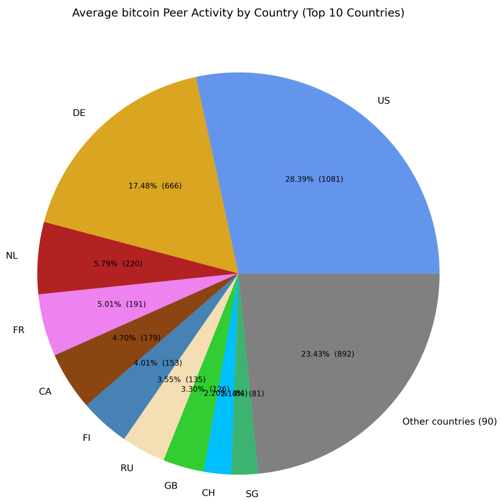
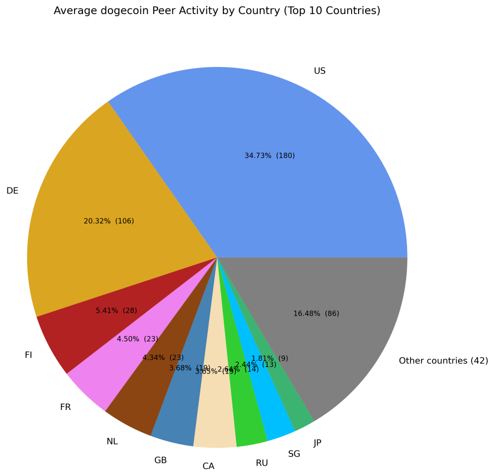


Figure 5.5: Heatmap of peer activity by country

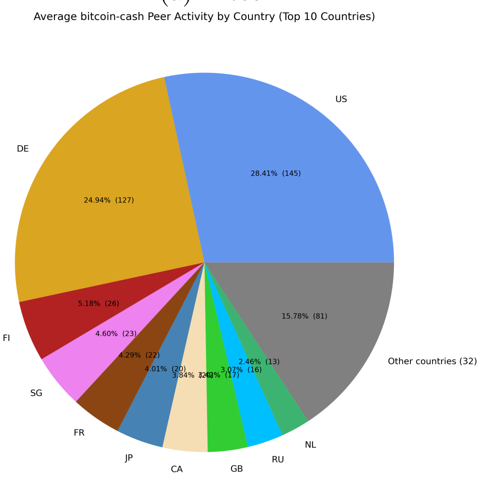
As it is apparent from Figures 5.4 and 5.6 that most of the activity comes from the US (dark blue) and Germany (orange) for all cryptocurrencies. This can also be observed by looking at Figure 5.5.



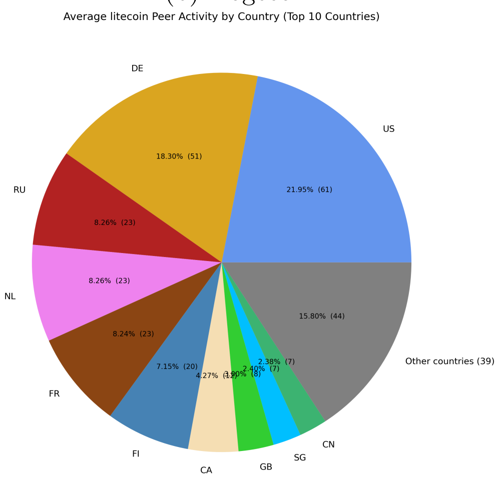
(a) Bitcoin



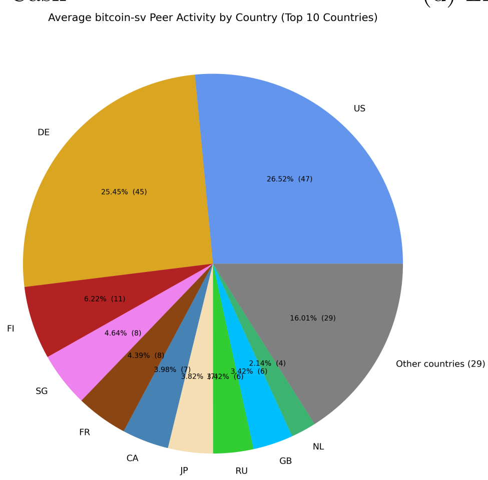
(b) Dogecoin



(c) Bitcoin-Cash



(d) Litecoin



(e) Bitcoin-SV

Figure 5.6: Peer Activity of various cryptocurrencies distributed across their top 10 countries

5.2.2 Distribution of Cryptocurrency Peers by ASN

The Autonomous System Number (ASN) is a unique identifier for a group of IP networks and routers that adheres to a single and clearly defined routing policy.

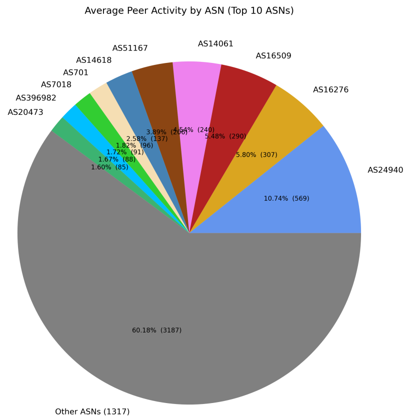


Figure 5.7: Total peer activity distributed across the top 10 ASNs.

Figure 5.7 reveals a more diverse distribution of peers across Autonomous System Numbers (ASNs) compared to the concentration observed across countries in Section 5.2.1. This diversity is likely attributed to the significantly higher number of ASNs (1327) relative to countries (103), contributing to a larger 'Other ASNs' category in the data.

5.2.3 Distribution of Cryptocurrency Peers by Organisation

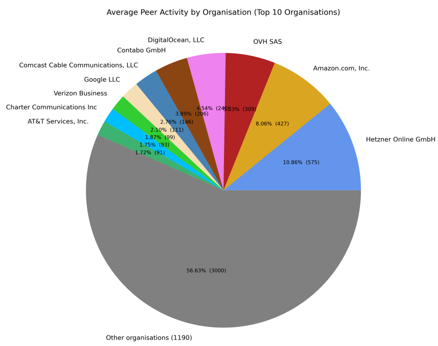
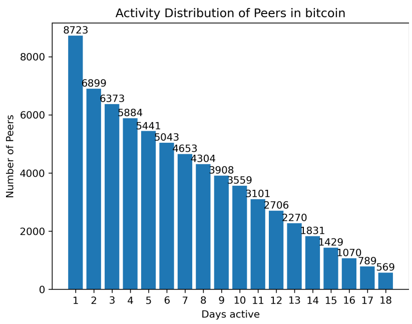


Figure 5.8: Total peer activity distributed across the top 10 organisations.

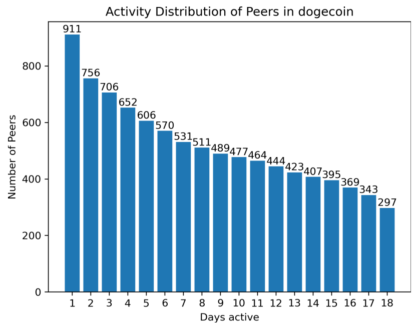
Similar reasoning to Section 5.2.2 can be used to explain Figure 5.8.

5.3 Activity by Number of Days

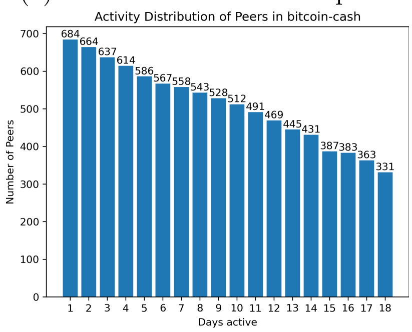
The figures in this section illustrate the distribution of peer activity in terms of the number of days they were active within the study period.



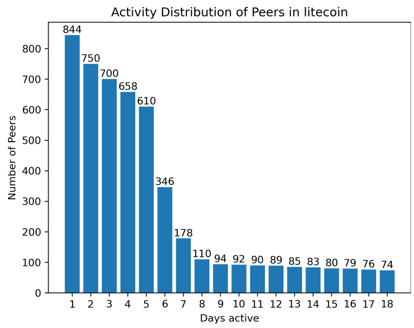
(a) Distribution of Bitcoin peers



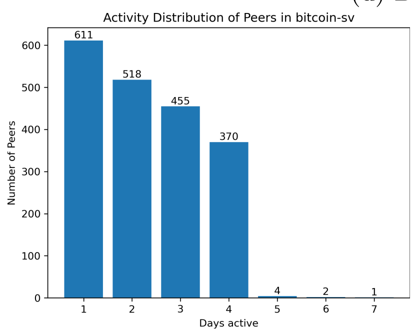
(b) Distribution of Dogecoin peers



(c) Distribution of Bitcoin-Cash peers



(d) Distribution of Litecoin peers



(e) Distribution of Bitcoin-SV peers

Figure 5.9: Activity distribution of peers in different cryptocurrencies, categorised by the number of days they were active.

Figure 5.9 presents data on peer activity within the cryptocurrency network, such as Bitcoin, over a specific study period. The plot shows the number of peers active for a minimum number of days. For example, in the case of Bitcoin, 8,723 peers were active for at least one day, 6,899 were active for at least two days, and so on. It's important to note how the peers are counted in this analysis: a peer that is active for multiple days is included in the count for each of those days. Thus, a peer active for three days is counted once in the ‘active for at least one day‘ category, once in the ‘active for at least two days‘ category, and once in the ‘active for at least three days‘ category.

Bitcoin (Figure 5.9a) and Dogecoin (Figure 5.9b) show a notable drop after 1 days, followed by an almost linear decrease in active peers with increasing time, while for Bitcoin-Cash the number of active peers seems to decrease linearly from the start. Notably, there is a sharp decline in active Litecoin peers after 6-7 days, as illustrated in Figure 5.9d, but as mentioned before Litecoin and Bitcoin-SV data is very likely unreliable, as the "spiky" nature of the ‘Peer activity over time‘ plots (Figures 5.3a and 5.3b) is very likely caused by errors during crawls.

5.4 Cross-Network Peer Participation

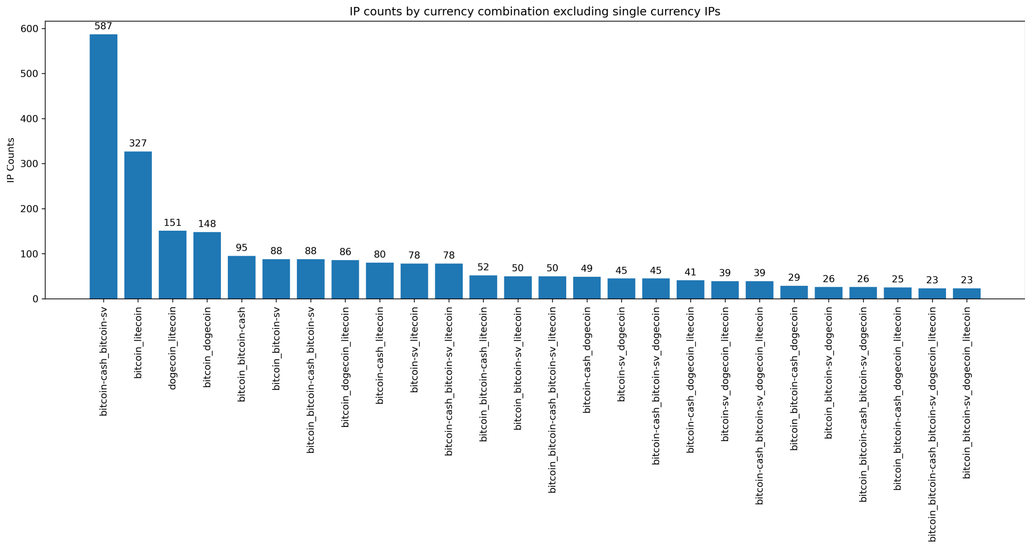


Figure 5.10: IP address overlap across cryptocurrency networks

Figure 5.10 exhibits the count of IP addresses active in multiple cryptocurrency networks. The most significant overlap occurs between Bitcoin SV and Bitcoin, indicating a shared user base. Conversely, unique pairs like Bitcoin and Dogecoin show lesser degrees of overlap, possibly pointing to more distinct communities.

5.5 Temporal Stability of Cryptocurrency Networks

The following figures present an analysis of the temporal stability of active peers within cryptocurrency networks. Each line represents a group of peers that were active at the corresponding labelled date. The persistence of these groups is tracked over successive days to reveal patterns in network participation stability.

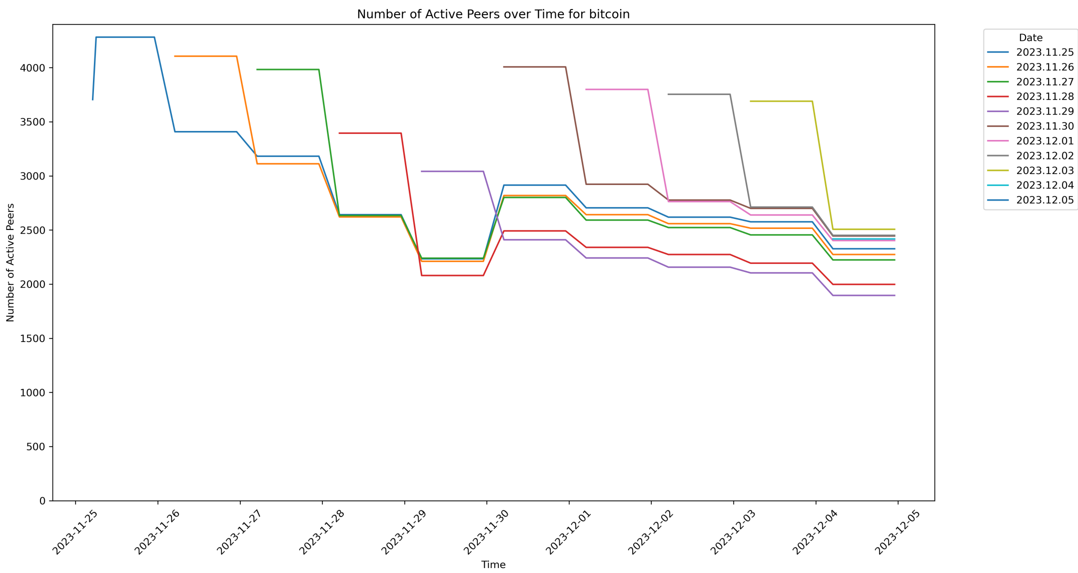


Figure 5.11: The graph tracks the number of originally active Bitcoin peers that remain responsive over time.

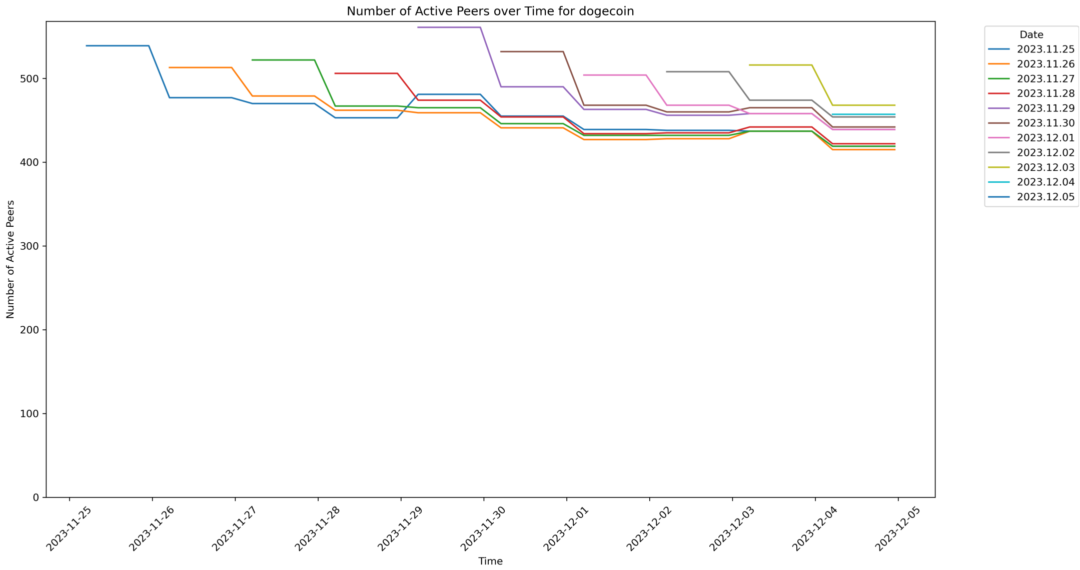


Figure 5.12: The graph tracks the number of originally active Dogecoin peers that remain responsive over time.

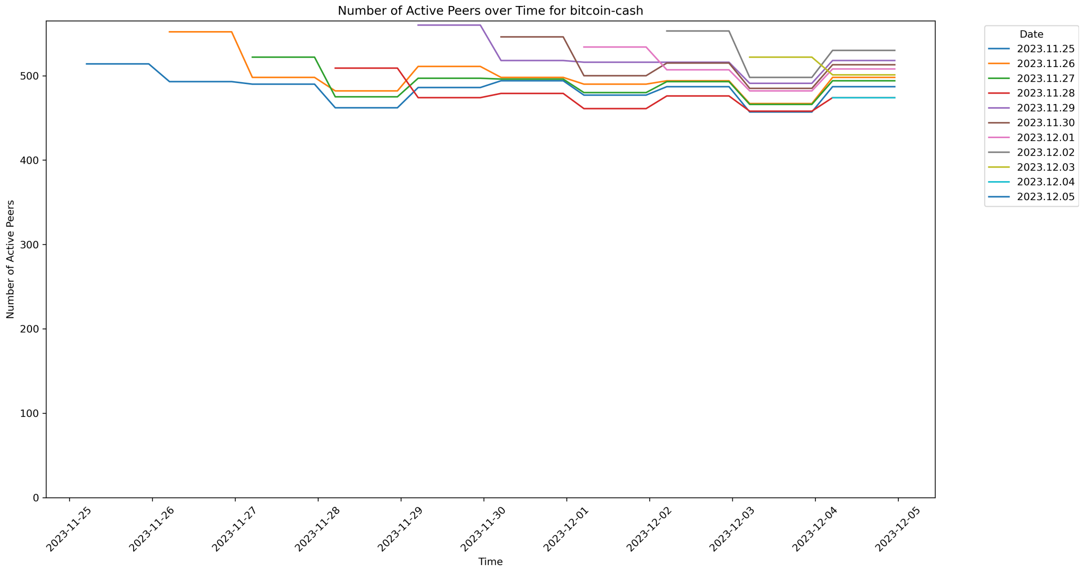


Figure 5.13: The graph tracks the number of originally active Bitcoin-Cash peers that remain responsive over time.

