Analysis on Ethereum's Decentralized Finance

1. Abstract

Blockchain technology is becoming prevalent in today's world as it offers decentralized applications to different businesses. Ethereum, in particular, is a decentralized and open-source blockchain with functionalities of smart contracts and tokens. This new type of blockchain has numerous potential applications and in the recent years, it has found a way into decentralized finance. Ethereum provides a stable ecosystem for modern financial applications to thrive in as its functionalities of smart contracts and tokens can optimize the financial processes with fewer resources and a greater reliability. Therefore, I am interested in how Ethereum's decentralized finance network operates, and I select the top 32 decentralized financial applications listed on the DeFi Leaderboard of Etherscan.io as research subjects. All the data regarding these applications are extracted from the Ethereum blockchain network and four types of networks are generated subsequently. Then I analyse and compare these networks using theories of graph properties, and as a result, meaningful observations and conclusions can be made.

Table of contents

2. List of tables and figures

3. Introduction

This paper aims to understand the network characteristics of Ethereum, specifically regarding its decentralized finance area, and explain these findings based on Ethereum's functionalities and environments.

Blockchain and cryptocurrency are developing to be some of the most game-changing technologies since the beginning of this century and their innovative applications have transformed the Internet and online businesses, especially in the development of finance and technology industries. With the arrival of Ethereum's smart contracts and tokens, decentralized applications of the blockchain technology pave the way for the new era of decentralized finance (DeFi), leading to a rapid growth in the number of decentralized financial applications on the Ethereum blockchain network.

The traditional financial infrastructure is centralized, and the entities involved such as banks and financial institutions are usually not transparent to the public and they charge huge fees for their services. The blockchain technology of the decentralized finance is structurally different and it can reduce transaction costs, provide transparency, and innovate new financial products, whereas the traditional financial infrastructure struggles to do so [1].

As Ethereum is the second biggest blockchain network after Bitcoin and it adopts the decentralized finance with smart contracts and tokens, Ethereum proves to be worthwhile to be researched and analysed. As of 2020, the finance category of the Ethereum decentralized applications' usage has the highest volume compared to Ethereum's other categories, and its transaction volume reached $9.5 billion [2].

However, decentralized finance on Ethereum is still in the premature stage currently, thus it is not ready for mass adoption. The decentralized finance has so far faced many risks and challenges yet, ranging from technical risks and usability risks to centralization risks and liquidity risks. For example, technical risks involve the integrity of the smart contracts which are very difficult to be written error-free, the vulnerability of the DeFi organizations susceptible to hacks, the scalability of the Ethereum facing an extremely high volume of DeFi transactions, and the substantial cost of Gas fees due to the process of a high volume of transactions [4]. With these hurdles in the development of decentralized finance, it does raise concerns with the practicality of Ethereum's decentralized finance in replacing the traditional model, but it has provided many useful insights due to its interesting properties and the promising applications from its decentralized infrastructure. Therefore, graph analytics of the Ethereum network regarding its decentralized finance sector would be helpful in the beginning of understanding such aspects and more.

The graph analytics of Ethereum blockchain network especially concerning the addresses of decentralized finance is significant in revealing the connectiveness and the dynamics of the decentralized finance network. It can also give useful information regarding decentralized finance on Ethereum, such as the trading culture, users' behaviours, transaction patterns, and emerging phenomena of the Ethereum blockchain network. As a result, more new insights can be acquired about Ethereum blockchain and they will help people to better understand the Ethereum and use Ethereum to make better decentralized applications or decentralized financial products suitable to the users' needs in the mainstream financial market.

3.1 Project Methodology

In order to virtualize and analyze the Ethereum decentralized finance network, charts, tables, distribution tables, and computational graph analytics will be used to analyze different networks on Ethereum, such as trace network, contract network, transaction network, and token network. Lee, Khan, Gupta, Ong, and Liu [5] used local and global graph properties such as in-tooutdegree correlation, centrality, reciprocity, assortativity, transitivity, and small-world characterization to study the Ethereum networks. One of the main focuses is the important behaviours and phenomena of the smart contracts, for example the smart contracts relied on other contracts to process a transaction and had generic codes that other contracts shared.

Chen, Zhang, Che,n, Zheng, and Lu also used graph analysis to investigate the entire Ethereum ERC20 token ecosystem and invented an algorithm based on crossed-graph analysis to discover the complicated relationship in the token systems [6]. These analyses found that over a quarter of all tokens were inactive, which entities were responsible to create many tokens using one account, and the strange phenomenon where one or group of trading users on the network traded with oneself or themselves.

These methods used by the past researchers will be crucial in finding certain properties and phenomenon on the Ethereum's DeFi networks. Additionally, this paper will use computational graph analytics as well to explore the centralities, such as the degree centrality, eigenvector centrality, Page Rank, and hyperlink-induced topic search, on Ethereum's DeFi networks. Hopefully more insights can be derived from this new research on decentralized finance.

The proposed data source is downloaded from the Google Cloud BigQuery and the computational analysis will be done with Python 3.9 with the help of mathematical libraries such as Matplotlib, NumPy, pandas, and igraph.

3.2 Contributions

To the best of my knowledge, I am the first to conduct a study on Ethereum Decentralized Finance blockchain networks investigating their interactions with trace network, contract network, token network, and transaction network constructed from the Ethereum blockchain data.

I study the Ethweum DeFi blockchain networks based on the network characteristics, local network properties, and global network properties. The measurements include vertices, edges, self-loops, and density, the vertex degree distribution, outdegree-to-in degree ratio of each vertex in the networks, degree centrality, eigenvector centrality, Page Rank, coreness, reciprocity, assortativity, strongly and weakly connected components, articulation point, adhesion, cohesion, average transitivity, and cluster co-efficient.

These analyses are useful in revealing the functionalities and behaviours of smart contracts and tokens in the decentralized finance area of the Ethereum blockchain, as well as the general characteristics of decentralized finance networks.

The code used to generate the data for the figure will be put online.

4. Related Work

The blockchain networks have been explored in many aspects since the creation of Bitcoin. Blockchain's one of new networks, Ethereum, although its development is still in progress, also has attracted public attention for its applications and functionalities, due to its transparency of the open-sourced blockchain network and the revolutionized concepts of smart contract and token.

Analysis of Ethereum blockchain networks: Smart contract has been the main focus in academia lately in the field of blockchain. Some research has explored the huge similarity of smart contracts and the reasons as to why [7, 8, 9]. While smart contracts have shared similar structures and functionalities on Ethereum, studies have shown that a small subset of contracts on Ethereum takes up most of the transaction volumes as the targets of the transactions are towards or from those contracts, and interestingly, it has been observed that most of these smart contracts are financially related [9, 10]. Moreover, the security issues of smart contracts on the Ethereum blockchain have been examined in recent research, as there were impactful attacks on the blockchain and the immutability of smart contracts is a pioneering feature but it in fact poses technical risks, therefore studies were conducted to find the ways the smart contracts can be maliciously exploited and the security measures could be taken to prevent such exploitations [11, 12].

Beside smart contracts, token as one of Ethereum's feature is also studied. Researchers have measured the complete network of ERC20-compatible tokens on Ethereum in order to study the overall characteristics of the network and the general behaviours of the network's nodes [12, 14, 15].

Measurement of Ethereum blockchain networks: Some papers have explored and measured Ethereum's blockchain network on different layers. One comprehensive study has taken into considerations of all aspects of the blockchain network across all the layers in order to study the interaction patterns among addresses, the similarities, and difference of those network layers [5]. These measurements which the paper used have provided me great help on how to measure networks fully and systematically. Also, a more specified study from the paper above has focused only on one the layers of Ethereum networks, the token layer, to examine the characteristics of tokens, their creators, their holders, purposes of tokens, and other phenomena in the token network including the case of the fake transactions in the network.

Analysis of Ethereum Decentralized Finance networks: Very few papers have focused their analysis on only the decentralized finance area of Ethereum, however, one paper has studied the key non-financial risks in decentralized finance on Ethereum blockchain in order to improve the risk management practices in this area [16].