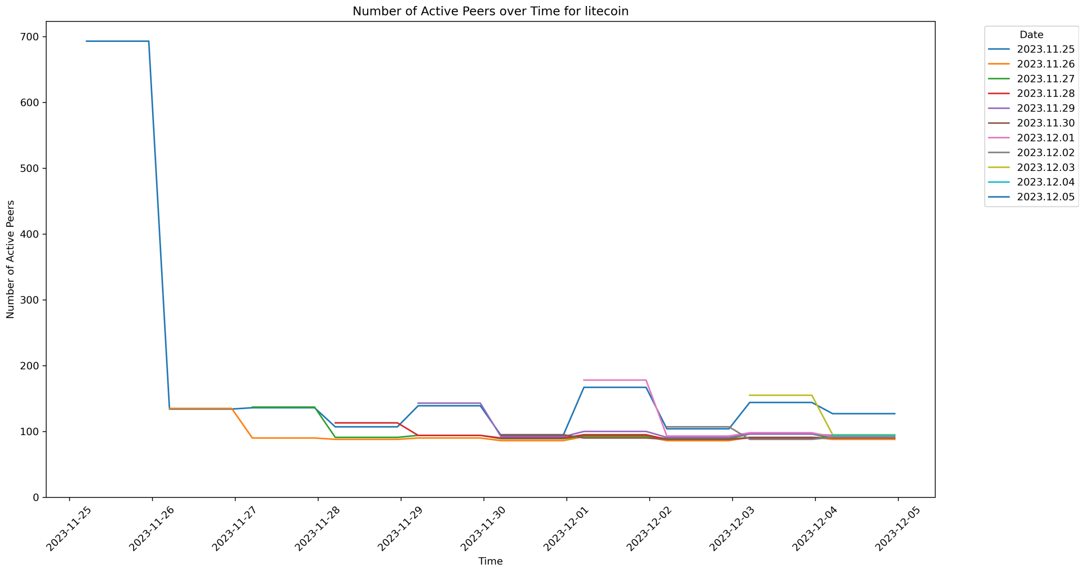


Figure 5.14: The graph tracks the number of originally active Litecoin peers that remain responsive over time.

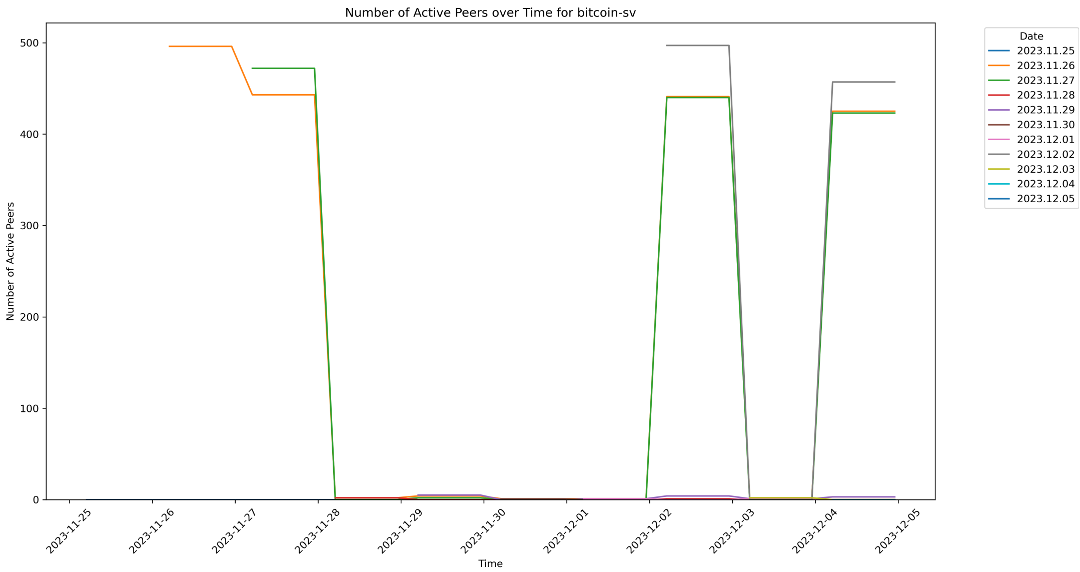


Figure 5.15: The graph tracks the number of originally active Bitcoin-SV peers that remain responsive over time.

5.6 Comparative Study

In order to ascertain the reliability of the data collected, a comparative analysis was conducted against data procured using the Blockchair API [11]. The data retrieval scripts for Blockchair were executed on an AWS server, which was not affected by the issues encountered with the university's machine.

However, as Blockchair does not gather information about Bitcoin-SV, no comparative study for that cryptocurrency could be made. This comprehensive comparison strengthens the credibility of the self-collected data, especially for trend analysis.

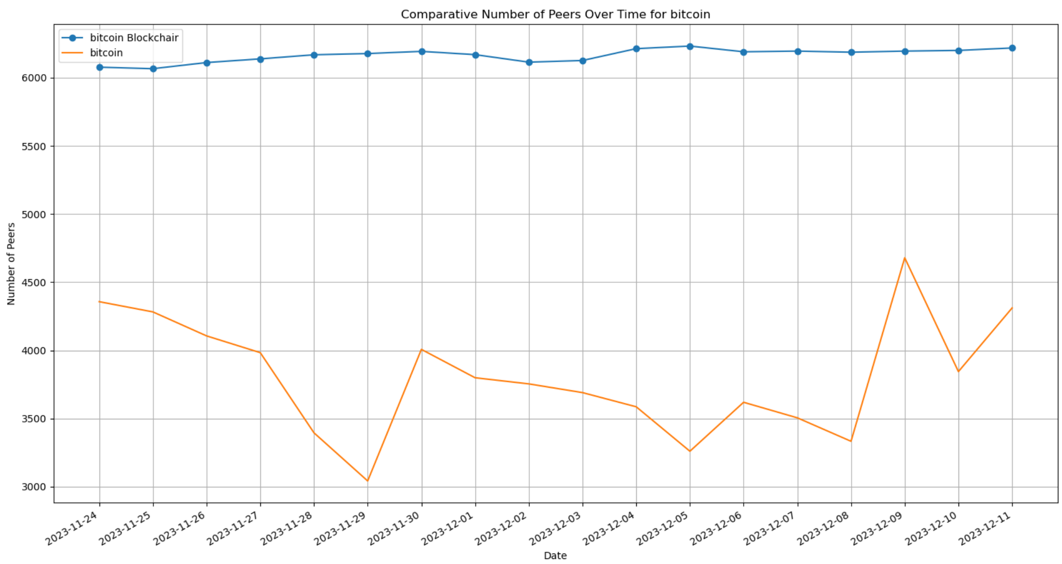


Figure 5.16: Comparison of peer counts over time for Bitcoin and Bitcoin using Blockchair data.

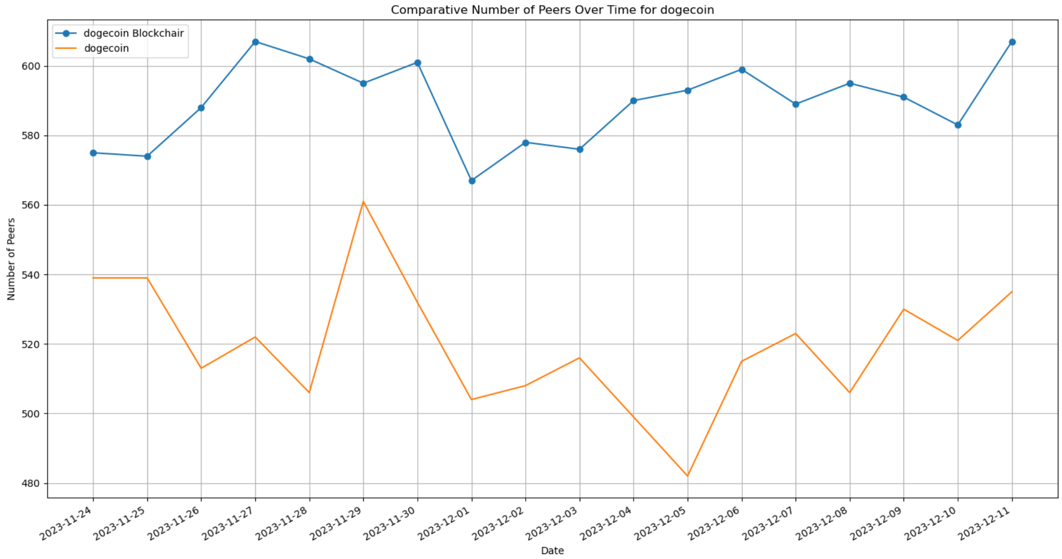


Figure 5.17: Comparison of peer counts over time for Dogecoin and Dogecoin using Blockchair data.

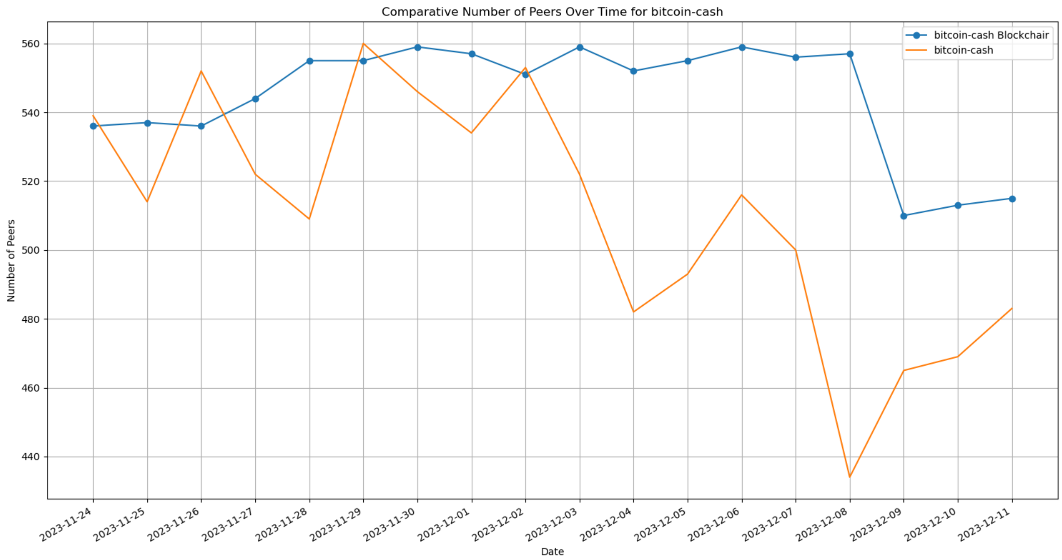


Figure 5.18: Comparison of peer counts over time for Bitcoin-Cash and Bitcoin-Cash using Blockchair data.

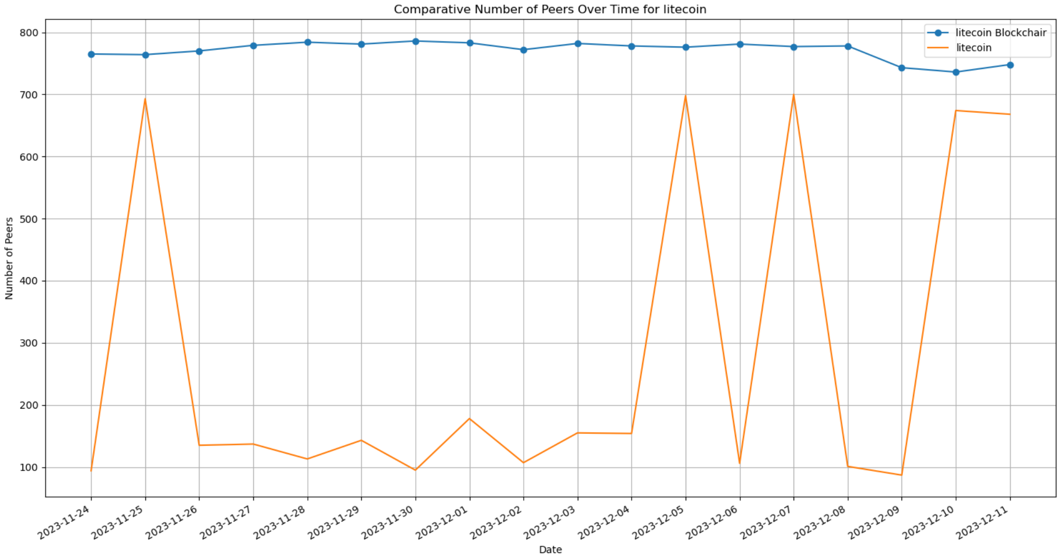


Figure 5.19: Comparison of peer counts over time for Litecoin and Litecoin using Blockchair data.

Chapter 6

Discussion and Conclusion

6.1 Interpretation of Results

6.1.1 Geographic and Organisational Insights

Analysis of Figures 5.4 reveals a significant concentration of cryptocurrency activity in the US (28.61%) and Germany (18.79%).

Unlike the geographic distribution where two countries account for 47.4% of the top 10 activity, the activity is more evenly distributed across ASN or Organization. The largest share is AS24940's 10.74%, followed by AS16276's 5.8% and AS16509's 5.48%. So in this case the top 3 ASN make up about 21% of the peers.

Similarly, when grouped by Organization, "Hetzner Online GmbH" leads with 10.86% followed by "Amazon.com, Inc." at 8.06% and "OVH SAS" at 5.83%.

Important to note is that a ASN can have multiple IP-ranges that correspond to different countries. And an organisation can be responsible for multiple ASNs.

For example both AS24940 and AS213230 (not in the top 10) are managed by "Hetzner Online GmbH". And while AS24940 mostly has IP-ranges local to Germany, AS213230 mostly has IP-ranges based in the US.

In addition to these findings, the 'Other' categories for ASNs and organisations account for more than half of the peers, with 60.18% and 56.63%, respectively. However, when we calculate the average peer count per entity, it amounts to only 0.046% for ASNs and 0.048% for organisations. This indicates that, on average, each entity outside the top 10 holds a relatively small share of peers, underscoring the significance of the top entities in these categories.

6.1.2 Activity Duration and Network Stability

The analysis in Figure 5.9 indicates a general trend, where peer activity almost linearly diminishes with increased duration. Notably, Litecoin's active peers, as shown in Figure 5.9d, experience a significant drop, suggesting a deviation from this trend, possibly due to connection issues. Bitcoin SV's data, represented in Figure 5.9e, also displays an inconsistent pattern, not entirely reflective of the network's standard behaviour, warranting further exploration. Additionally, Figures 5.11, 5.12, and 5.13 hint at a consistent base level of network activity. However, a more extended data collection period is necessary for a comprehensive and meaningful analysis of this aspect.

6.1.3 Comparative Study with Blockchair Data

The choice to use Blockchair for the comparative study, as discussed in Section 5.6, was influenced by familiarity with the platform from previous work "Measuring Cryptocurrency Networks" [4]. Blockchair was selected over alternatives like bitnodes.io [14] because it provides comprehensive data on four cryptocurrencies of interest. This consistency in sourcing data from a single provider reduces methodological variability and enhances the reliability of the comparison.

The comparative study, shows minor variations in peer counts for Bitcoin, Bitcoin-Cash, and Dogecoin when compared to Blockchair data (Figures 5.16, 5.18, 5.17). These variations suggest that while the crawler data aligns well for general trend analysis, it may not be as precise for exact counts at its current state. The discrepancies in Litecoin's data, observed in Figure 5.19, point towards potential data collection issues, which should be further investigated. For Bitcoin SV, a similar inference of data anomalies is drawn due to its resemblance to Litecoin's activity pattern.

6.2 Limitations of the Study and Future Work

The data collection process, as discussed in Section 2.2, encountered several challenges. Future improvements could include leveraging cloud services like AWS or Microsoft Azure, potentially addressing issues like the absence of IPv6 data collection. Exploring data collection from TOR nodes, while beyond the current scope, could be an interesting avenue. Moreover, the crawler's performance, particularly regarding the irregular activity patterns observed for Litecoin and Bitcoin-SV (Figures 5.3a and 5.3b), warrants a review and update. Lastly, the parameters used for crawling were selected under time constraints and should be further optimised and tested for each specific cryptocurrency to enhance data accuracy.

Additionally, long-term data collection is essential for capturing the dynamic nature of cryptocurrency networks accurately. Extended observation periods can reveal trends and patterns that short-term data may overlook, especially in understanding network resilience and peer lifecycle. Future studies should aim for prolonged data collection to yield more comprehensive and reliable insights into the behaviour and stability of cryptocurrency networks over time.

Bibliography

[1] S. Delgado-Segura, C. Pérez-Solà, J. Herrera-Joancomartí, G. Navarro-Arribas, and J. Borrell, "Cryptocurrency networks: A new p2p paradigm," *Mobile Information Systems*, vol. 2018, p. Article ID 2159082, 2018. [Online]. Available: https://www.hindawi.com/journals/misy/2018/2159082/

[2] M. Dotan, Y. Pignolet, S. Schmid, S. Tochner, and A. Zohar, "Survey on cryptocurrency networking: Context, state-of-the-art, challenges," *CoRR*, vol. abs/2008.08412, 2020. [Online]. Available: https://arxiv.org/abs/2008.08412

[3] Btcsuite, "btcd/wire: Package implementing the bitcoin wire protocol," https://github.com/btcsuite/btcd/tree/master/wire, 2023.

[4] H.-M. Chang, "Measuring cryptocurrency networks," https://pub.tik.ee.ethz.ch/students/2022-HS/BA-2022-46.pdf, 2023.

[5] S. Yegulalp, "Whats the go programming language (golang) really good for?" *InfoWorld*, 2023. [Online]. Available: https://www.infoworld.com/article/3198928/whats-the-go-language-really-good-for.html

[6] Go Dev, "The go programming language," https://go.dev/, 2023.

[7] Btcsuite, "btcd: An alternative full node bitcoin implementation written in go (golang)," https://github.com/btcsuite/btcd/, 2023.

[8] bitcoin.it, "Version handshake," https://en.bitcoin.it/wiki/Version_Handshake, 2023.

[9] Ltcsuite, "ltcd/wire: A fork of btcd/wire for litecoin," https://github.com/ltcsuite/ltcd/tree/master/wire, 2023.

[10] Dashpay, "dashd-go: Fork of btcd for dash," https://github.com/dashpay/dashd-go, 2023.