Measurement of Ethereum Decentralized Finance networks: Currently, no papers have measured Ethereum networks in details regarding the decentralized finance area on Ethereum. However, as the decentralized finance networks use smart contracts and tokens extensively, some of the research, mentioned above, have analysed the Ethereum's smart contracts and tokens, therefore they can give insights to the nature of smart contracts and tokens and how they may adapt in the new area of decentralized finance.

5. Research Setup

In order to gain an understanding of the Decentralized Finance network on Ethereum, I retrieve all records related to the 32 decentralized financial applications on the network and create four networks such as trace net, transaction net, contract net, and token net, where vertices are Ethereum address accounts and the edges are the transaction interactions. In these networks, there are three major types of interactions which are Contract-to-User, User-to-Contract, and Contract-to-Contract. The rest of transaction interactions, whose to address or from address is Null address, are the records of creating smart contracts.

5.1 Data Extraction

I select the top 32 decentralized financial applications on the DeFi Leaderboard on Etherscan.io on the date of 16th of December as main research subjects and I collect on their contract addresses on Etherscan.io. The number of the total contract addresses is 3560.

The 32 decentralized financial applications are arranged by their rankings as follows: WBTC, Maker, Compound, Uniswap, Aave, Curve.fi, Balancer, Synthetix, Ren, Yearn.Finance, Bancor, InstaDApp, Nexus Mutal, Kyber, dYdX, mStable, imBTC, TokenSets, 0x Stakin,g Nuo Network, DeversiFi, Erasure, Opyn, DDEX, pTokens BTC, Loopring, Flexacoin, MCDEX, ForTube, Augur, Veil, Robo-Advisor Yield.

Then I use Google Cloud BigQuery to curate these addresses' records from the Ethereum blockchain, of which the source is crypto_ethereum dataset under the Google Cloud bigquerypublic-data repository from their times of creation to 2020-12-21 03:43:24 UTC. Five datasets are downloaded from the BigQuery server, as in Table 1.

For the year-by-year analyses and the ERC compatibility analysis, I use the same source as above but I take data from their times of creation to 2021-03-16 09:16:32 UTC.

Table 1: **Downloaded Ethereum data from BigQuery: From creation to 2020-12-21**

Table	Approximate Size of Dataset
contracts	**272 MB**
transactions	**5.5 GB**
traces	**49.8 GB**
token transfers	**4.02 GB**

5.2 Network Creation

As I intend to understand the workings of decentralized finance area of Ethereum, the tables of traces, transactions, and token transfers from the downloaded Ethereum data are used mainly. The traces table has the most comprehensive records of the 32 DeFi applications' addresses in the Ethereum network and the rest of tables has specific types of records for the specific DeFi applications' addresses. The from addresses and the to addresses of each table are vertices of the network and each record containing valid from address and to address represents an edge of the network.

Trace network is created from the traces table with all the 32 DeFi applications' smart contract addresses and all their corresponding users as vertices of the network and with successful trace's records with valid from address and to address as edges of the network. The edges of this network include all types of interactions among contracts and users, e.g. internal transactions on Ethereum which are by-products of Eth or token transfer on contract execution.

Contract network is created from the traces table with only the 32 DeFi applications' smart contract addresses as vertices of the network and with successful contract's records with valid from address and to address as edges of the network. The edges of this network include the contract related interactions such as the creation of contracts, the suicides of contracts, and the calling of contract. The smart contracts are created with or without ERC standards.

Transaction network is created from the transaction table with all the 32 DeFi applications' smart contract addresses and all their corresponding users as vertices of the network and with successful transaction's records with valid from address and to address as edges of the network. The edges of this network include the transaction of ether and the creation of contracts.

Token network is created from the token transfers table with all the 32 DeFi applications' smart contract addresses and all their corresponding users as vertices of the network and with successful token transfers' records with valid from address and to address as edges of the network. The edges of this network include the token transfer and token contracts facilitation of other token contracts' transfers.

5.3 Setup Equipment

The codes used to query Ethereum blockchain data on Google Cloud BigQuery are based on SQL. Data cleaning and data transformation of the blockchain data are achieved with Python codes. As for data analysis, python-igraph is used on Python 3.9 and Jupyter Notebook. The analyses are performed on an Intel Core i7 CPU 2.3 GHz.

6. Research Findings

6.1 Network Characteristics: Vertices and Edges, Self-loops and Density

Table 2: ***Network characteristics of decentralized finance networks on Ethereum***

			Directed Graph			Undirected Graph	
	Vertices	Edges	Self-loops (% of Edges)	Density	Edges	Self-loops (% of Edges)	Density
Token Network	945740	**30780402**	161 (0.00052%)	**3.44E-05**	**2365226**	10 (0.00042%)	**5.29E-06**
Transaction Network	2853444	35913284	747 (0.00208%)	**4.41E-06**	35902972	13(0.00004%)	**8.82E-06**
Contract Network	3109	84675173	**2741249 (3.23737%)**	**8.76303125**	**36915627**	**32(0.00009%)**	**7.640794**
Trace Network	3324463	365414737	2741996 (0.750379%)	**3.31E-05**	199140748	45(0.00002%)	**3.60E-05**

Each of the four networks has two variations of graph presentation, which are the directed graph and the undirected graph. In the directed graph, multiple edges between a pair of vertices are kept as to show that the pair of addresses has more than one interaction. In the undirected graph, multiple edges between a pair of vertices are only counted as one edge as to show that the pair of addresses has a connection. In table 2, the edges of the directed graphs are more than the edges of the undirected graph.

Self-loops are also counted in both the directed graph and undirected graph of each network. Self-loops are created when from address and to address of a record are of the same address and this can indicate that the address made a transaction with itself or call itself to update its contract's state.

In Table 2, self-loops are counted as well as its percentage over all the edges. The selfloops of the directed graphs are more than those of the respective undirected graph as directed graphs naturally have more edges than the respective undirected graphs. It can be observed that contract network's directed graph has the higher self-loops percentage than the rest of the networks' directed graphs and this means that these contracts are calling on themselves repeatedly to perform functionalities the contracts themselves process.

Density of a graph, undirected or directed, is calculated as the ratio of the number of edges and the number of possible edges. While calculating the densities of both directed graphs and undirected graphs, self-loops are not taken into consideration. In common sense, as a network's directed graph has more edges than the undirected graph, the directed graph is generally denser than the undirected graph. Therefore, both token network and contract network's directed of are denser than their undirected graphs, as the number of edges in their directed graphs are thirteen times and twice those in their undirected graphs respectively. It is worth noting that the token network's directed graph is six times denser than its undirected graph, which means that the token network's directed graph has many multiple edges between vertices denoting a major number of activities running in the token network of the decentralized finance. This can be interpreted as tokens are transferred vehemently between addresses in decentralized finance as token exchanges are a prominent part of decentralized finance applications on Ethereum and the major decentralized applications like Maker and Wrapped Bitcoin offer tokens against the U.S. dollar and Bitcoin respectively, which make them very popular on Ethereum DeFi.

To the contrary in the trace network and transaction network, their undirected graphs are denser than their directed graphs. Looking at the number of edges of these graphs, it can be seen that the transaction network's directed graph only has 0.02% more edges than its undirected graph, whereas the trace network's directed graph has almost double the edges of its undirected graph. This is because trace network and transaction network have very few bidirectional edges between vertices in their graphs. This can be interpreted as the decentralized finance applications have made a lot of transactions with their users, but the interactions are mostly unilateral.

Moreover, it is notable that the contract network has the highest density in both directed and undirected graphs. This indicates that the contract network is the most tightly connected among the four networks.