[11] blockchair.com, "Blockchair api - fetch data from 17 blockchains," https://blockchair.com/api, 2023.

[12] WhatsOnChain, "Whatsonchain api - bsv explorer - mainnet," https://whatsonchain.com/, 2023.

[13] ipinfo.io, "Ipinfo.io - ip address geolocation api," https://ipinfo.io, 2023.

[14] bitnodes.io, "Bitnodes - global bitcoin nodes distribution," https://bitnodes.io/, 2024.

Appendix A

Source Files

A.1 GitLab Repository

The complete source code, along with all the data used in this study, is accessible in a dedicated GitLab repository. This repository includes the crawler scripts, data analysis tools, and the datasets obtained during the research. Detailed documentation on the usage and functionality of these resources is provided in the repository's README file.

Access the repository at: https://gitlab.ethz.ch/changh/crawling-the-bitcoin-ecosystem

Appendix B

All Plots

B.1 Peer Activity

B.1.1 Plots over time

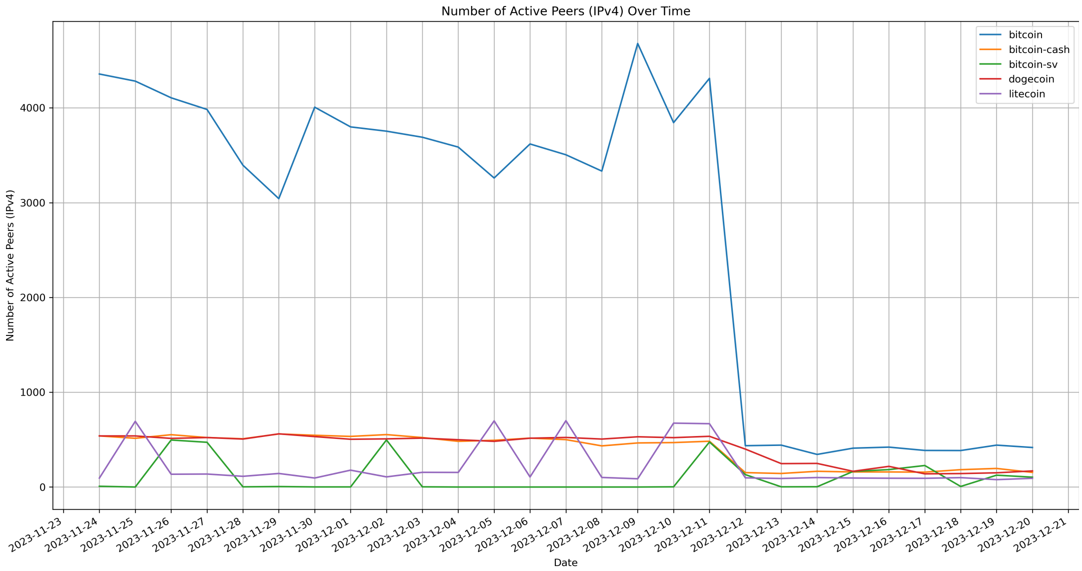


Figure B.1: Peer Activity Over Time

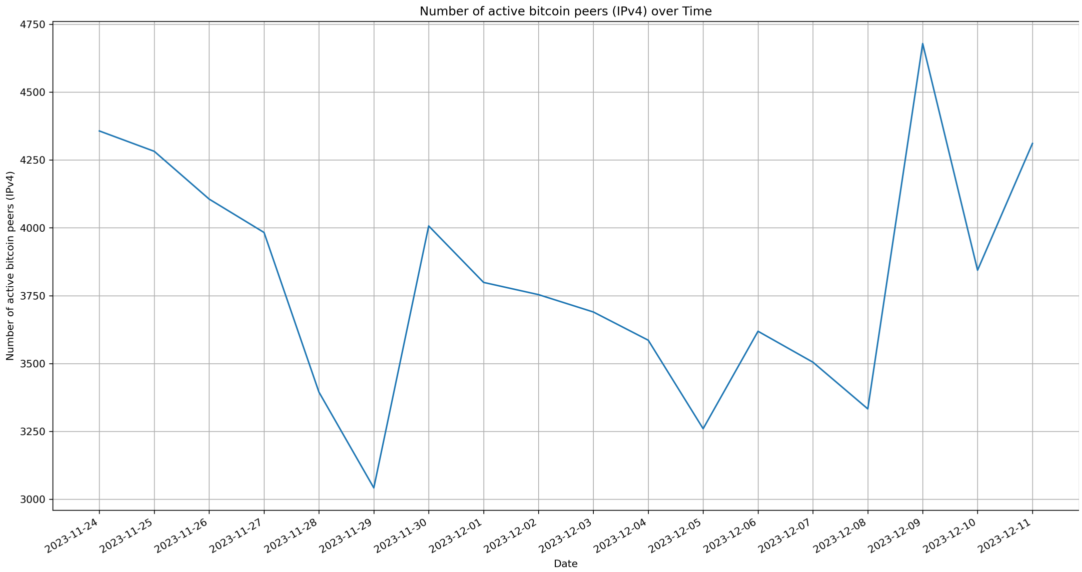


Figure B.2: Bitcoin Peer Activity Over Time

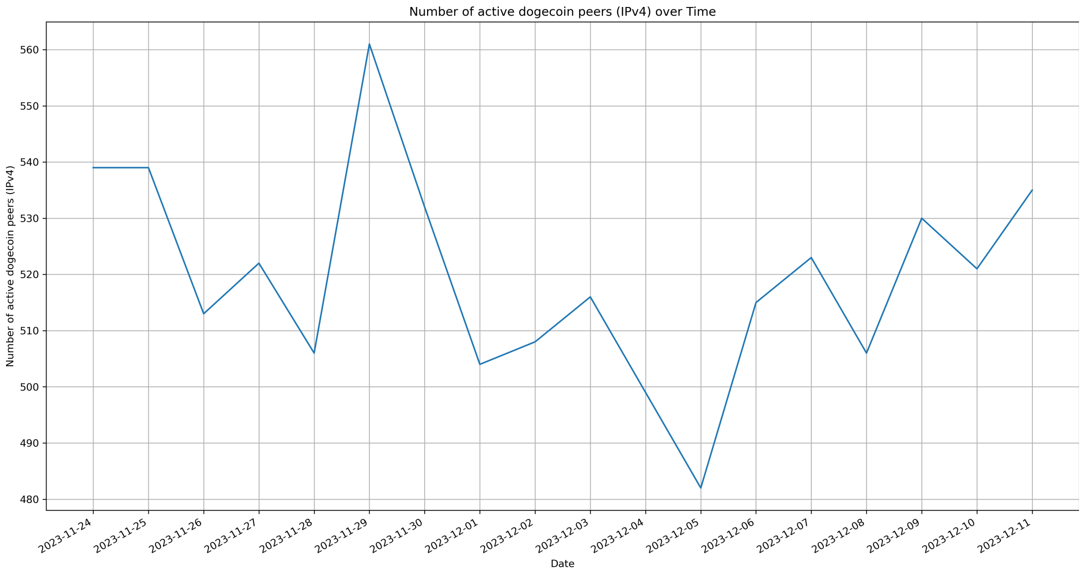


Figure B.3: Dogecoin Peer Activity Over Time

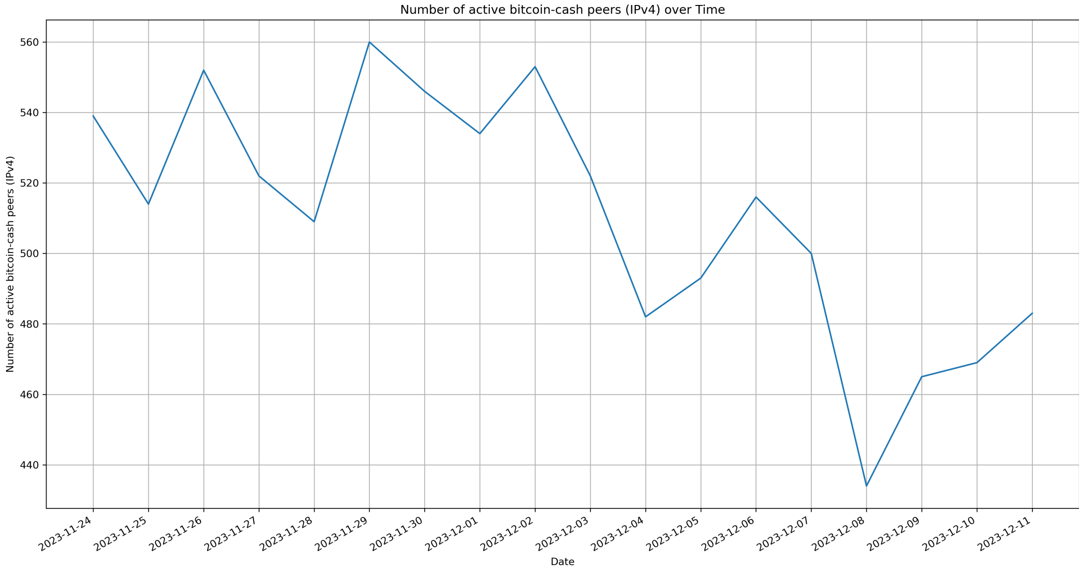


Figure B.4: Bitcoin-Cash Peer Activity Over Time

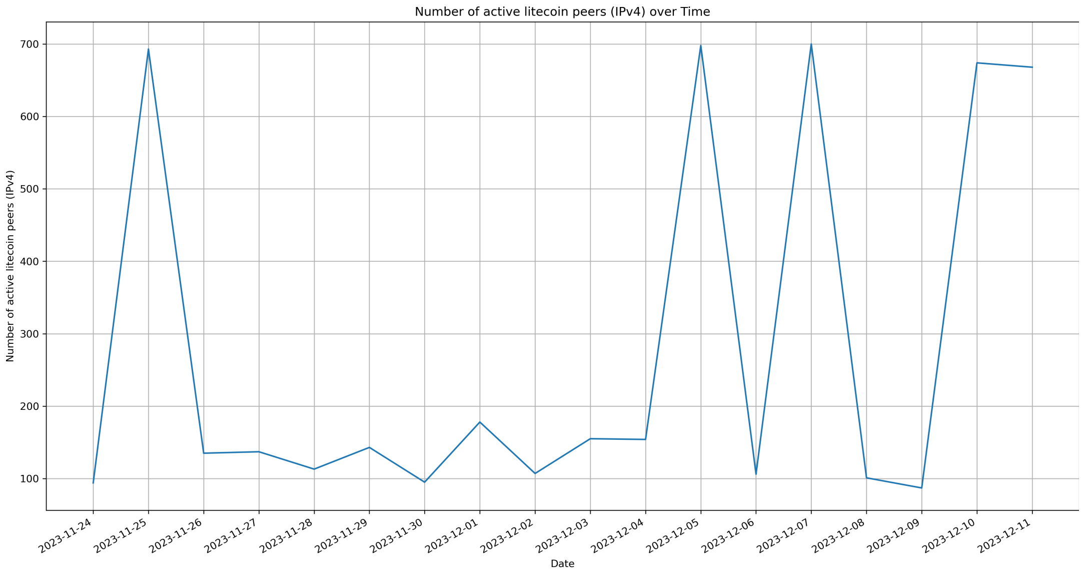


Figure B.5: Litecoin Peer Activity Over Time

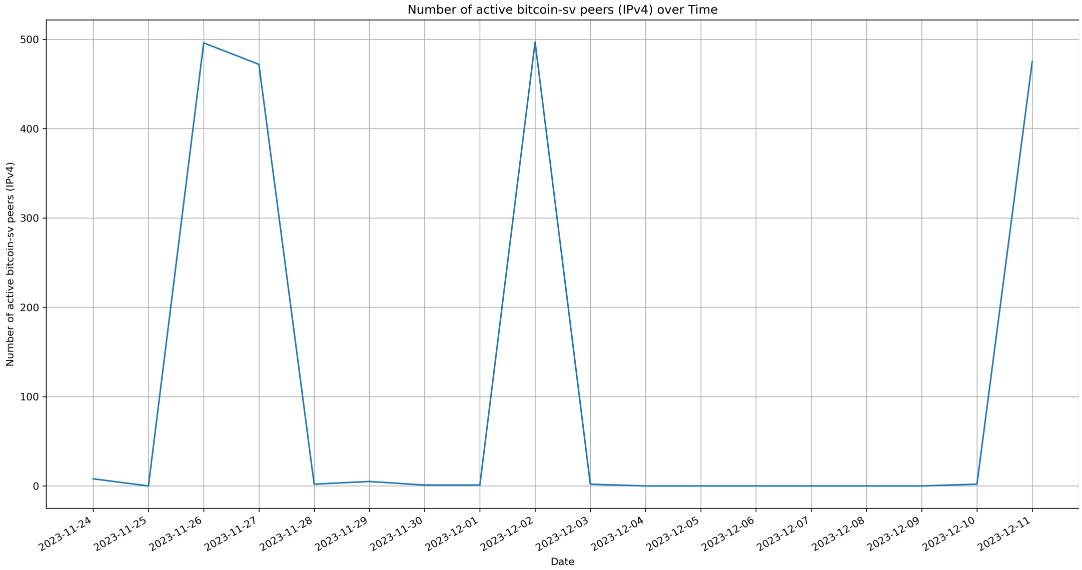


Figure B.6: Bitcoin-SV Peer Activity Over Time

B.2 Geographic Distribution of Cryptocurrency Peers

B.2.1 Plots over time

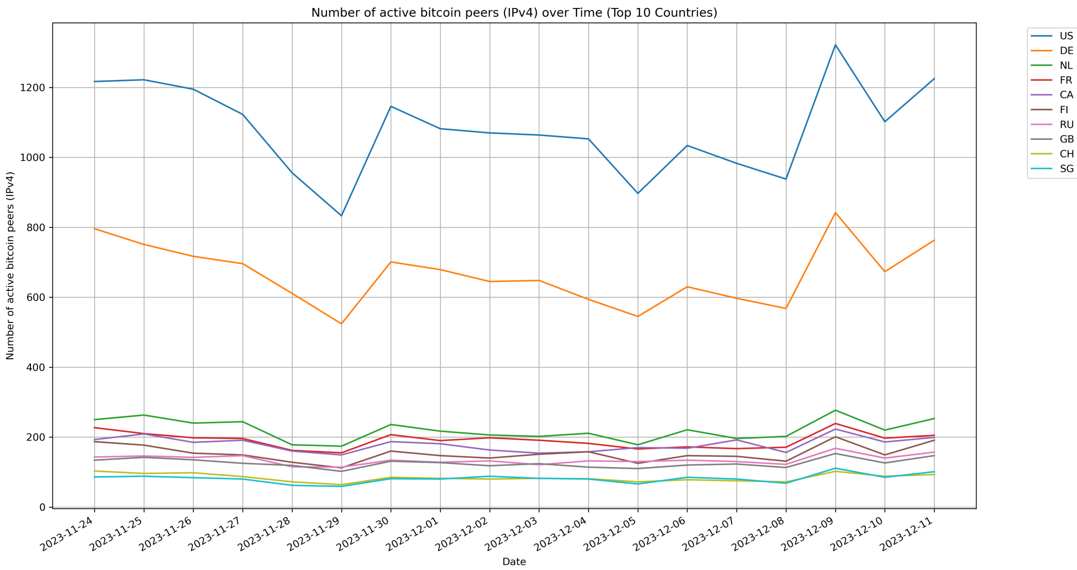


Figure B.7: Geographic Distribution of Bitcoin Peers Over Time

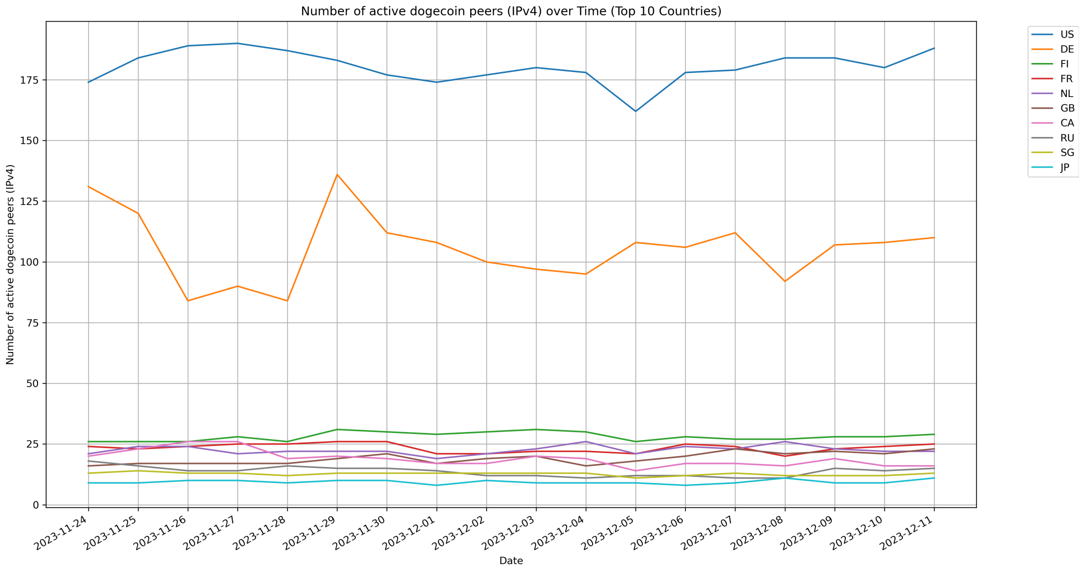


Figure B.8: Geographic Distribution of Dogecoin Peers Over Time

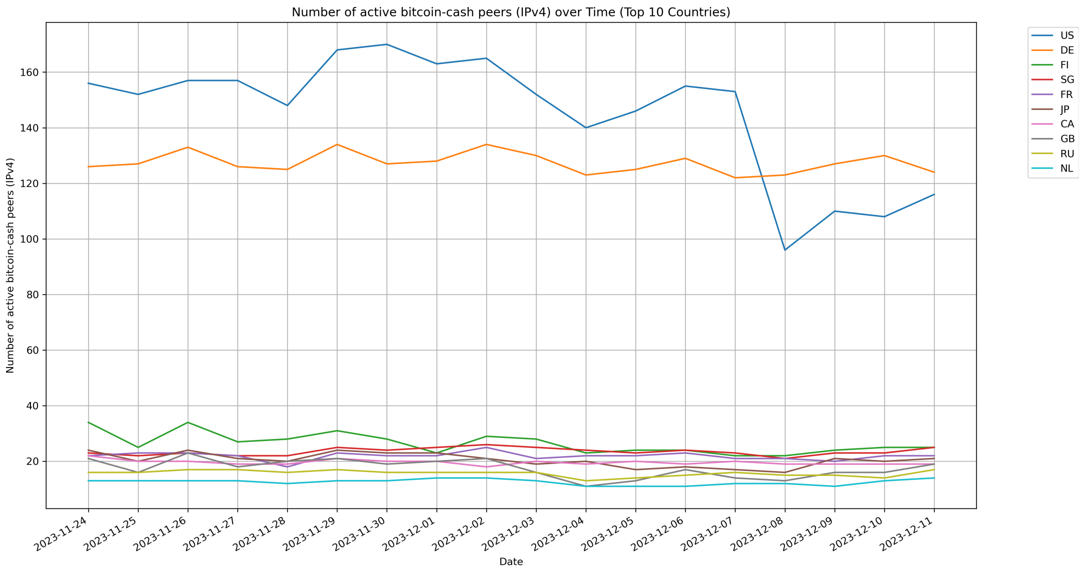


Figure B.9: Geographic Distribution of Bitcoin-Cash Peers Over Time

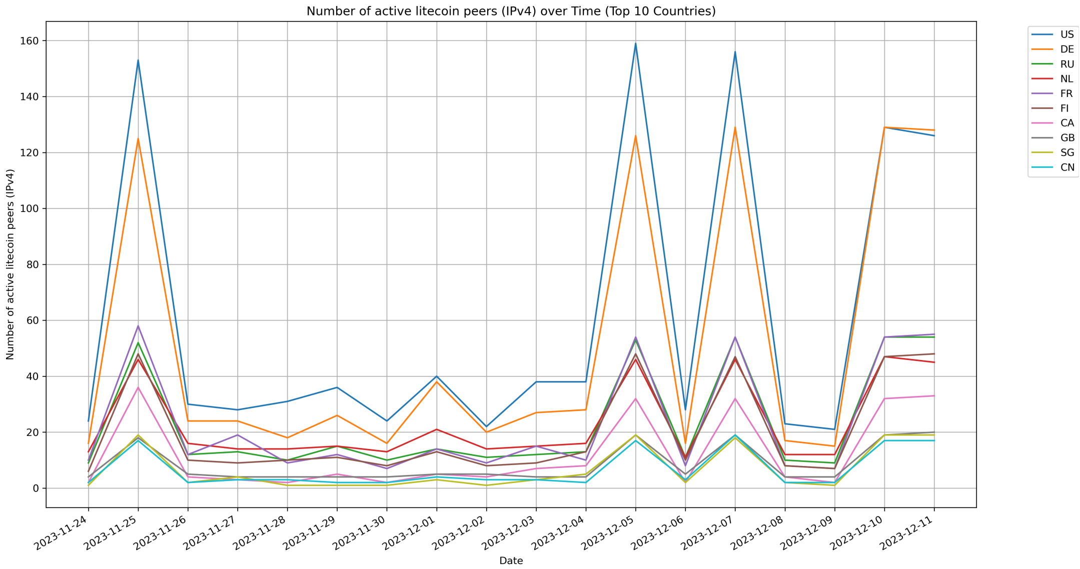


Figure B.10: Geographic Distribution of Litecoin Peers Over Time

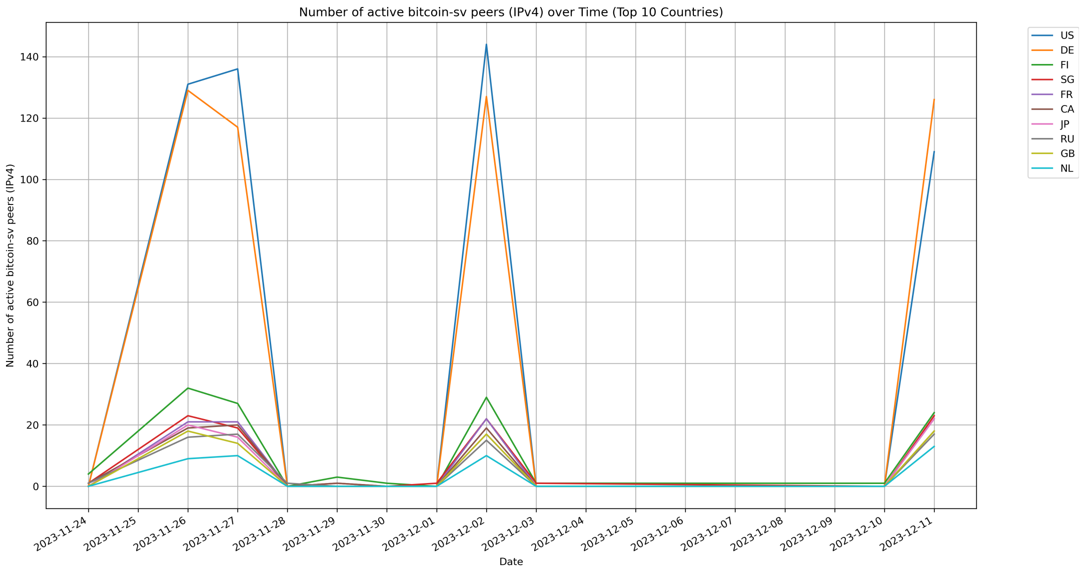


Figure B.11: Geographic Distribution of Bitcoin-SV Peers Over Time

B.2.2 Pie Charts

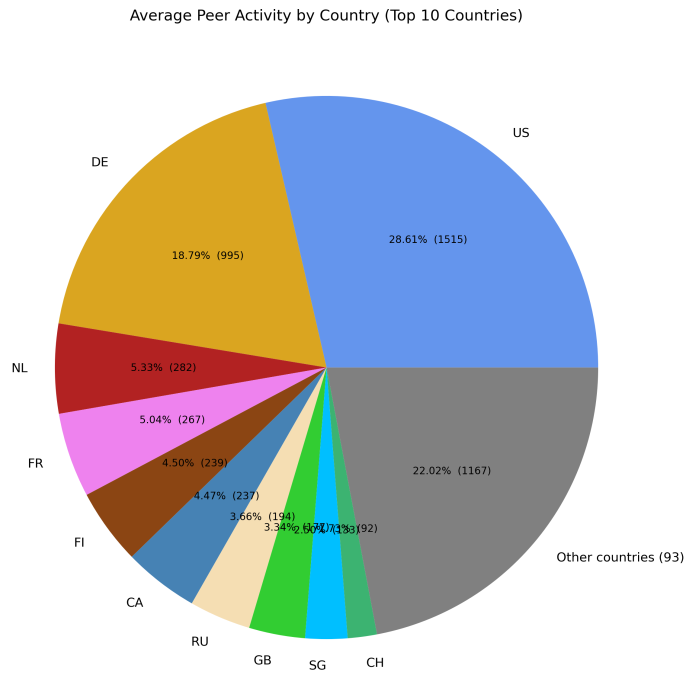


Figure B.12: Geographic Distribution of all Peers

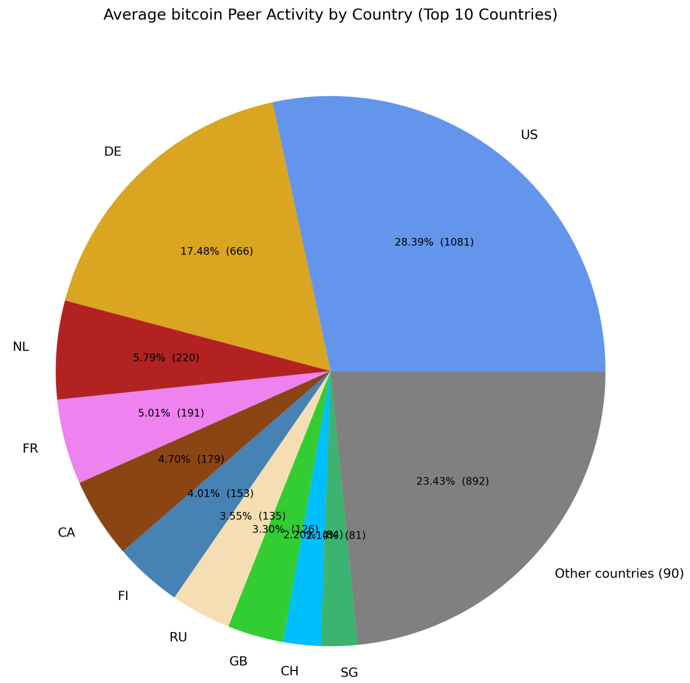


Figure B.13: Geographic Distribution of Bitcoin Peers

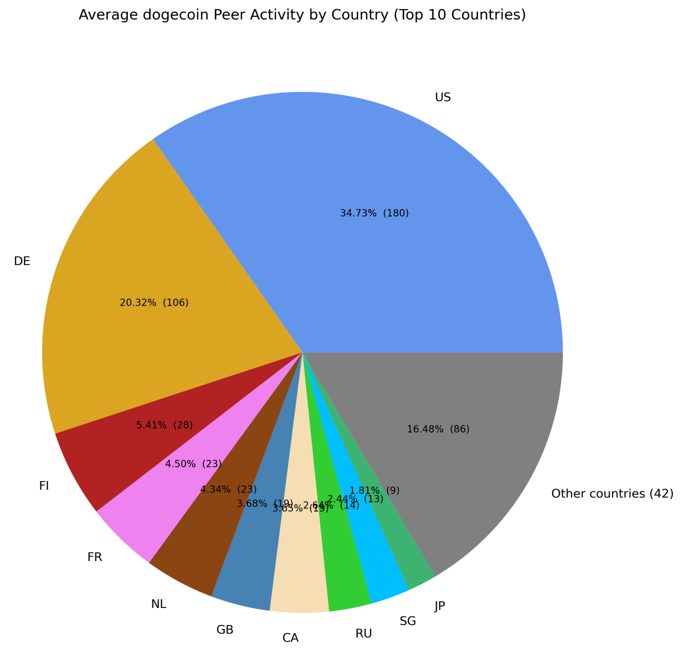


Figure B.14: Geographic Distribution of Dogecoin Peers

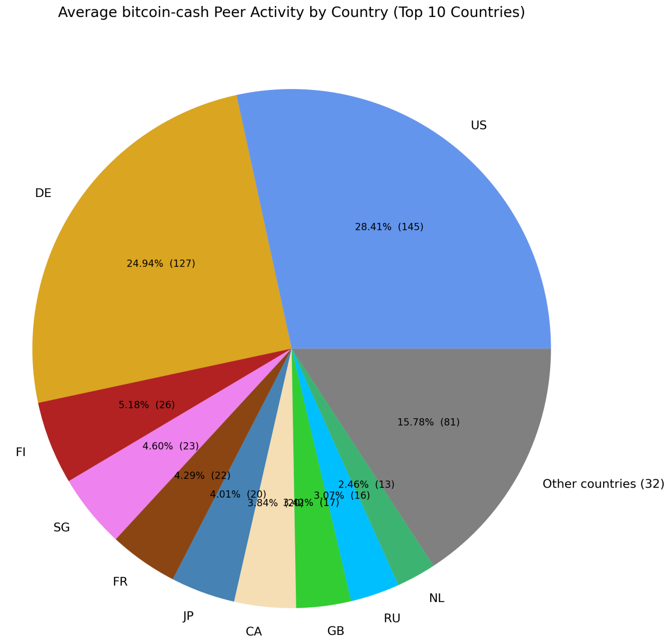


Figure B.15: Geographic Distribution of Bitcoin-Cash Peers

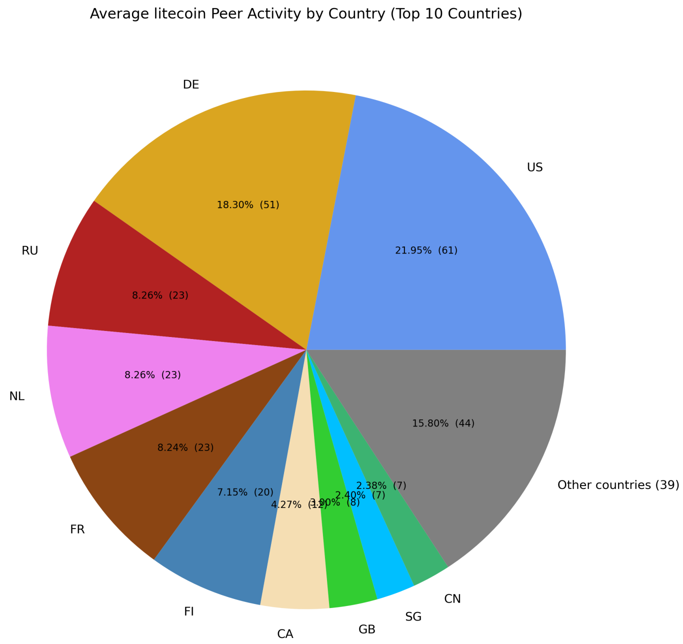


Figure B.16: Geographic Distribution of Litecoin Peers