6.2 Local Network Properties

6.2.1 Vertex Degree Distribution

Table 3: ***Degree distribution of decentralized finance networks***

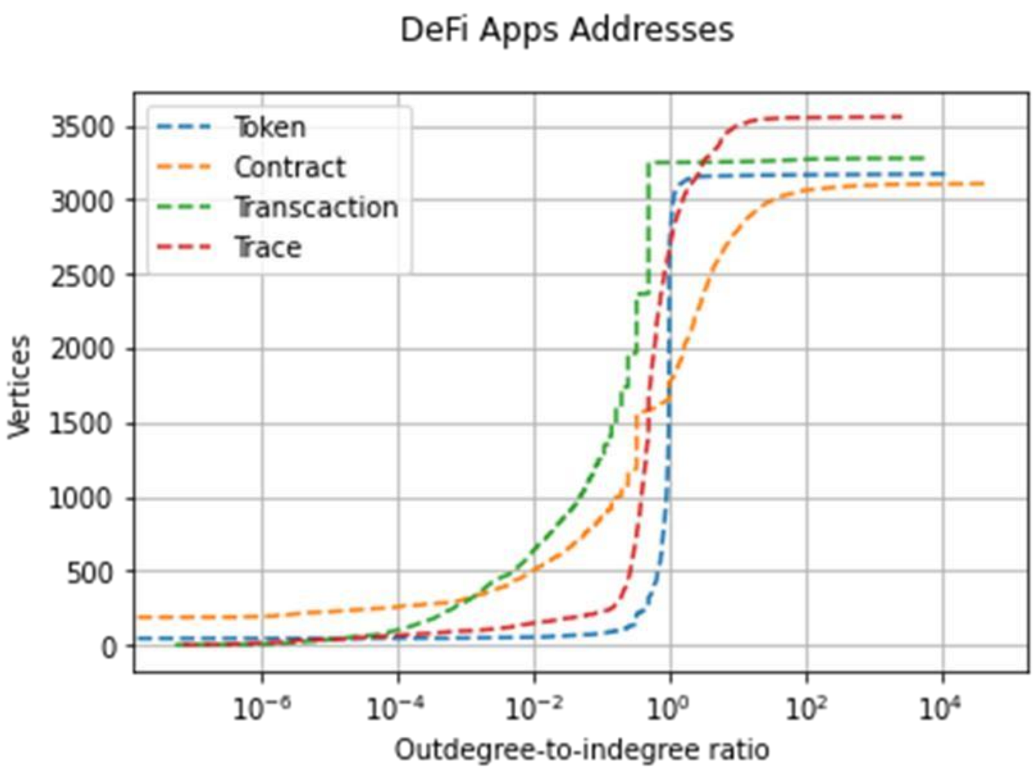
	Mean	Standard Deviation
Token Network	65.09	13537.70
Transaction Network	25.17	11051.73
Contract Network	54471.00	385521.90
Trace Network	219.83	62782.71

In Table 3, the degree distributions of all the four networks' directed graphs are not normal distributed, as their standard deviations are larger than their corresponding means. The standard deviations of token network, transaction work, and trace network are at least 200 times larger than the means of these networks, while, in the contract network, its standard deviation is 7 times larger than its mean. This means that in the token, transaction, and trace networks, a few of vertices have extremely high vertex degrees than most vertices. These vertices with high vertex degrees are the smart contract addresses of the decentralized finance applications.

As for the contract network, the extreme difference of vertex degree is lessened across the vertices because only the contract addresses are recorded in this network and these addresses call on each other frequently to use their functionalities or to send information.

6.2.2 Outdegree-to-Indegree Ratio

Figure 1: ***DeFi Apps Addresses' Outdegree-to-Indegree Ratio***



In Figure 1, I select the 32 decentralized finance contracts' addresses and compare the outdegree-to-indegree ratio of each vertex. Some outdegrees or indegrees of the vertices are zeros, and so it results in unbalanced results. Therefore, normalization of the outdegree and indegree is achieved by adding one to each of the degrees to solve the ratio's zero problem. As a result, the outdegree-to-indegree ratio is calculated by this formula: Ratio = (1+ outdegree) / (1+ indegree).

I find that about 15% of trace network's and 3% of token network's vertices, and about half of contract network's vertices have higher outdegrees than indegrees, however less than 1% of transaction network's vertices do. Moreover, around 30% of token network's and half of contract network's vertices, and around 75% of trace network's vertices have higher indegrees than outdegree, whereas almost all of the transaction network's vertices (>99%) do.

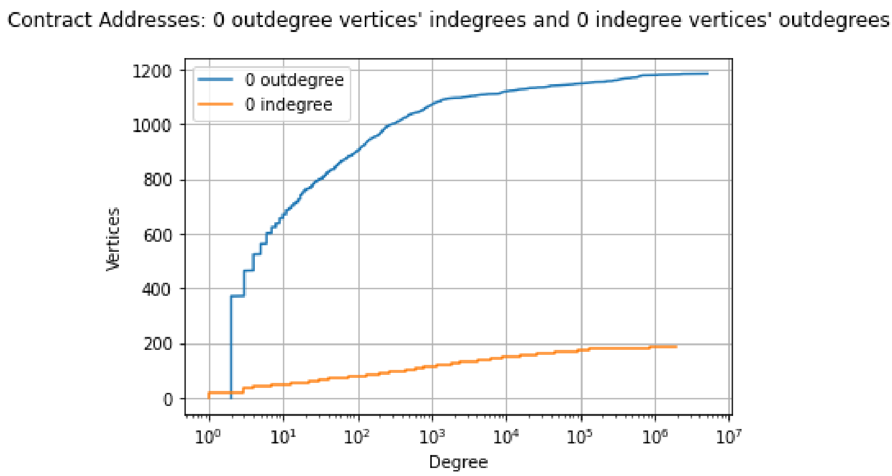
I also find that more than 60% of token network's vertices and less than 1% of both contract network's vertices and trace network's vertices have similar number of outdegrees and indegrees, while no transaction network's vertices (<1%) do.

Based on these observations, it can be seen that the four networks do not have balanced interactions among addresses and these networks follows the power-law distribution where, a certain small subset of addresses has some of the highest outdegree-to-indegree ratios or some of the highest indegree-to-outdegree ratios, while most addresses are the targets of recipient more than they are the targets of sender.

6.2.3 0-outdegree Vertices' Indegrees and 0-indegree Vertices' Outdegrees

Of all the DeFi applications' contract addresses in the four networks, I focus on those particular contract addresses with 0-outdegree and 0-indegree and look at their corresponding indegrees and outdegrees.

Figure 2: ***0-outdegree vertices' indegrees and 0-indegree vertices' outdegrees in Contract Network***



In Figure 2, in the contract network, over a third of the 3560 contract addresses/vertices have 0 outdegrees and over 5 percent of all addresses/vertices have 0 indegrees. Half of these 0outdegree vertices have less than 10 indegrees and the other half mostly have indegrees between 10 to 10000. A very small number of 0-outdegree vertices have more than 10000 indegrees. As for the 0-indegree vertices, half these vertices have more than 10000 outdegrees and the other half has less than such outdegrees.

This indicates that a third of all contract addresses is responsible only for being called on by other contracts to execute some functions they have. A very small number of such addresses, around 5 percent, has been called on more than 100000 times.

*Figure 3: **0-outdegree vertices' indegrees and 0-indegree vertices' outdegrees in Token Network***

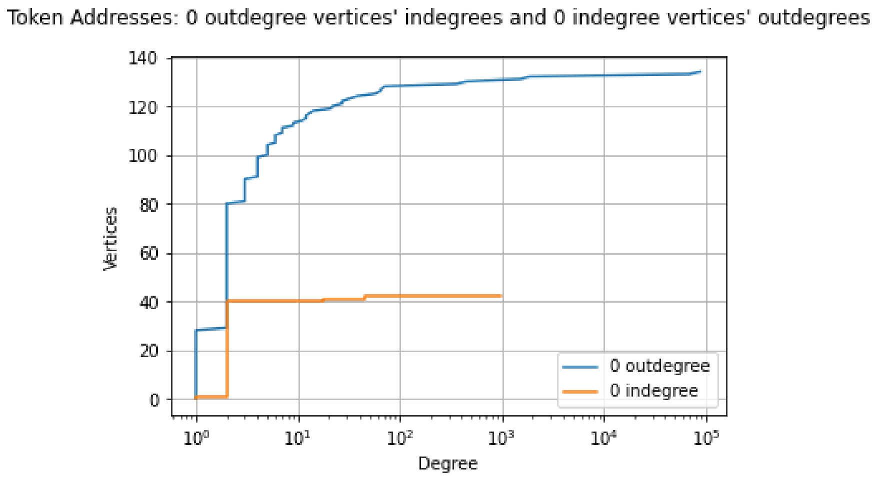
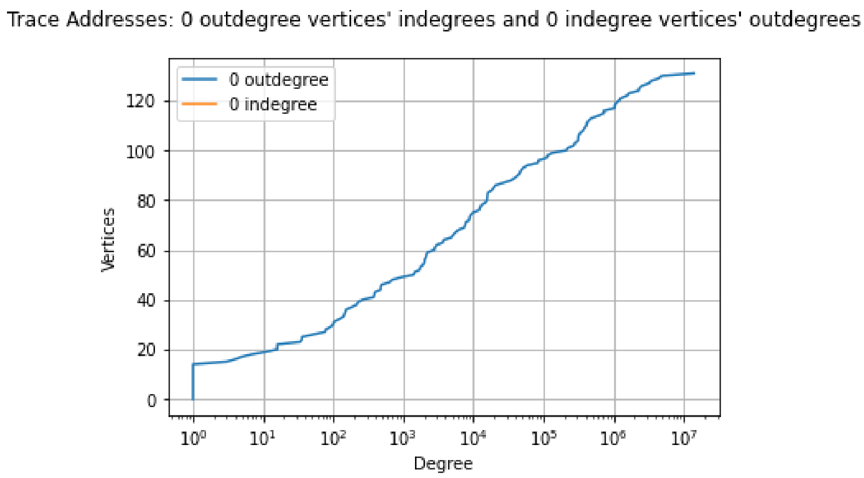


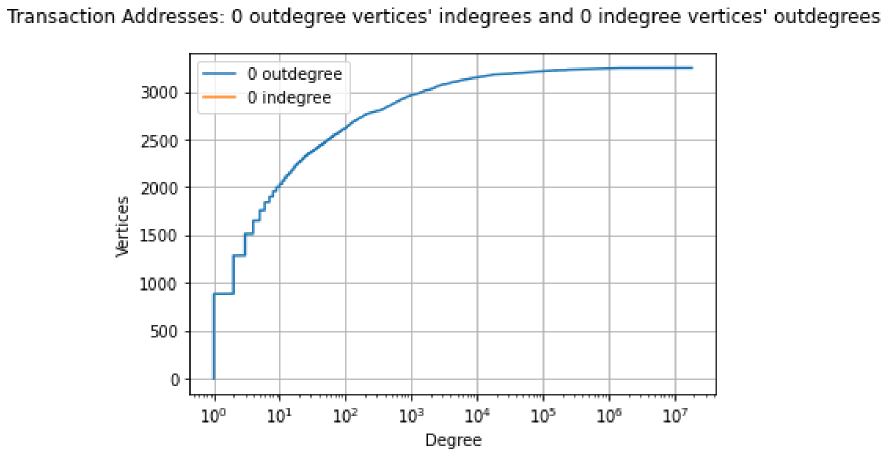
*Figure 4: **0-outdegree vertices' indegrees and 0-indegree vertices' outdegrees in Trace Network***



In Figure 3 and 4, in the token network and trace network, over three percent of the 3560 contract addresses/vertices have 0 outdegrees, but less than 10 percent of those 0-outdegrees in the token network have indegrees from 10 to 10 to the power of 5 and the overall 0-outdegree vertices in the trace network have an exponential distribution of indegrees from 1 to 10 to the power of 7. As for the 0-indegree vertices, there is no 0-indegree vertices in the trace network and there is around 1 percent 0-indegree vertices of the 3560 contract addresses that have outdegrees between 1 to 1000 in the token network.

This indicates that a very small number of contract addresses, i.e. more than 120, in the token network is only responsible for receiving tokens from other contracts and users, of which only a few of them receive over 10e5 times. As for the trace network, all of the contract addresses have been the target recipients of transactions more than or once. Interestingly, these 0-outdegree vertices in the trace network are the recipients of transactions in an exponentially even manner, indicated by the almost straight line in the Figure 4, which implied these contracts never initiate the transactions themselves and only act as recipients in transactions. This could be interpreted as common practice for contracts to receive only instructions, however the sample of these 0outdegree vertices are only of around 120 and it could be insignificant to put focus on.

*Figure 5: **0-outdegree vertices' indegrees and 0-indegree vertices' outdegrees in Transaction Network***



In Figure 5, in the transaction network, all of the 3560 contract addresses/vertices have 0 outdegrees and over forty percent of these 0-outdegree addresses have more than 10 indegrees and the other sixty percent have less than 10 indegrees. Also, there is no 0-indegree vertices in transaction network. It can be observed that all the contract addresses are used to receive ether from users and contracts or are being called to create the contracts themselves.

In Figure 2, Figure 3, Figure 4, and Figure 5, the four networks share the similar characteristics where a minority of 3560 contract addresses have zero outdegrees and a few of these addresses are the target recipients of most transaction records. Moreover, it can be seen that all these four networks have significantly more 0-outdegree vertices than 0-indegree vertices. As a result, I estimate that the DeFi applications' smart contracts are used mostly as recipients rather than as active senders on Ethereum activities and that could be due to the fact that there are Ethereum users use DeFi applications for their services for the first time as a an experiment and do not require a second interaction initiated by the DeFi application and hence there is no transaction record thereafter.

6.2.4 Centrality Measures

Centrality measures of all vertices and only the decentralized finance contract addresses in undirected graphs of the four networks are presented below. There are two types of graphs for each centrality measure; the first graph targets all addresses including contracts and users in the networks, and the second graph targets only the 3560 DeFi contract addresses in the networks.

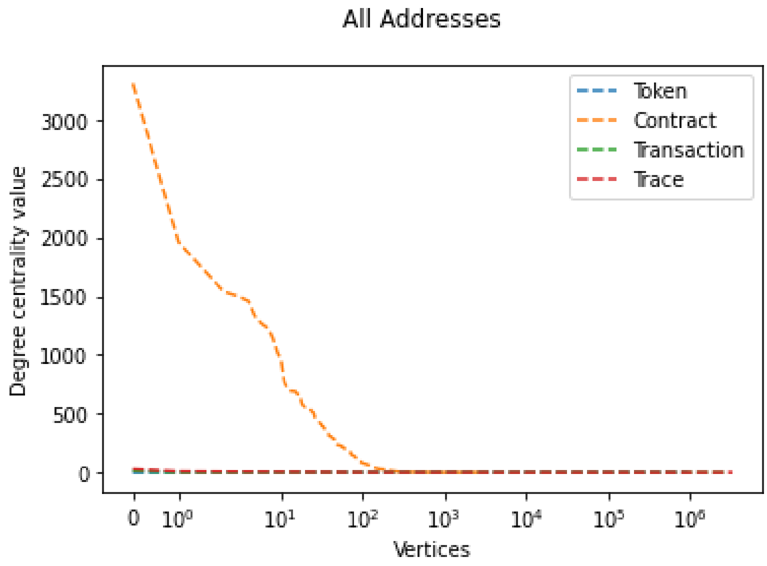
I measure these vertices in terms of their degree centrality, eigenvector centrality, and Page Rank, in order to score and rank them according to their connectedness and importance.

Degree centrality is the measure of the number of vertices adjacent to a vertex normalized by the total number of vertices in the undirected graph of a network, in order to show an aspect of the vertex's importance. For all addresses in the graph, the degree centrality is calculated by this formula, Degree Centrality = Number of vertices adjacent / Total number of vertices. As for only the DeFi contract addresses in the graph, their degree centrality values are calculated by this formula, Degree Centrality = Number of vertices adjacent / (Total number of only the DeFi contract addresses/vertices -1). This may be an unconventional approach to look at the relationship of DeFi Apps addresses and their adjacent addresses, but it shows the number of connections a DeFi Apps address has in the networks, while normalized by the total number of DeFi Apps addresses.

Eigenvector centrality is the measure of the degree of the vertex and its neighbouring vertices' degree in the directed graph of a network, to show the importance of a vertex while considering the importance of its adjacent vertices as well. For all addresses and only the DeFi contract addresses in the graph, the eigenvector centrality is calculated by this formula, $x_i = \sum_{j \in N(i)} (x_j) = \sum_j (A_{ij} x_j)$.