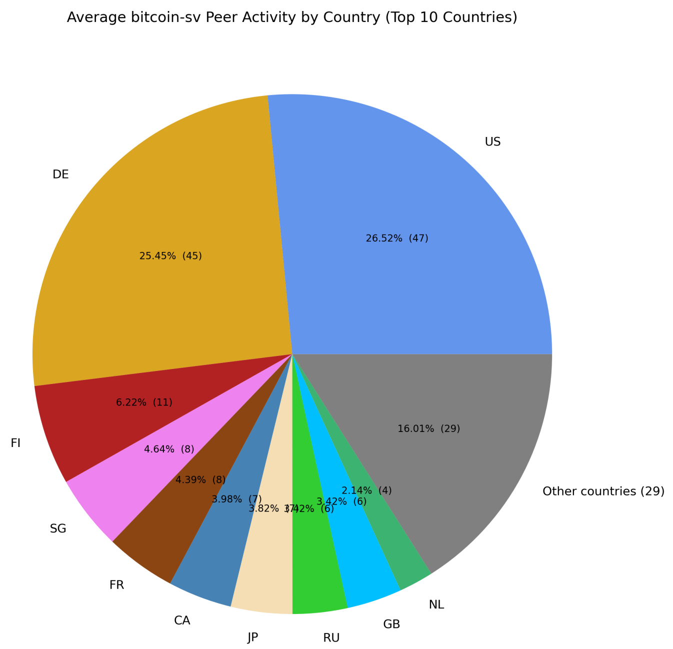


Figure B.17: Geographic Distribution of Bitcoin-SV Peers

B.2.3 Heatmaps

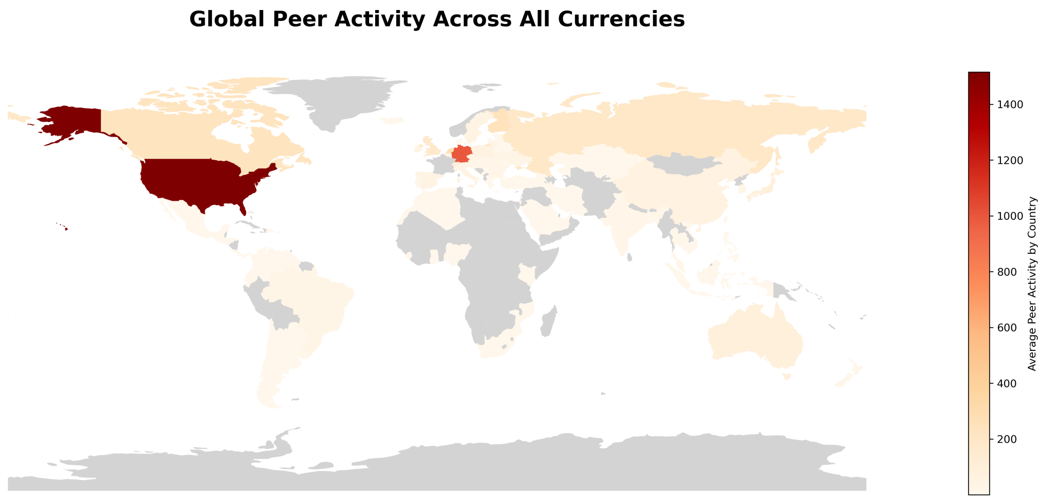


Figure B.18: Heatmap of Active Peers

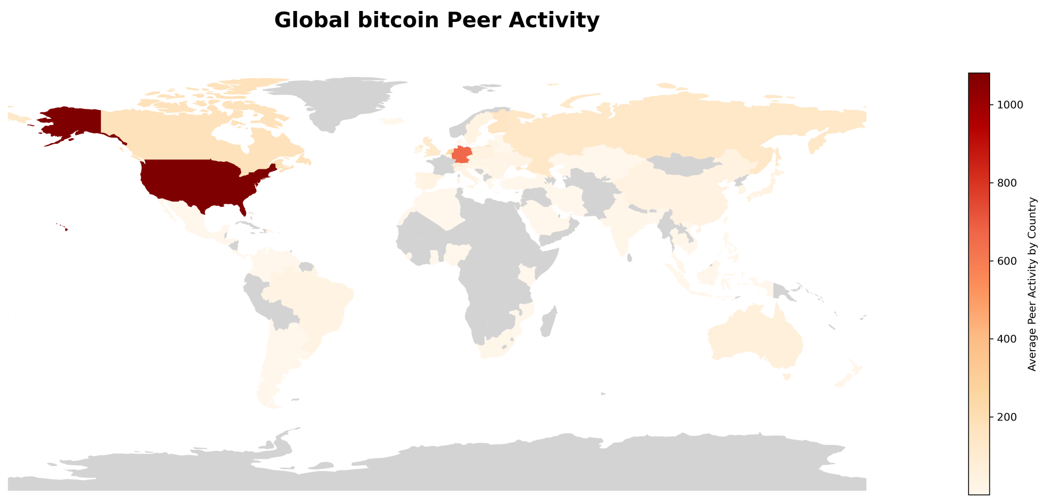


Figure B.19: Heatmap of Active Bitcoin Peers

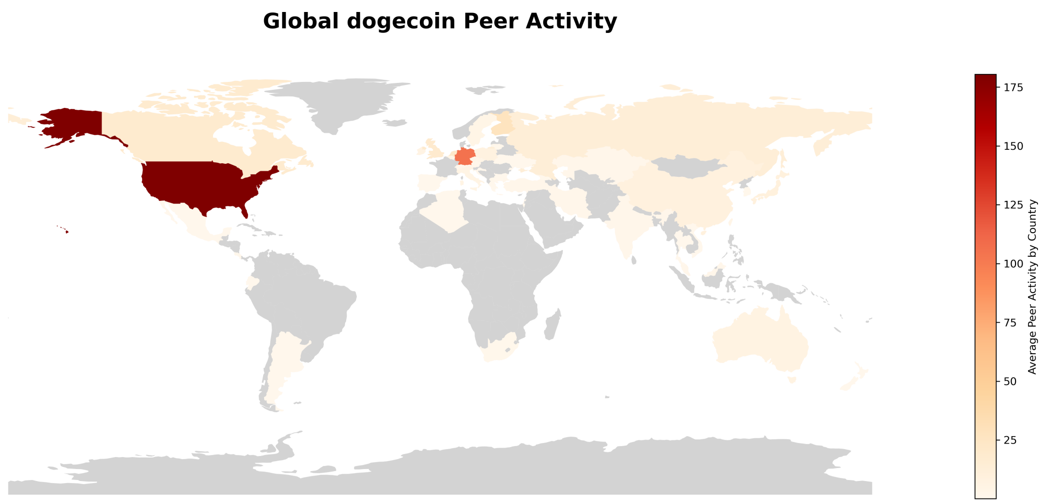


Figure B.20: Heatmap of Active Dogecoin Peers

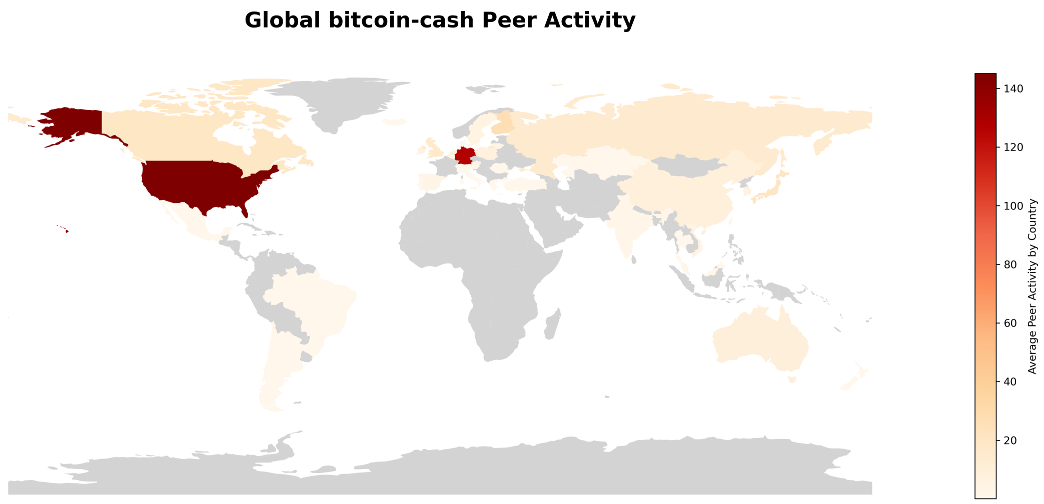


Figure B.21: Heatmap of Active Bitcoin-Cash Peers

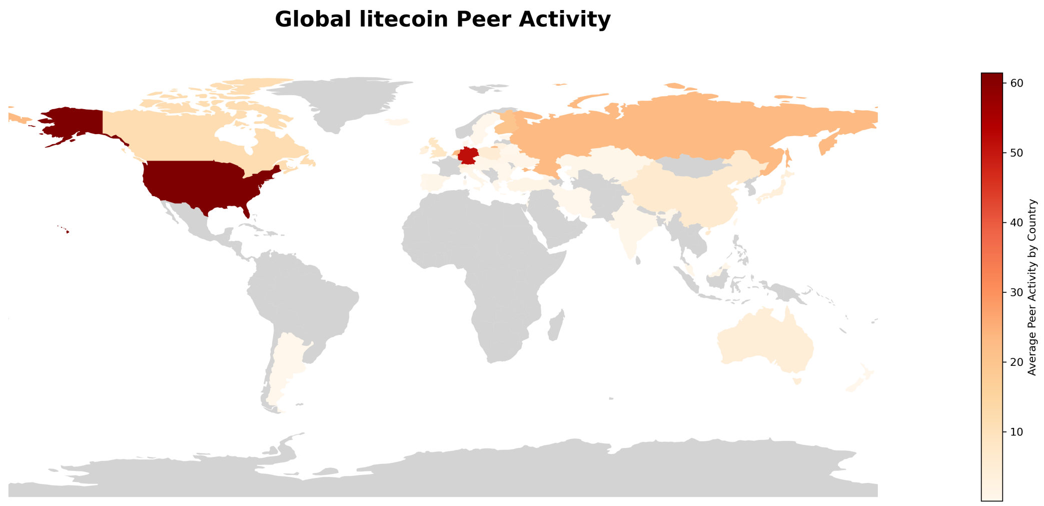


Figure B.22: Heatmap of Active Litecoin Peers

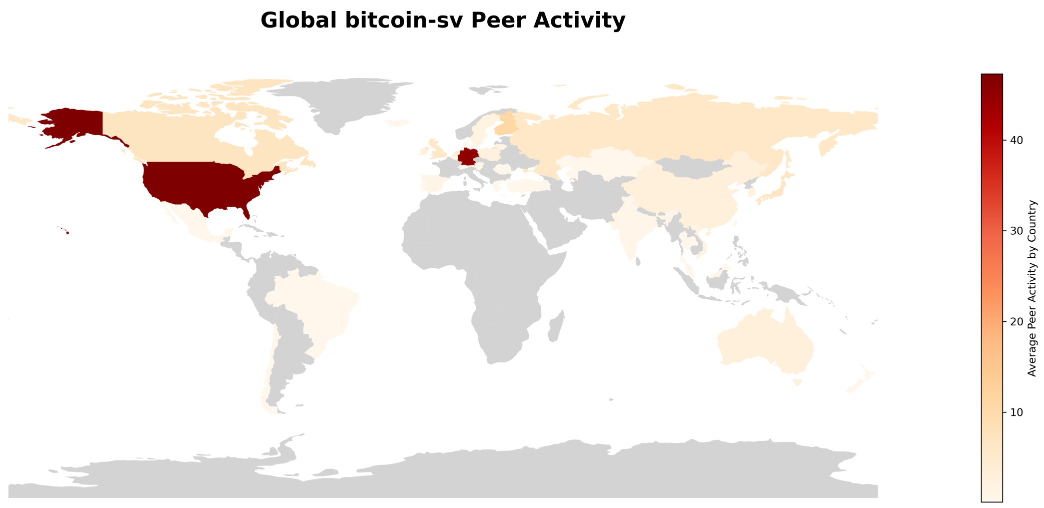


Figure B.23: Heatmap of Active Bitcoin-SV Peers

B.3 Distribution of Cryptocurrency Peers by ASN

B.3.1 Plots over time

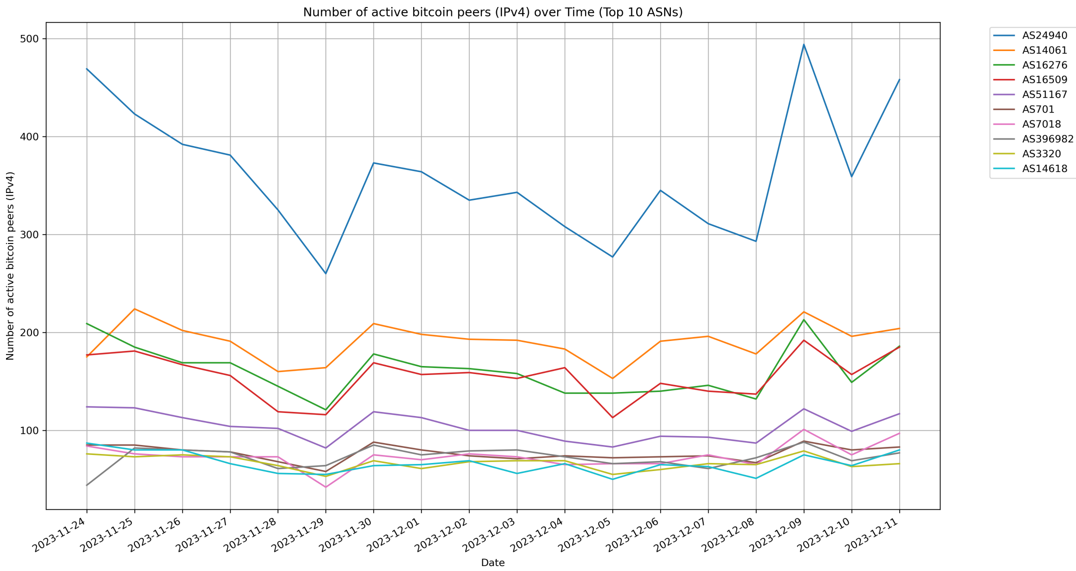


Figure B.24: ASN Distribution of Bitcoin Peers

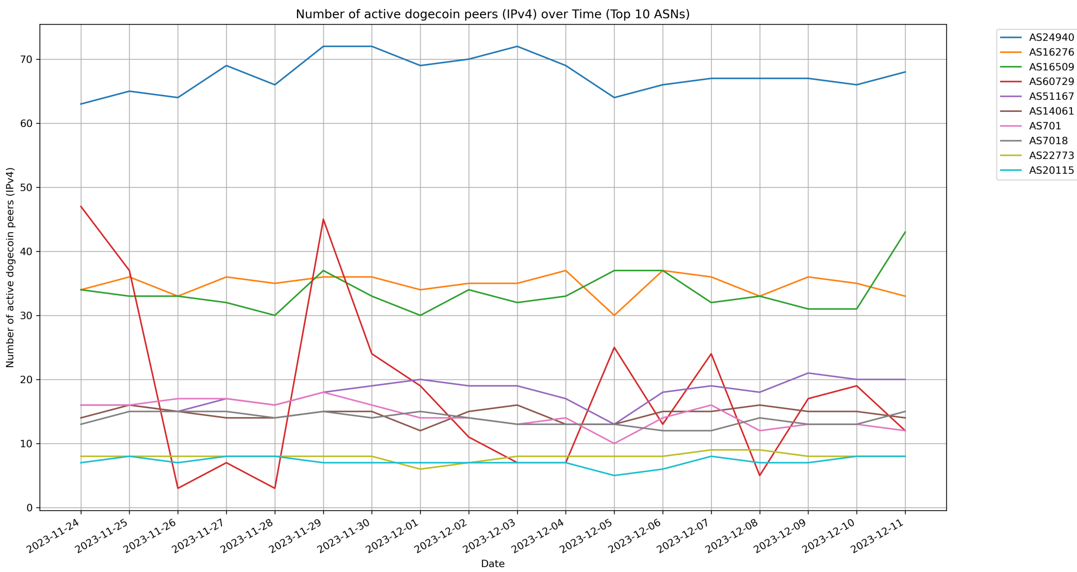


Figure B.25: ASN Distribution of Dogecoin Peers

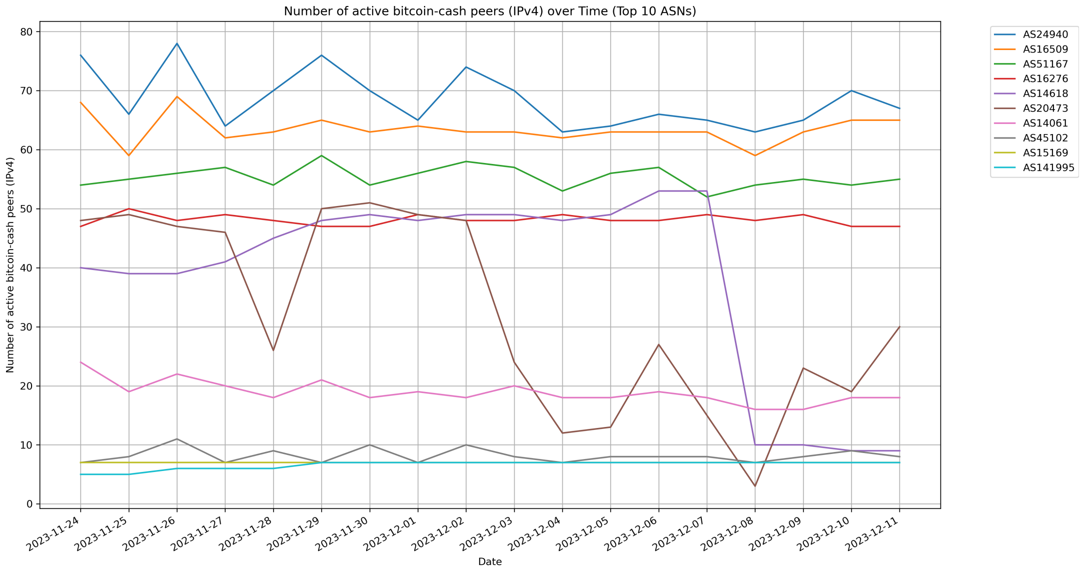


Figure B.26: ASN Distribution of Peers Bitcoin-Cash

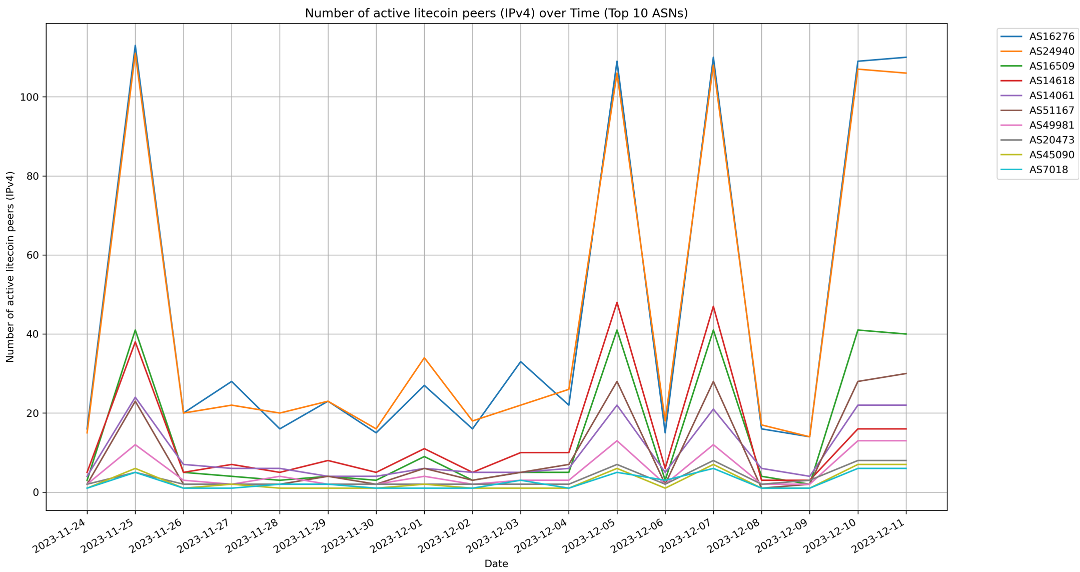


Figure B.27: ASN Distribution of Peers Litecoin

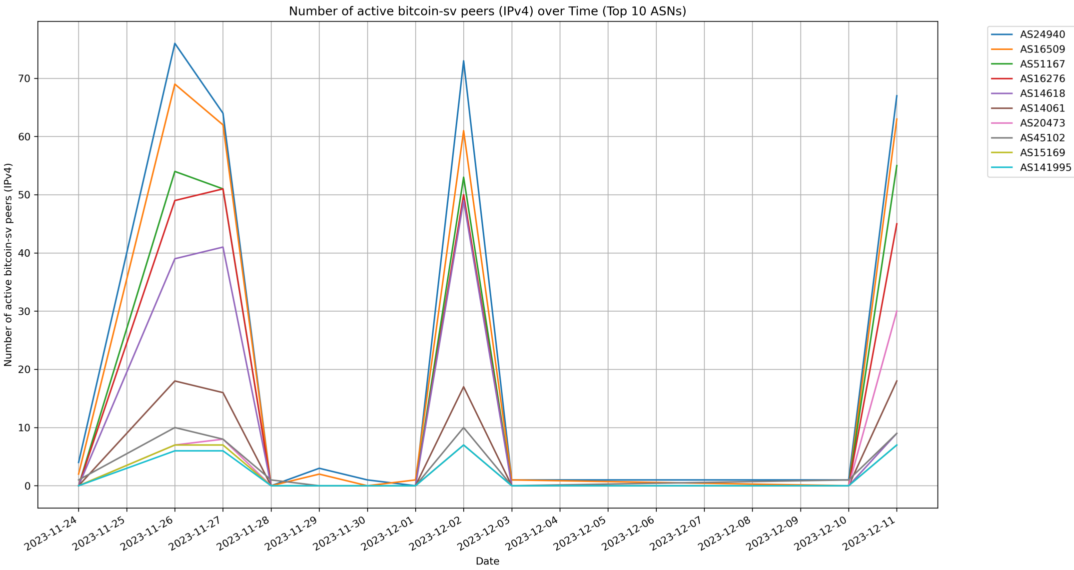


Figure B.28: ASN Distribution of Peers Bitcoin-SV

B.3.2 Pie Charts

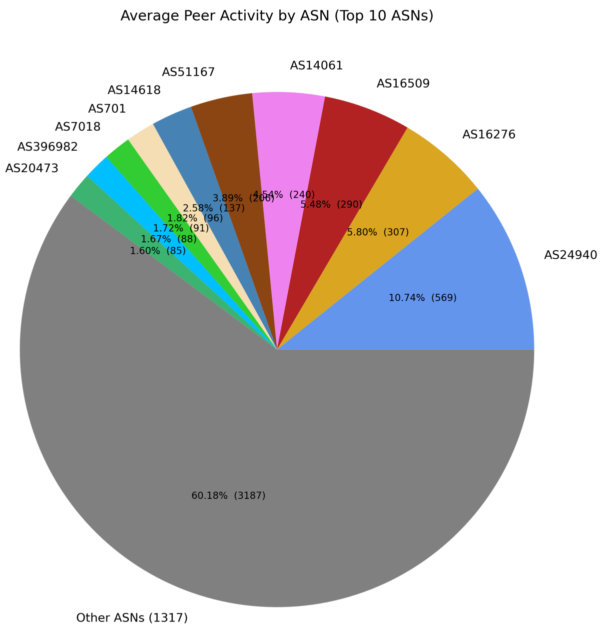


Figure B.29: ASN Distribution of all Peers