The used Page Rank is a variant measurement of the eigenvector centrality of the vertices in the undirected graph of a network, to show that the importance of a vertex while considering the number of in-bound adjacent vertices, the importance of these vertices, and the number of connections these vertices have. For all addresses and only the DeFi contract addresses in the graph, the Google PageRank value is calculated by this formula, $PR(A) = (1\text{-}d) + d\ (PR(T_1)/C(T_1) + ... + PR(T_n)/C(T_n))$, where $PR(T_i)$ is the Page Rank of pages T_i which link to page A, $C(T_i)$ is the number of outbounds links on a given T_i page, and d is the damping factor in the range 0 and 1.

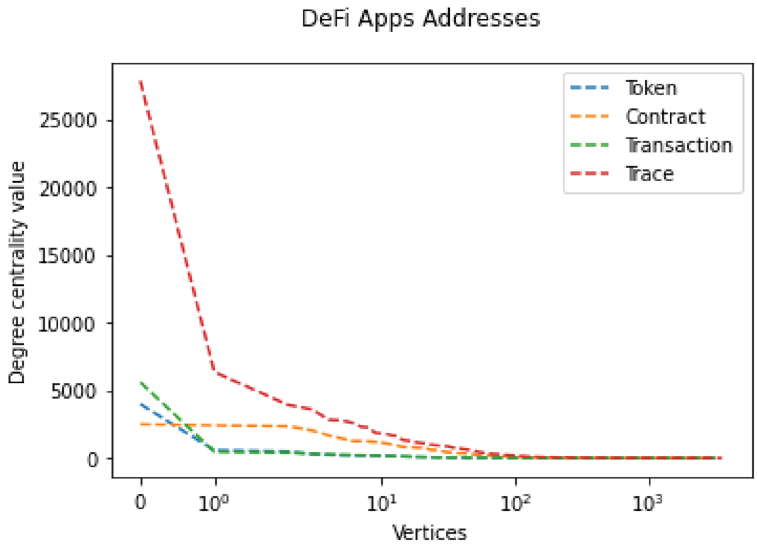
*Figure 6: **All addresses' degree centrality values***



In Figure 6, while the degree centrality values for all vertices in the token, transaction, and trace networks are close to zero, there are 100 vertices in the contract network which have degree centrality values ranging from 0 to over 3000.

This can be interpreted as the contract network's vertices are the closest to each other comparing to other networks' vertices and there are around 100 of these vertices that are very important in terms of influence and communications in the contract network. As in the rest of the networks, all the addresses appear to be not central at all due to the division by the enormous number of the addresses in the formula.

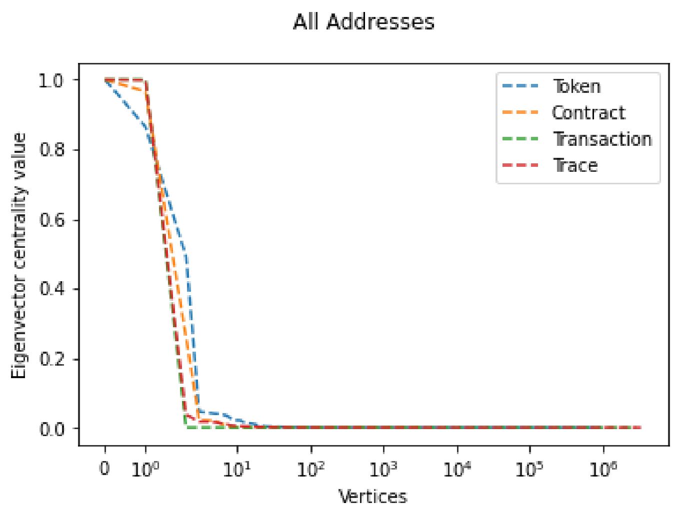
*Figure 7: **DeFi Apps addresses' degree centrality values***



In Figure 7, the adjusted formula shows the number of connections each vertex has in each network normalized by the total number of the DeFi Apps addresses minus 1, as the degree centrality. Therefore, as it can be observed, among the 3560 DeFi Apps addresses in the four networks, 100 vertices in the trace network have higher degree centrality values than the vertices in other networks. Most DeFi Apps addresses/vertices in the token and transaction networks have degree centrality values close to 0, in the range of 0 and 1000, except one particular vertex of each of these two networks have values around 5000. As for the contract network, 100 DeFi Apps addresses/vertices have degree centrality values from 0 to 3000.

Therefore, it can be interpreted as, among all the 3560 DeFi Apps addresses, the trace network has some vertices that are the most central comparing to other networks' vertices. The contract network has some vertices that are similarly central whereas the token and transaction networks have most vertices that are not central at all.

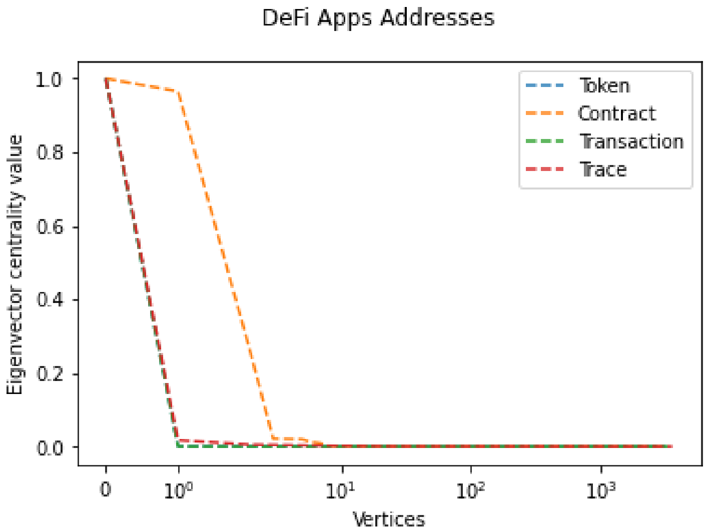
Looking at the Figure 6 and Figure 7, we can see that although Figure 6 shows that the contract network has more important vertices than the rest of the networks, Figure 7 reveals that, focusing on only the DeFi Apps addresses of these four networks, the trace network has more important DeFi Apps addresses than contract network does, while the token and transaction networks have very few important DeFi Apps addresses. *Figure 8: **All Addresses' eigenvector centrality values***



In Figure 8, all the four networks share similar number of vertices with similar eigenvector centrality values. Only a few vertices, i.e., less than ten, in these four networks have high eigenvector centrality values while the rest of the vertices have very low values.

This means that the four networks have most vertices whose adjacent vertices are very poorly connected. As a result, we can see that the four networks are weakly connected.

*Figure 9: **DeFi Apps addresses' eigenvector centrality values***



In Figure 9, regarding the 3560 DeFi Apps addresses in the four networks, most of the vertices have very low eigenvector centrality values except for a small number of vertices in the contract network. These vertices in the contract network have relatively higher eigenvector centrality values comparing to the vertices in the rest of the networks.

It is reasonable to see that some DeFi Apps addresses in the contract network have higher eigenvector centrality values because the DeFi Apps smart contract addresses trade many times with other smart contract addresses in a small tight knitted circle and so a few of DeFi Apps smart contract addresses are rated more influential as their adjacent contracts addresses are well connected as well.

*Figure 10: **All addresses' page rank value***

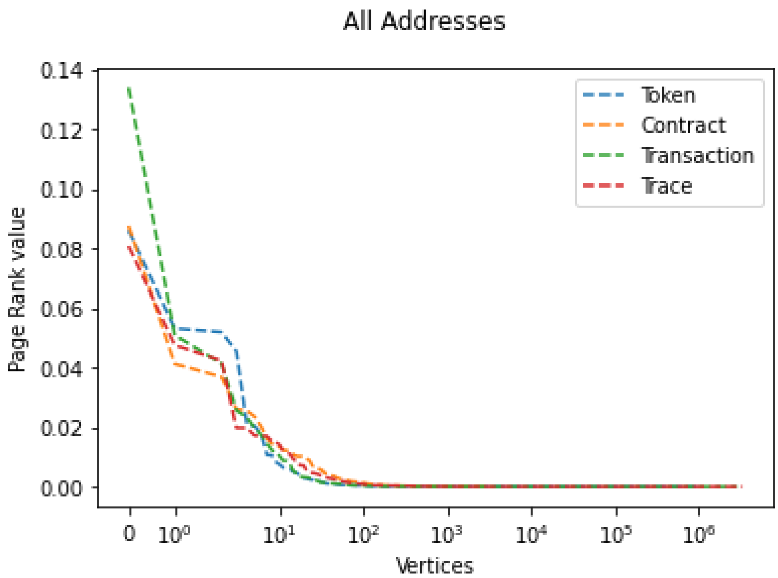


*Figure 11: **DeFi Apps addresses' page rank value***

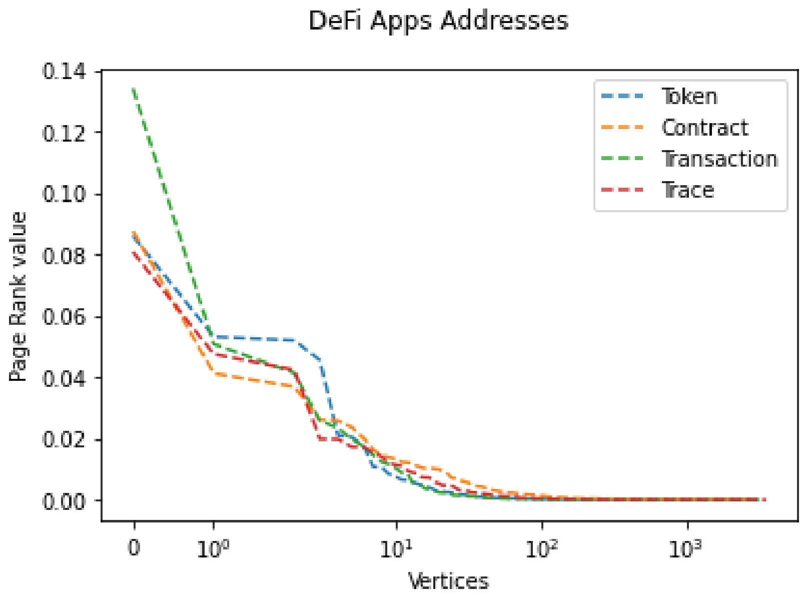


Figure 10 and Figure 11 together further show that all the vertices beside the DeFi Apps addresses have almost zero Page Rank values. In Figure 11, it shows that the DeFi Apps addresses in the four networks share similar Page Rank values.

It is notable to mention that the Page Rank values of these DeFi Apps addresses are very small overall. This can be interpreted as the four networks have very low-quality vertices connected to very low-quality vertices in general.

Since betweenness and closeness centralities are expensive to calculate over these huge networks, I measure the centralities of the networks in aspects of degree centrality, eigenvector centrality, and Page Rank.

6.3 Global Network Properties

6.3.1 Reciprocity and Assortativity

Reciprocity is the measure of the likelihood of vertices to be mutually linked in a directed network. Its definition is based on the adjacency matrix notation and its formula is sum(i, j, (A.*A')ij) / sum(i, j, Aij), where A.*A' is the element-wise product of matrix A and its transpose. If the network's reciprocity is equal to 1, it means that the network is a purely bidirectional network. If the network's reciprocity is equal to 0, it means that the network is a purely bidirectional network.

Assortativity is the measure of the likelihood of vertices of similar nature, e.g. vertices with high degrees, linked with each other in an undirected network. The assortativity value is based on the vertex degree minus 1 as the vertex values. This measurement is defined as sum(jk(e(j,k)-qout(j)qin(k)), j, k) / sigma(qin) / sigma(qout), where e(j,k) is the fraction of edges connecting vertices of type j and k, qout(i) is sum(e(i,j), j), qin(i) is sum(e(j,i), j), and sigma(qout) and sigma(qin) are the standard deviations of qout and qin respectively. The range for assortativity is from -1 to 1. A network is assortative if the value is closer to 1, and this means that the vertices with similar degrees are more likely to link together. While a network is disassortative if the value is close to -1, and this means that the vertices with different degrees are more likely to link together.

*Table 3**: Reciprocity and assortativity of the four networks***

Network	Reciprocity	Assortativity
Contract Network	0.128	0.072
Token Network	0.789	-0.300
Transaction Network	1.04E-05	-0.375
Trace Network	0.115	-0.061

In Table 3, the reciprocity value of token network is significantly higher than the rest of the networks and the reciprocity value of contract network is the second highest among the four networks. It is surprising to see that the token network is significantly higher in reciprocity than the contract network, as the smart contracts rely on each other to process a transaction very often and the reciprocity value of contract network is usually significantly higher than the rest of the networks in the overall Ethereum blockchain network. It still ascertains the concept that the smart contracts rely on each other so that bidirectional edges are created in the contract network, as these smart contracts will return information to the smart contracts which call on them. It is interesting to see that the token network is the highest in reciprocity and this shows that in the decentralized finance networks of Ethereum, the tokens are traded back and forth vehemently between the sending addresses and the receiving addresses. This can be interpreted as tokens are the most traded commodity in the Decentralized Finance area of Ethereum and the DeFi users or smart contract addresses trade the tokens with a specific set of users or addresses frequently.

In Table 3, the assortativity value of contract network is the most positive among the four networks, while the rest of the network all have negative assortativity values. For the contract network, assortativitity is more prevalent and this shows that smart contracts in DeFi contract network trade with the contracts with similar degrees. This can be further interpreted as that the DeFi Apps' smart contracts call the same kind or the similar kind of contracts or themselves to execute transactions. As the DeFi's smart contracts serve some specific functions in the DeFi networks, they may not have to call on generic smart contracts often to execute a process, unlike the general smart contracts on Ethereum. Therefore, the DeFi smart contracts relate to their kind more because of their specific purposes in the DeFi area. As for the rest of the networks, they all tend to be disassortative that entities within the networks have interactions with entities of different natures, for instance the decentralized financial applications act as central hubs using their own contracts for other users and smart contracts to interact with.

6.3.2 Strongly and Weakly Connected Components

A graph is connected if every pair of vertices in the graph is connected. Considering directed graphs, a component is strongly connected if all its vertices have bidirectional paths reaching each other and a component is weakly connected if a connected graph is produced when its directed edges are replaced with undirected edges.

*Table 4: **Characterization of blockchain networks' connected components***

Directed Networks	Number of strongly connected components	Largest strongly connected component (vertices, edges)	Number of weakly connected components	Largest weakly connected component (vertices, edges)
Contract Network	**2851**	**100**, **4653190**	**4**	**3095, 84536769**
Token Network	483854	461861, 29592664	**26**	**945682, 30780338**
Transaction Network	**2853313**	**78**, **210**	**18**	**2853403, 35901485**
Trace Network	2734423	589972, 159436778	**1**	**3324463, 365414737**

In Table 4, for each of the four directed networks, the number of weakly connected components are significantly fewer than the number of strongly connected components and the largest weakly connected component has extraordinarily more vertices and edges than the largest strongly connected component. This shows that weakly connected components have connected the strongly connected components into a small number of components, and the relatively higher number of strongly connected components indicates that the networks have few bidirectional edges among the vertices.

Referring to the directed networks' vertices and edges in Table 2, I compare them to the largest weakly connected components of each of the undirected networks and it can be observed that each of the largest weakly connected component cover most of the portion of the respective networks.

In Table 4, the number of contract network's strongly connected components is the least among the four networks but its largest strongly connected component has edges 46000 times more than its vertices. This indicates a very strong connectivity for the smart contracts/ vertices in the contract network. Moreover, it is more interesting to note that the number of strongly connected components in transaction network is very close to the directed transaction network's number of vertices and its largest strongly connected component has very few vertices and edges. Therefore, it indicates a very weak connectivity for the vertices in the transaction network.

6.3.3 Articulation Points, Ahesion, Cohesion

Articulation points are important vertices to the network, and when any of them is removed, the number of connected components in the network is increased.

Adhesion value of a network is the minimum number of edges required to be removed in order to obtain a subnetwork which is not strongly connected and it is considered as the overall edge connectivity of a network.

Cohesion value of a network is the minimum number of vertices required to be removed in order to obtain two separate components in a network ad it is considered as the overall vertex connectivity of a network.