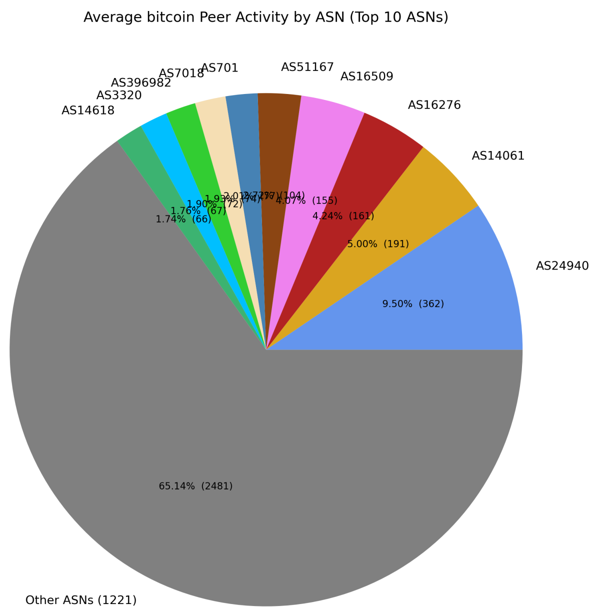


Figure B.30: ASN Distribution of Bitcoin Peers

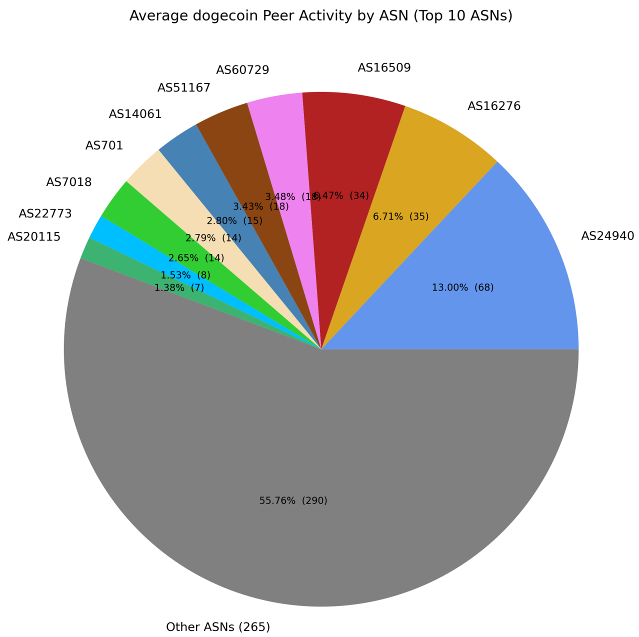


Figure B.31: ASN Distribution of Dogecoin Peers

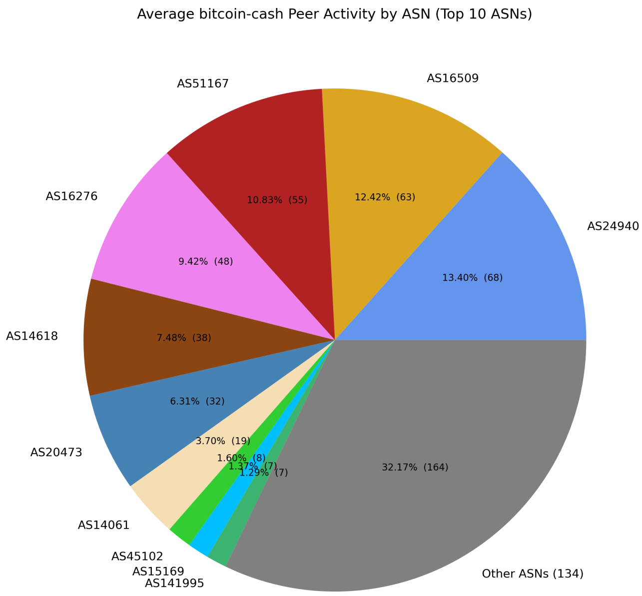


Figure B.32: ASN Distribution of Bitcoin-Cash Peers

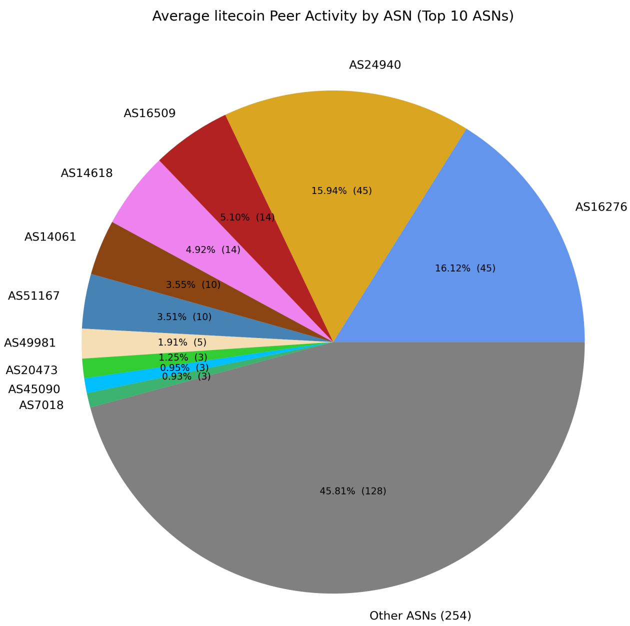


Figure B.33: ASN Distribution of Litecoin Peers

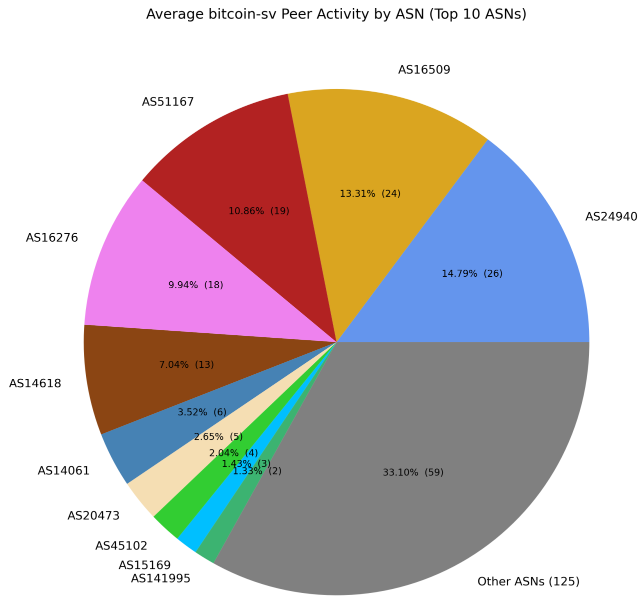


Figure B.34: ASN Distribution of Bitcoin-SV Peers

B.4 Distribution of Cryptocurrency Peers by Organisation

B.4.1 Plots over time

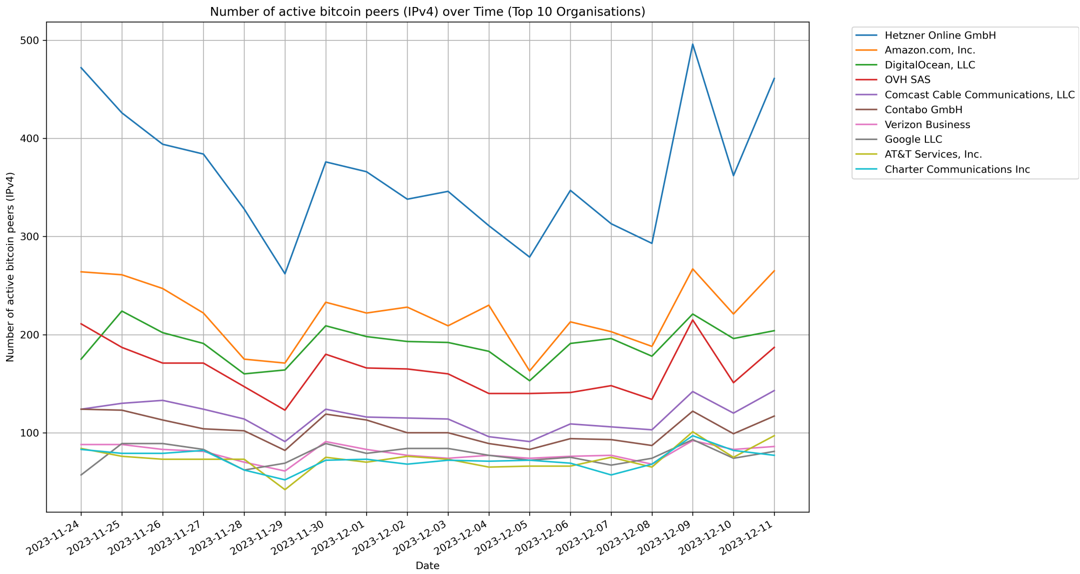


Figure B.35: Organisation Distribution of Bitcoin Peers

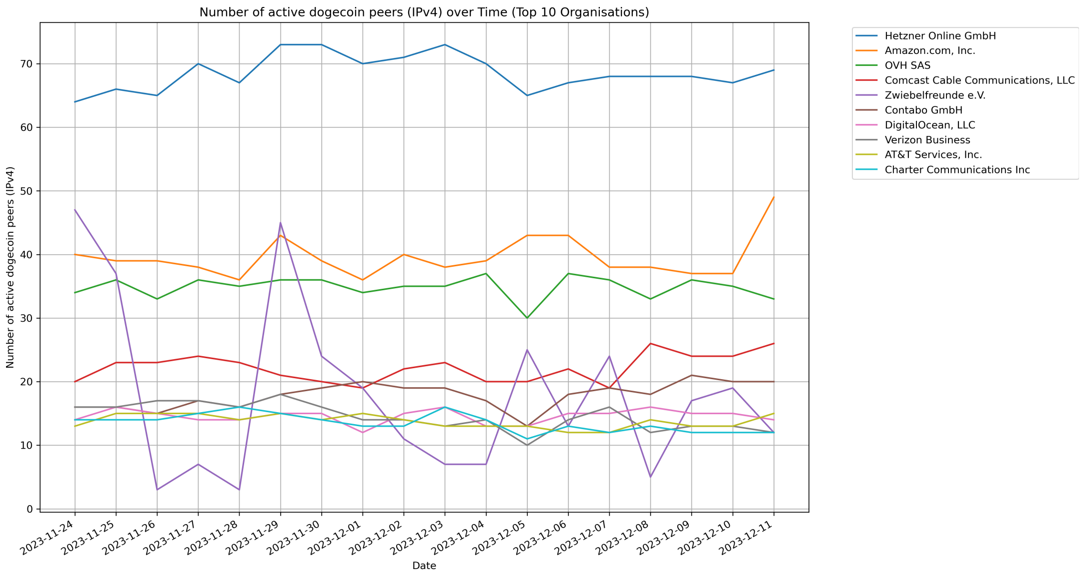


Figure B.36: Organisation Distribution of Dogecoin Peers

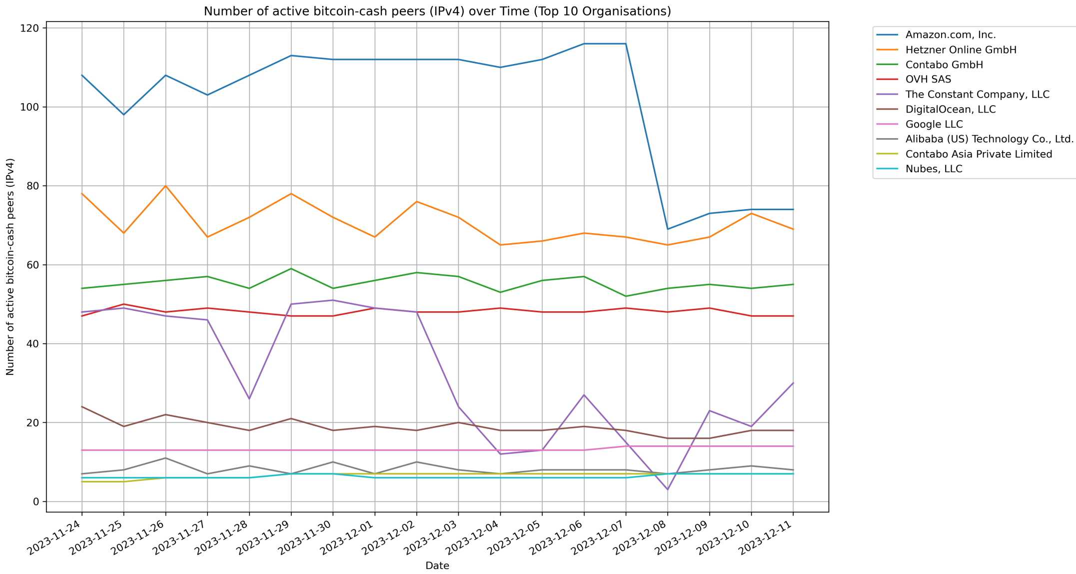


Figure B.37: Organisation Distribution of Peers Bitcoin-Cash

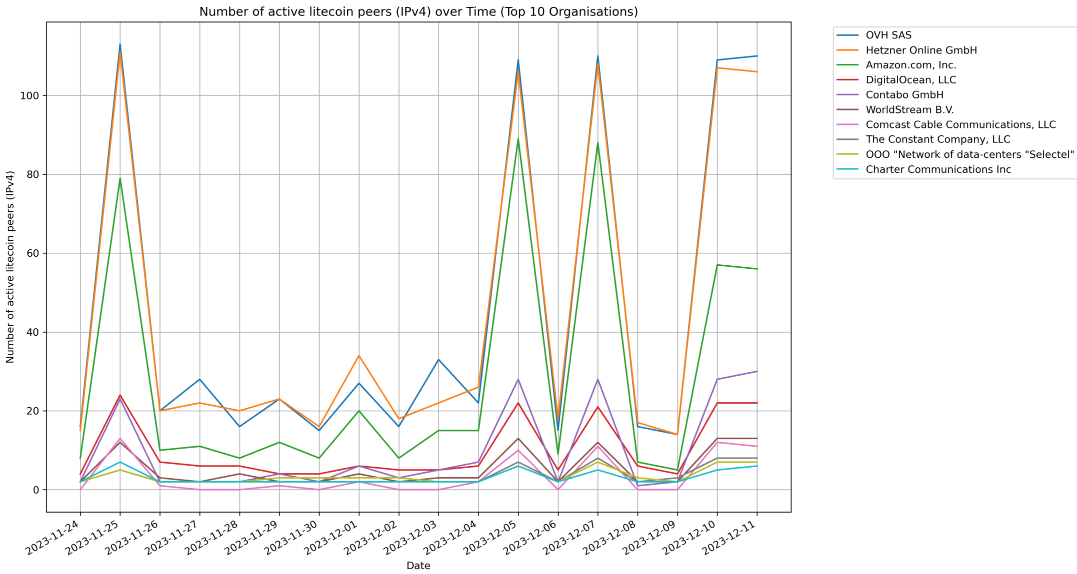


Figure B.38: Organisation Distribution of Peers Litecoin

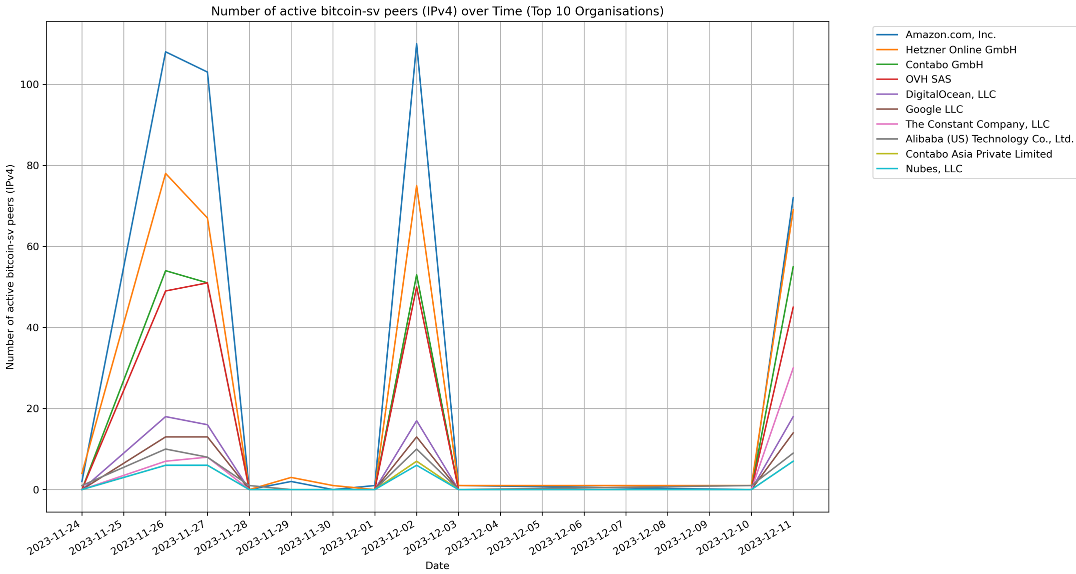


Figure B.39: Organisation Distribution of Peers Bitcoin-SV

B.4.2 Pie Charts

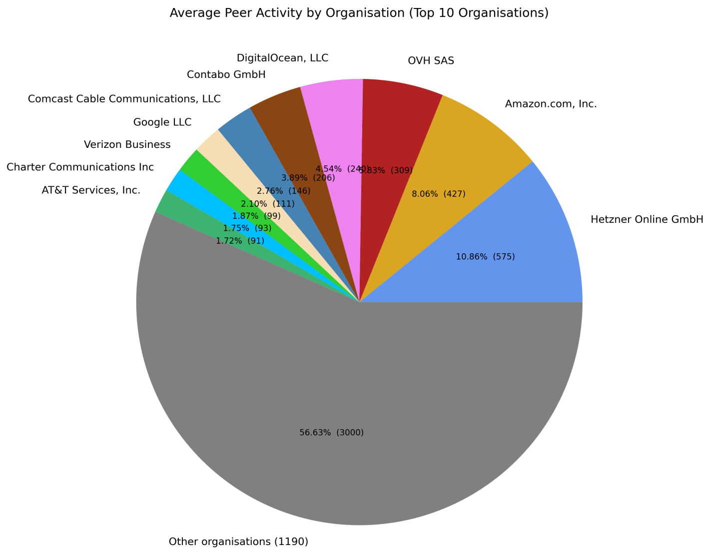


Figure B.40: Organisation Distribution of all Peers

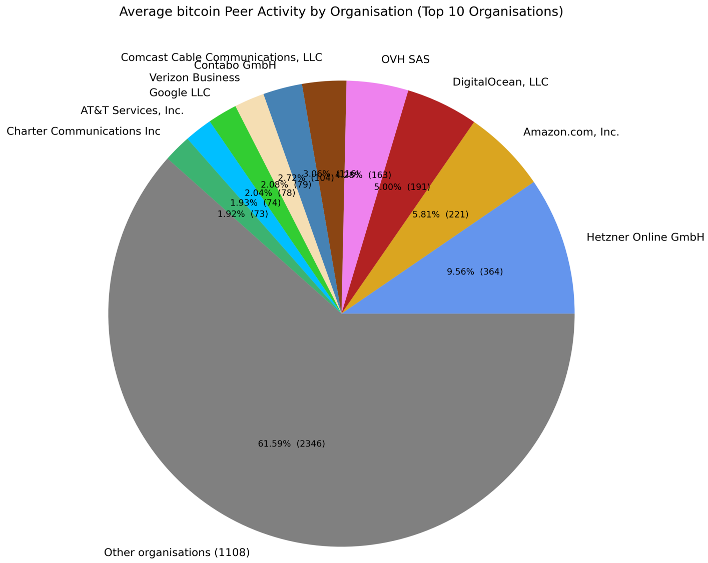


Figure B.41: Organisation Distribution of Bitcoin Peers