*Table 5: **Articulation points, adhesion, and cohesion of the four networks***

Network	Number of articulation points	Ahesion	Cohesion
Contract Network	68	0	0
Token Network	1312	0	0
Transaction Network	1190	0	0
Trace Network	872	0	0

In Table 5, comparing each network's number of articulation points to the total vertices in Table 2, it can be observed that the contract network has the highest ratio of the number of articulation points to the total vertices, i.e. 2.19%, and the rest of the three network have similarly low ratios, from 0.14% to 0.03%. The contract network has proportionately more articulation points than the other three networks' and this might suggest that the contract network has more single vulnerable points whose failure can split the network into 2 or more components comparing to the other three networks, in the DeFi area of Ethereum. However, the number of sampled smart contract addresses from the DeFi networks is only 3109 which is relatively and significantly smaller than any of the other networks, this should be taken into consideration in the connectivity of the contract network. As for the other three networks, they have proportionately smaller articulation points and this suggests that they are more connected as they have fewer vulnerable points.

We can also observe that all the adhesion and cohesion values of all networks are zeros in Table 5. This means that for all four of the networks, each of them is already a network of more than or equal to two disconnected components, in terms of edges and vertices constructing the network. It could be interpreted as in the decentralized finance area of Ethereum, there provides many different services to meet many needs and so, there exists certain sub-categories of the decentralized finance that serve a very particular kind of people who do not use other services from the decentralized finance atmosphere. As a result, this assumed situation produces disconnected components in all four networks of the DeFi area on Ethereum.

6.3.4 Average Transitivity and Cluster Coefficient

The average transitivity is the average of all the vertex transitivities of the network, where the transitivity is the probability that two neighbouring vertices of a vertex are connected in the network. As for the cluster coefficient, it can be considered as global transitivity of the network but it is different from the average transitivity, as it measures the probability that two neighbouring vertices of a vertex are connected in the network with the use of ratio of the triangles and connected triplets. For both average transitivity and cluster coefficient, undirected networks are used.

*Table 6: **Average transitivity and cluster coefficient of the four networks***

Network	Average transitivity	Cluster Coefficient
Contract Network	18.3175201%	0.3637821%
Token Network	6.0720913%	0.0009410%
Transaction Network	0.0091503%	0.0000007%
Trace Network	10.7084667%	0.0012651%

In Table 6, among the four networks, the contract network has the highest average transitivity and cluster coefficient and the transaction network has the least of both. Overall, the four networks have low average transitivities and cluster coefficients.

The networks’ low average transitivities and cluster coefficients indicate a lack of community structure. Because the DeFi network are non-social networks whose DeFi applications’ purpose is to provide services to users, it is understandable for the DeFi network to have high-degree vertices, i.e. the DeFi applications’ smart contracts, connected to many lowdegree vertices, i.e. the DeFi users. Therefore, the DeFi networks do not have tightly connected communities.

As for the contract network, the community structure is more connected and stronger than the rest of the networks because the density of the contract network is very high relatively to other networks, as shown in Table 2.

6.4 Year-over-year analyses

A year-over-year analyses using all the research methods from network characteristics, local network properties, and global network properties above are done on token network, transaction network, and contract network, from 2016 to March 2021. However, due to the enormous size of the trace network, the research equipment is not able to conduct a comprehensive analysis on the trace network, therefore, the trace network is omitted here in this section.

Also, as the completed analyses are many, I present only those with meaningful interpretations.

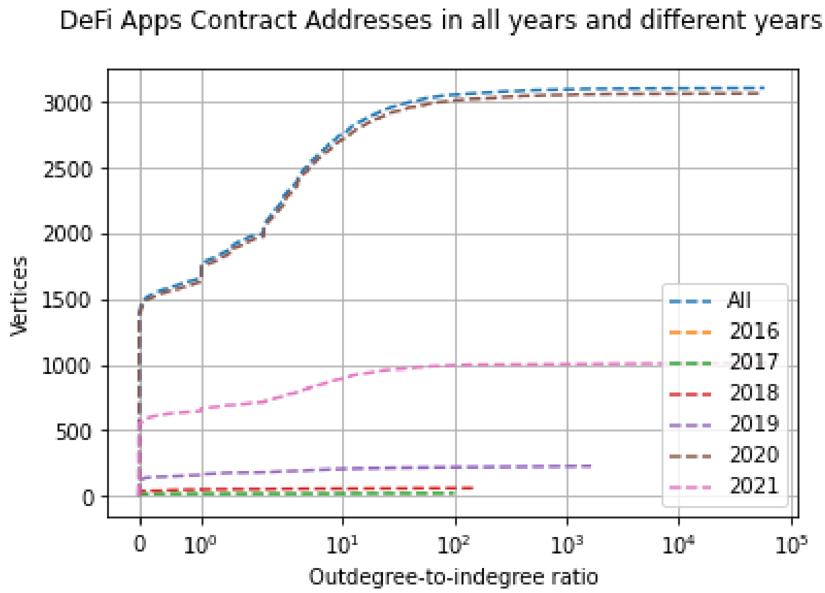
Table 7: ***Year-over year network characteristics on three directed networks***

Directed																							
Token Network	2016	2017	2018	2019	2020	2021	All	Transaction Network	2016	2017	2018	2019	2020	2021	All	Contract Network	2016	2017	2018	2019	2020	2021	All
Vertices	2	36838	36438	257393	728171	148291	1044223	Vertices	0	58994	259755	548914	2242544	1487869	345950	Vertices	0	20	61	227	3066	1012	3109
Edges	1	52288	907990	4139802	27132626	7016071	39248778	Edges	0	179133	1719458	4146775	27632111	10090905	921554	Edges	0	35595	1373440	17571671	67256359	6716817	92953883
Loops	0	0	6	4	171	6	187	Loops	0	0	1	672	77	20	17	Loops	0	4	12	202544	2637548	413873	3253981
Density	0.5	3.85E-05	0.000684	6.25E-05	5.12E-05	0.000319	3.60E-05	Density	nan	5.15E-05	2.55E-05	1.38E-05	5.49E-06	4.56E-06	7.70E-06	Density	nan	93.67105	375.2568	342.5143	7.156995	6.564956	9.619795

In Table 7, the number of edges increases year by year and in 2020, all three networks have the most edges comparing to the other years. This can be concluded that Ethereum Decentralized Finance is gaining popularity and more and more people are starting to use these DeFi services. Also, these three networks appear to have most activities in 2020, however, for 2021, the samples taken is only until March, 2021 may very well have the potential to have the most activities comparing to 2020.

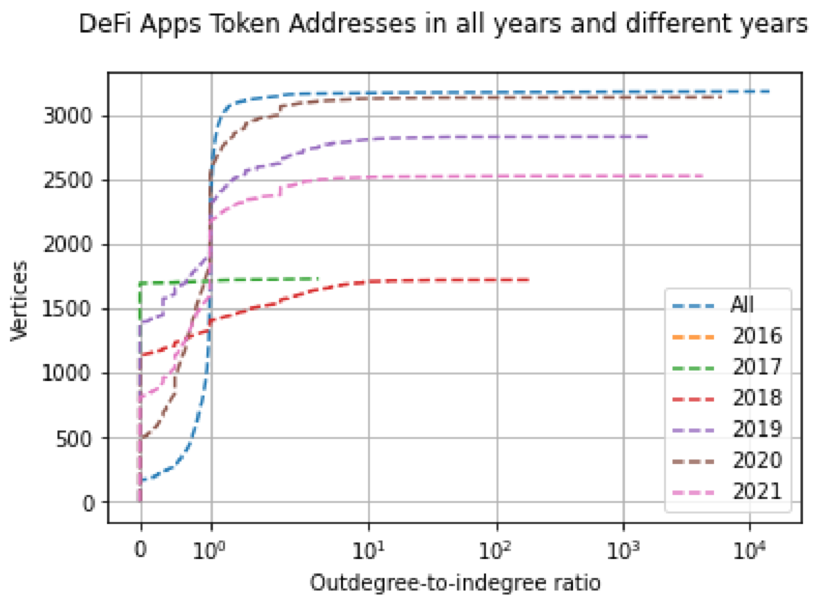
The year-over-year analyses of reciprocity, assortativity, strongly connected components, weakly connected components, articulation points, and number of cores on these three networks suggest the same observation as above.