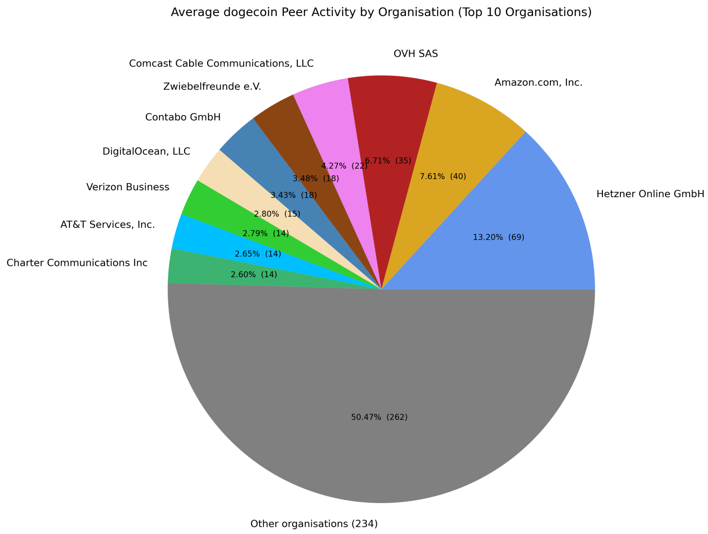


Figure B.42: Organisation Distribution of Dogecoin Peers

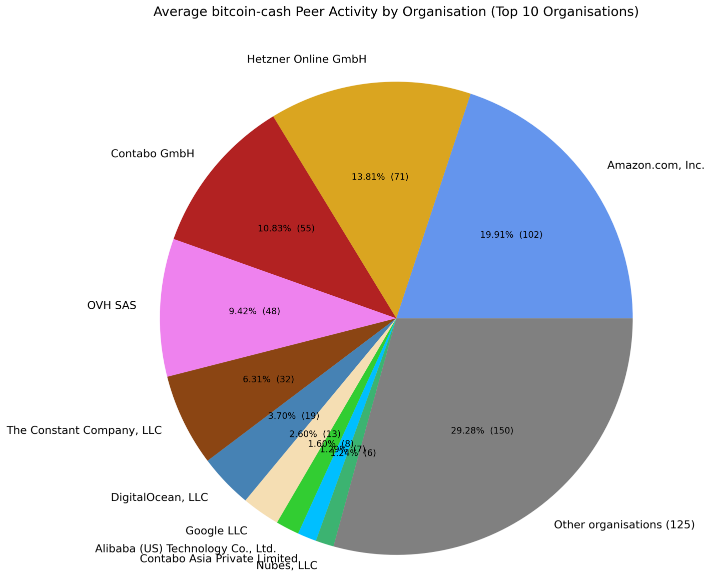


Figure B.43: Organisation Distribution of Bitcoin-Cash Peers

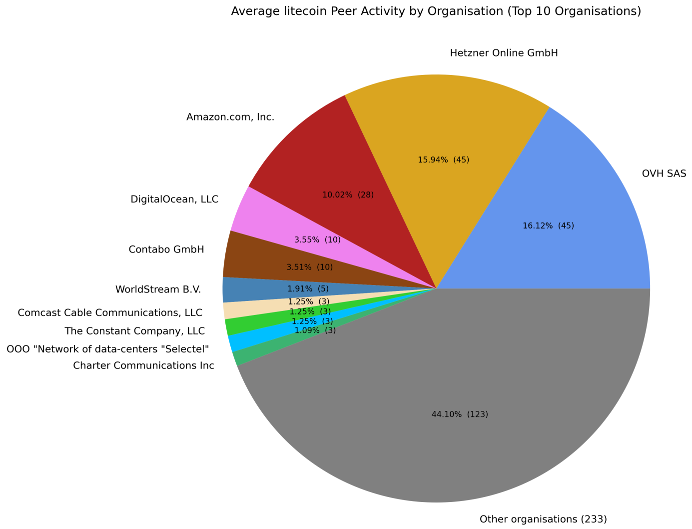


Figure B.44: Organisation Distribution of Litecoin Peers

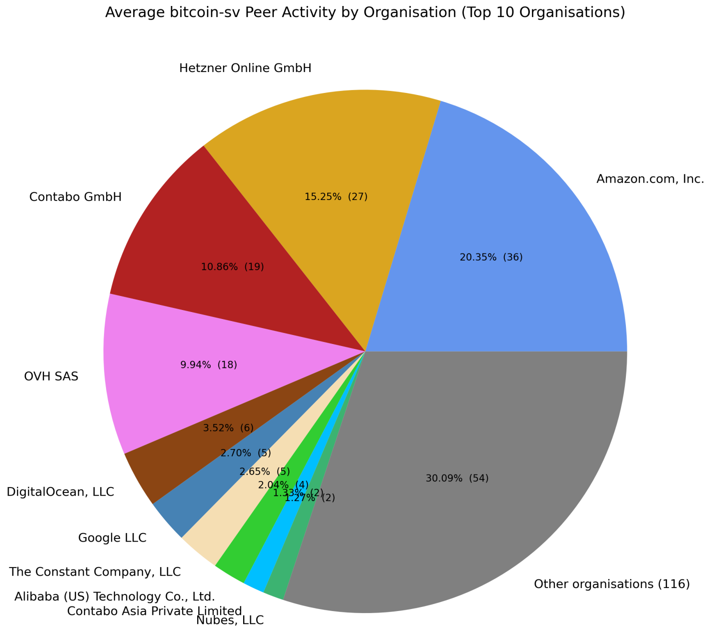


Figure B.45: Organisation Distribution of Bitcoin-SV Peers

B.5 Activity by Number of Days

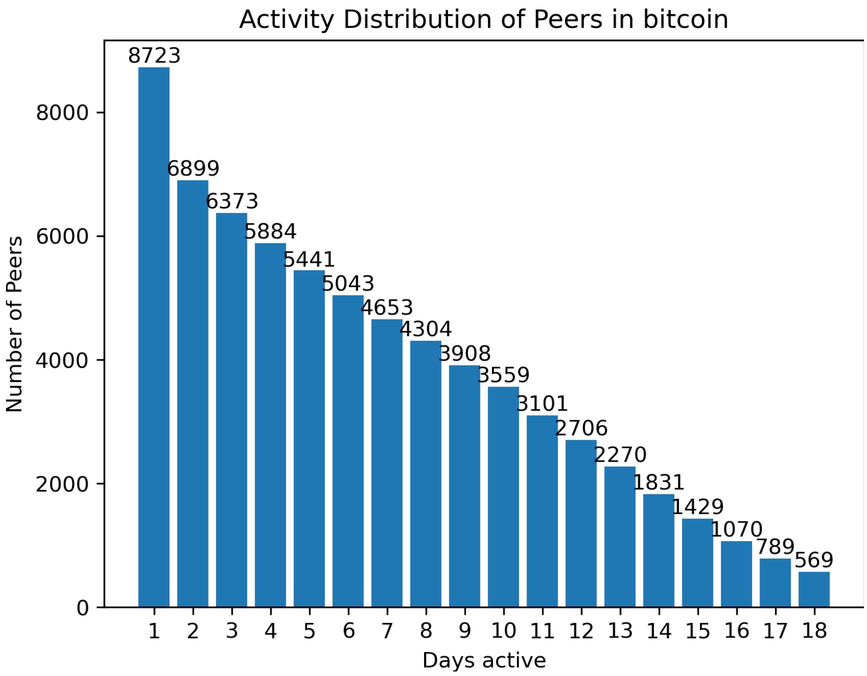


Figure B.46: Distribution of Active Days for Bitcoin Peers

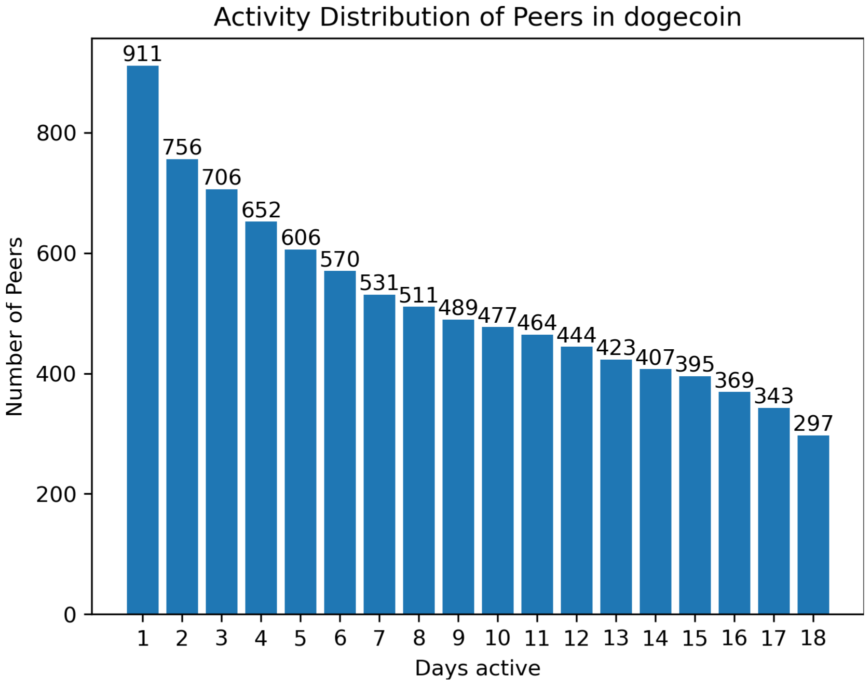


Figure B.47: Distribution of Active Days for Dogecoin Peers

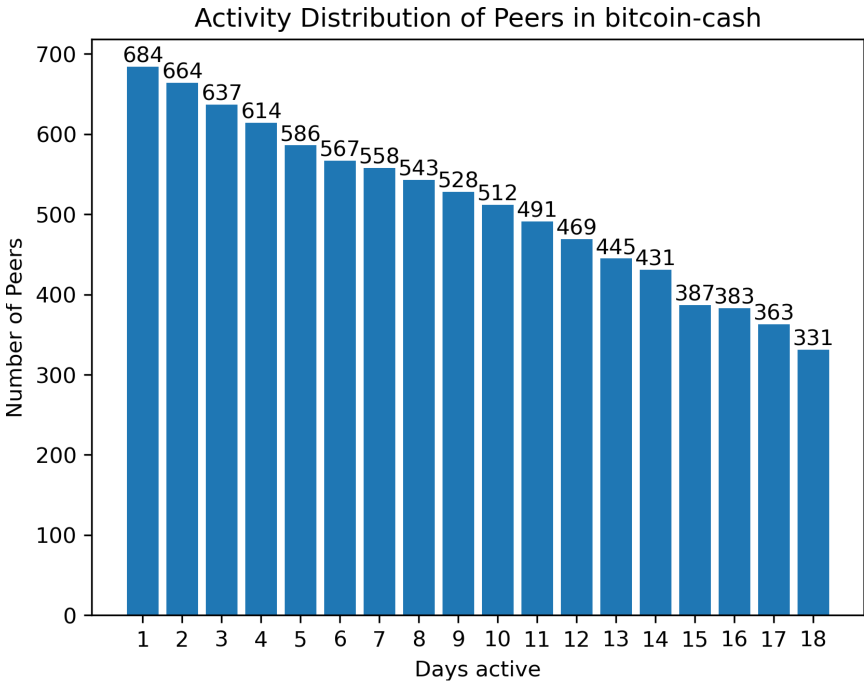


Figure B.48: Distribution of Active Days for Bitcoin Cash Peers

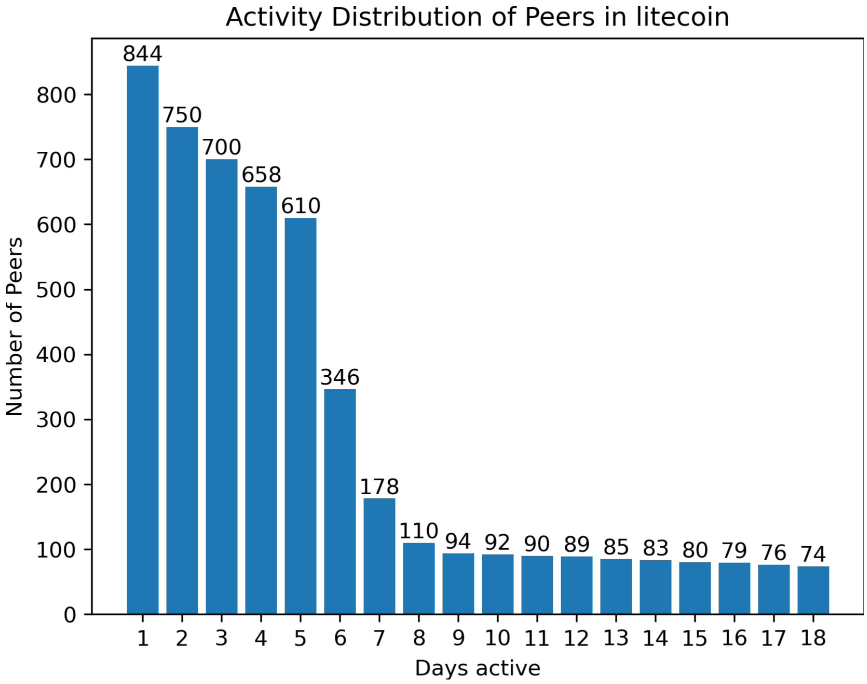


Figure B.49: Distribution of Active Days for Litecoin Peers

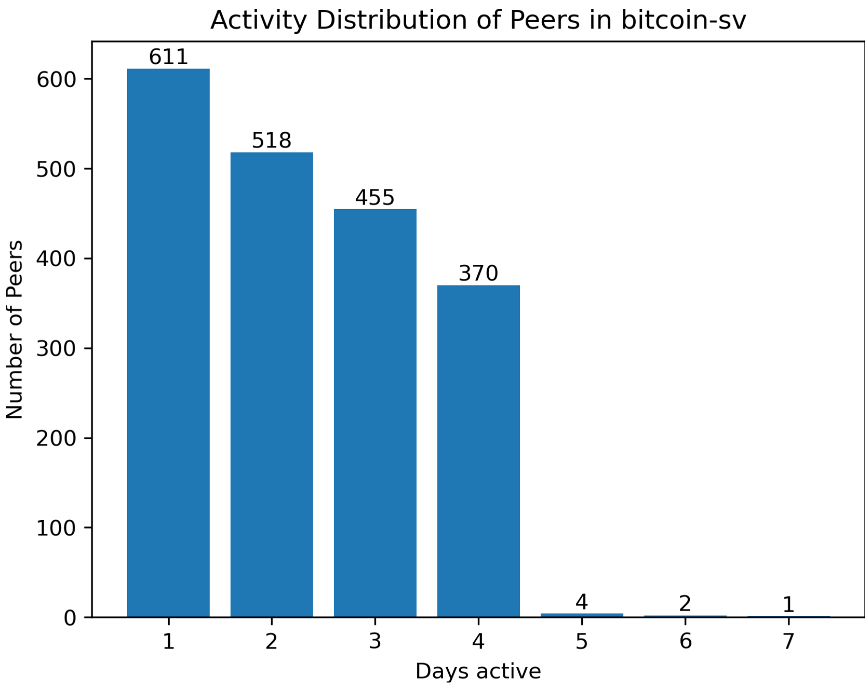


Figure B.50: Distribution of Active Days for Bitcoin SV Peers

B.6 Cross-Network Peer Participation

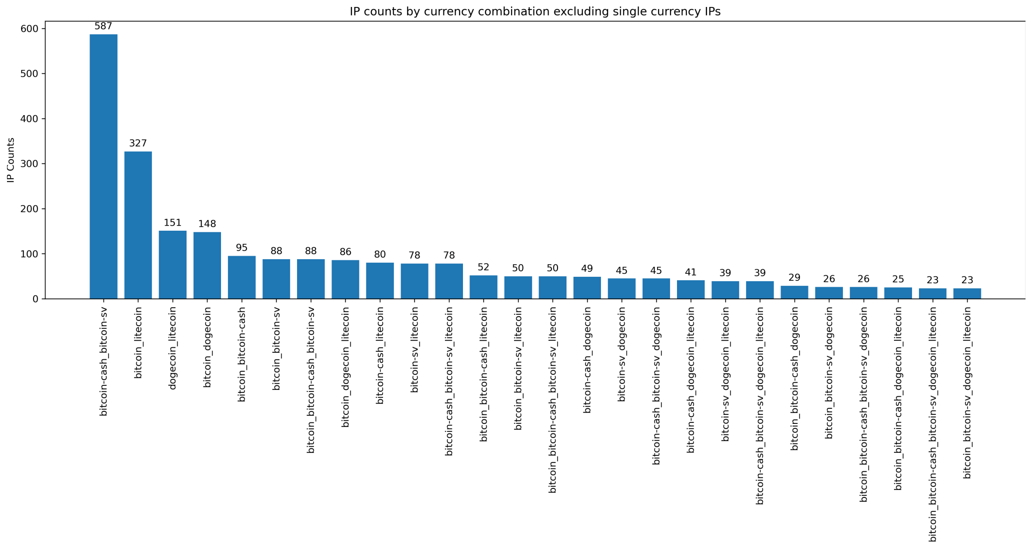


Figure B.51: IP Address Overlap Across Cryptocurrency Networks

B.7 Temporal Stability of Cryptocurrency Networks

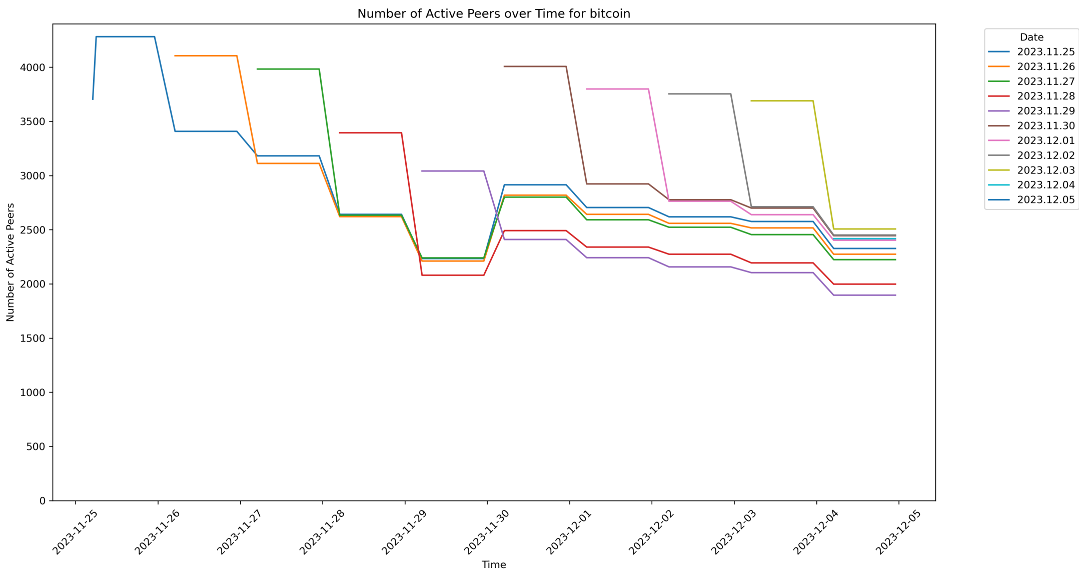


Figure B.52: The graph tracks the number of originally active Bitcoin peers that remain responsive over time.

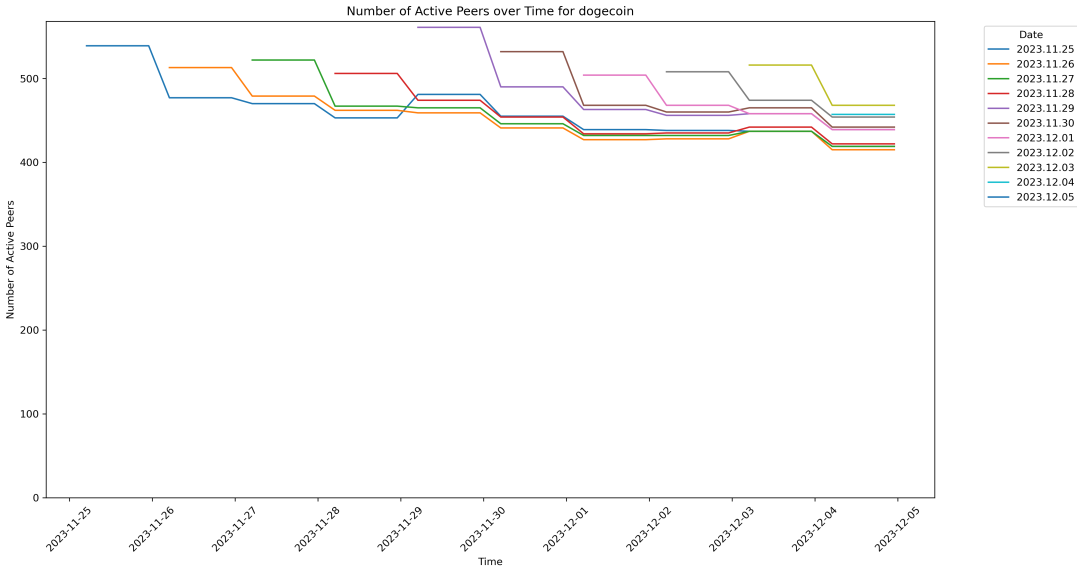


Figure B.53: The graph tracks the number of originally active Dogecoin peers that remain responsive over time.

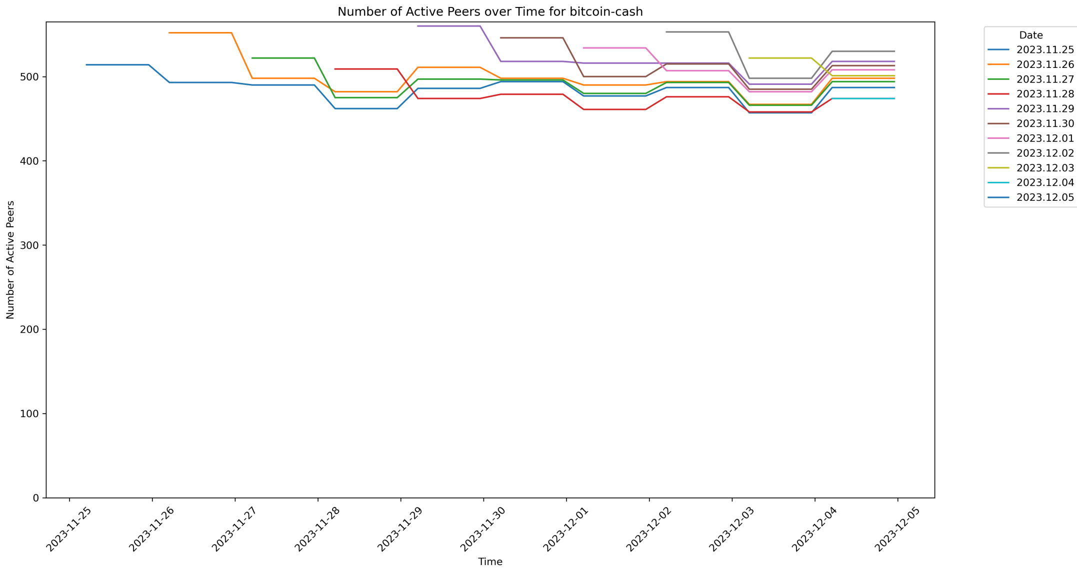


Figure B.54: The graph tracks the number of originally active Bitcoin-Cash peers that remain responsive over time.

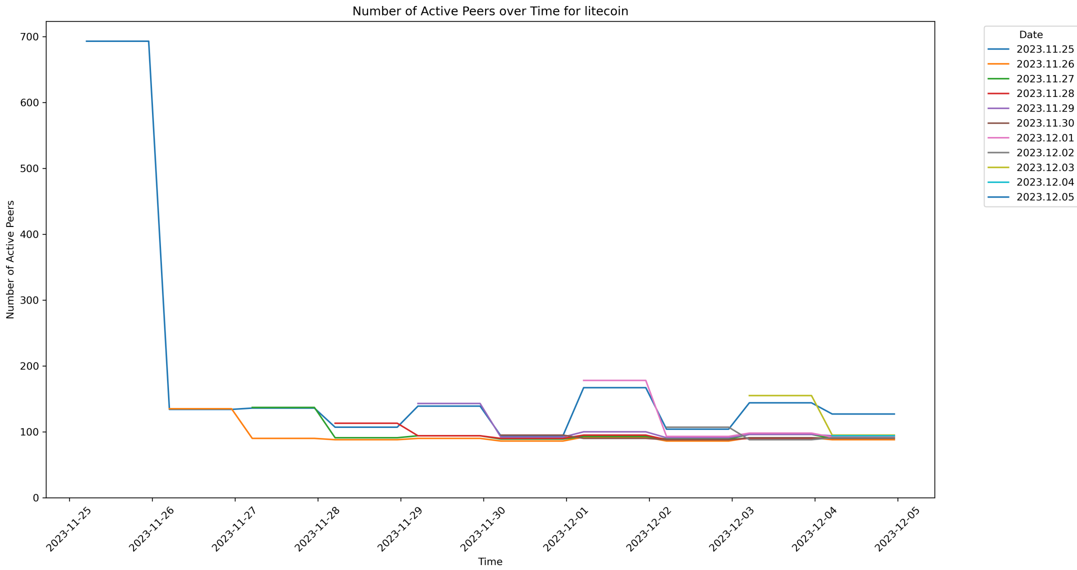


Figure B.55: The graph tracks the number of originally active Litecoin peers that remain responsive over time.

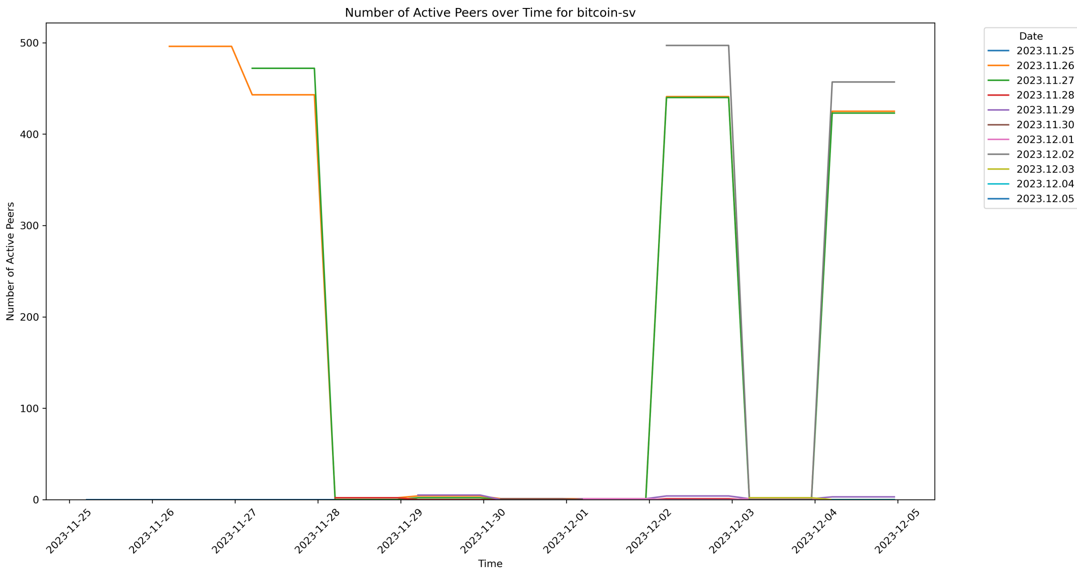


Figure B.56: The graph tracks the number of originally active Bitcoin-SV peers that remain responsive over time.

B.8 Comparative Study

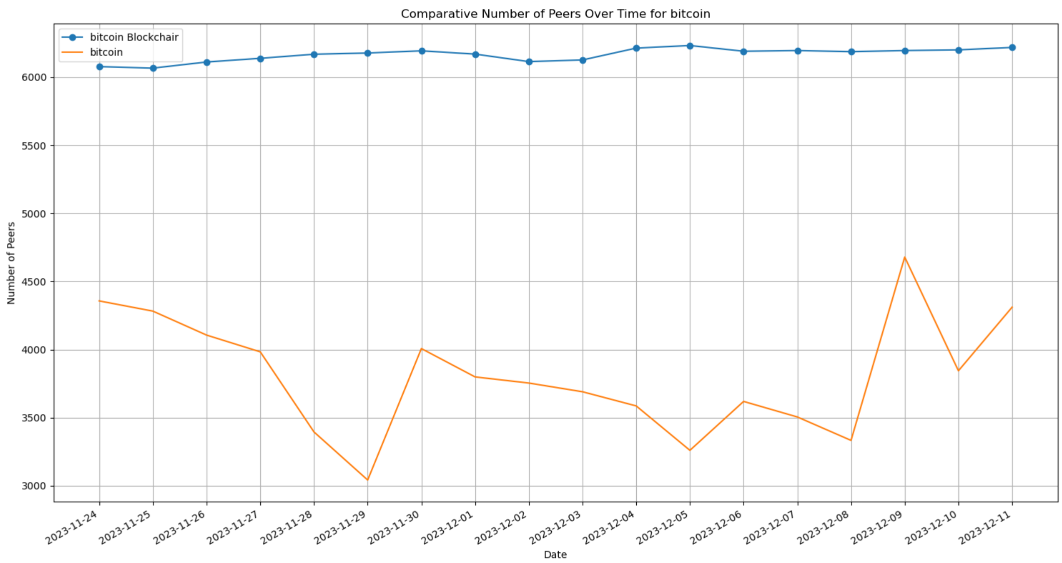


Figure B.57: Comparison of peer counts over time for Bitcoin and Bitcoin using Blockchair data.

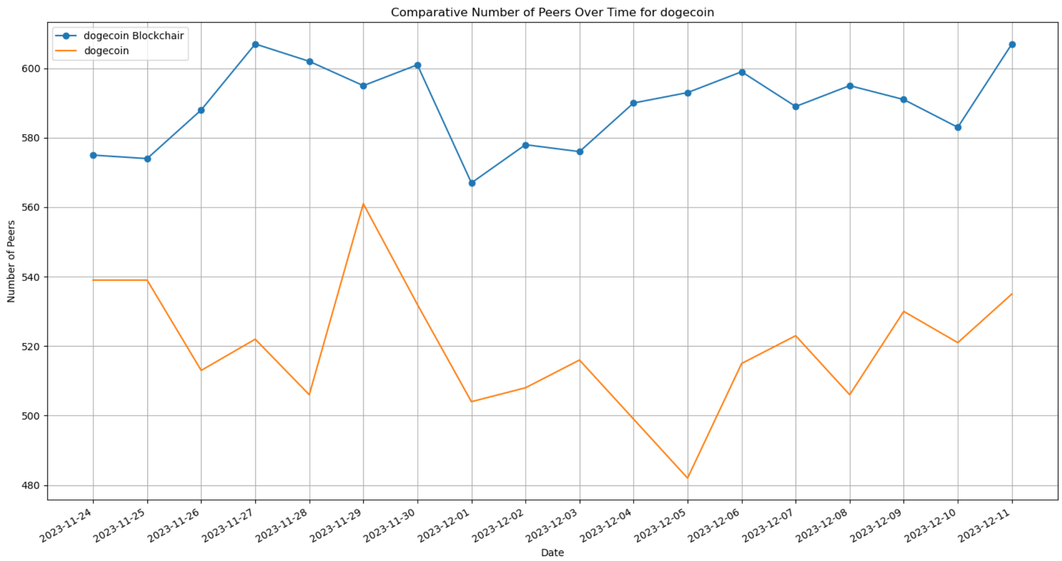


Figure B.58: Comparison of peer counts over time for Dogecoin and Dogecoin using Blockchair data.

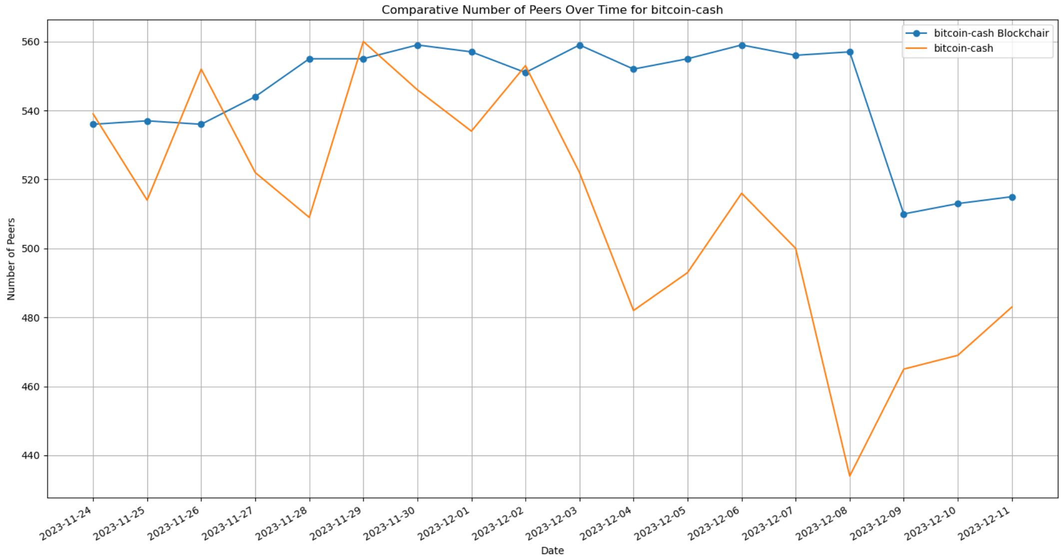


Figure B.59: Comparison of peer counts over time for Bitcoin-Cash and Bitcoin-Cash using Blockchair data.

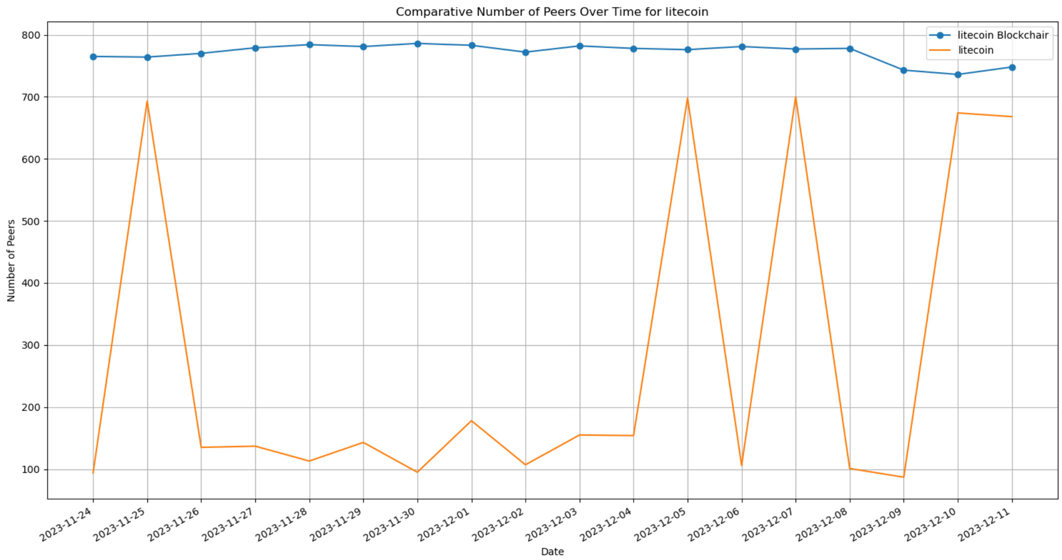


Figure B.60: Comparison of peer counts over time for Litecoin and Litecoin using Blockchair data.