Figure 12: ***Outdegree-to-indegree ratio of DeFi Apps Contract Addresses in all years and different years***



In Figure 12, looking at the outdegree-to-indegree ratio of the DeFi Apps contract addresses, the contract network in 2020 and 2021 has half of its vertices having outdegree-to-indegree ratios higher than 1, but around 2000 contract addresses in 2020 do not seem to have appeared again in the contract network in the first three months of 2021. This may be due to those contract addresses being outdated to the services and new smart contracts have been made to replace them, and during the process, these outdated contract addresses are usually terminated in suicide calls. But there is a possibility that they have yet to make calls.

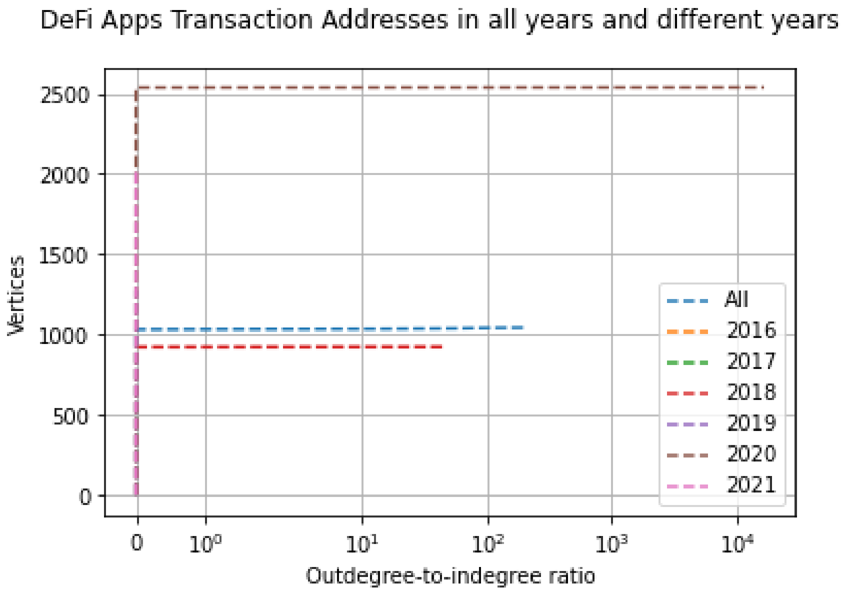
Figure 13: ***Outdegree-to-indegree ratio of DeFi Apps Token Addresses in all years and different years***



In Figure 13, looking at the outdegree-to-indegree ratio of the DeFi Apps token addresses, the token network in different years all have more vertices with outdegree-to-indegree ratio less than

1 and larger than 0 and the token network in 2019, 2020, and 2021 have similar number of vertices having the same outdegree-to-indegree ratios. Therefore, it can be observed that there is this same number of token addresses being active from 2019 to 2021, where they are mostly being called on during the different years.

*Figure 14: **Outdegree-to-indegree ratio of DeFi Apps Transaction Addresses in all years and different years***



In Figure 14, looking at the outdegree-to-indegree ratio of the DeFi Apps token addresses, the transaction network in different years share the similar phenomenon where there exist very few outlier vertices which have the highest outdegree-to-indegree ratios while most of the vertices have 0 outdegree-to-indegree ratios.

*Figure 15: **Degree centrality values of DeFi Apps Contract Addresses in all years and different years***

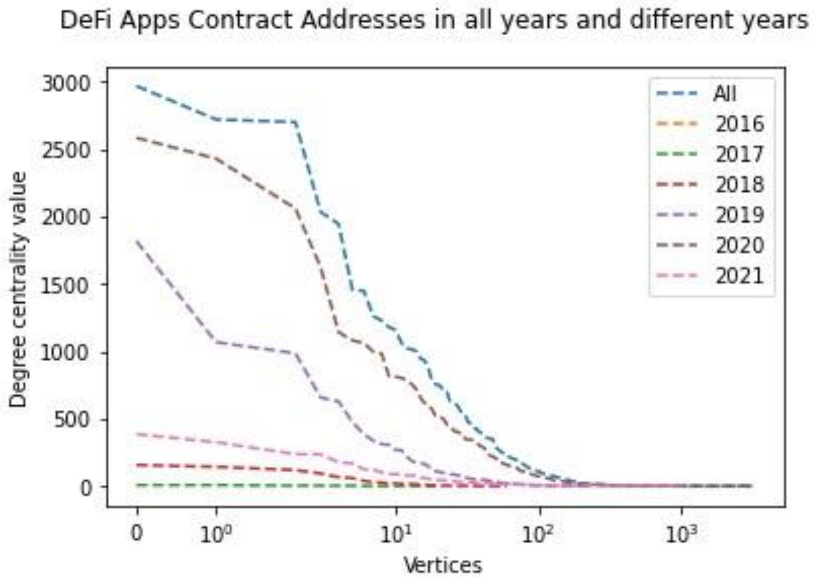


Figure 16: ***Degree centrality values of DeFi Apps Token Addresses in all years and different years***

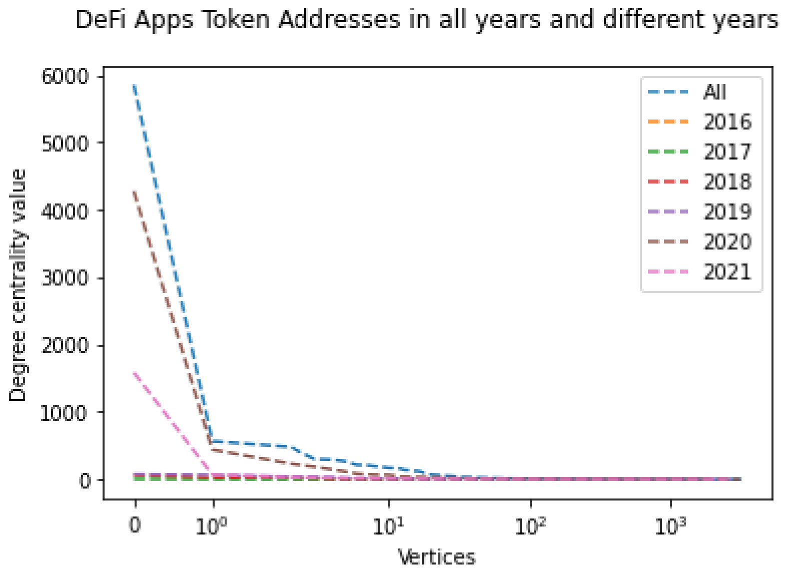
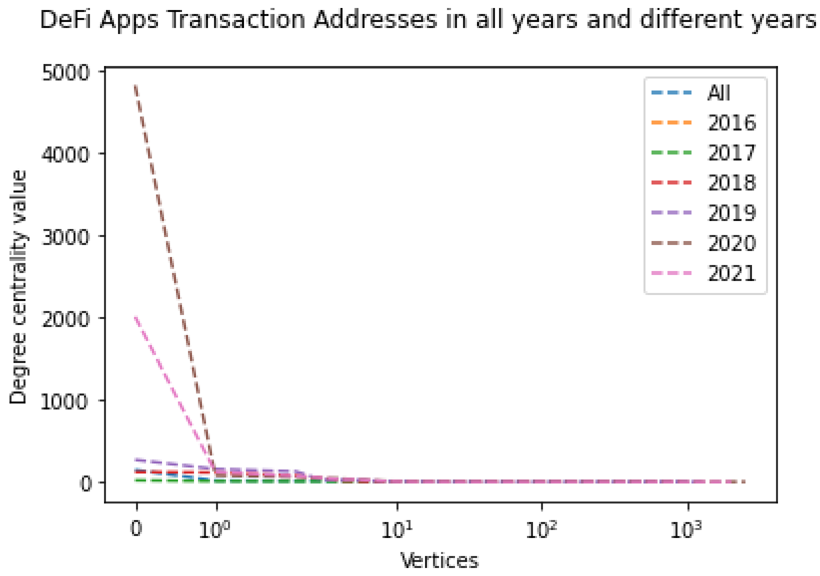


Figure 17: ***Degree centrality values of DeFi Apps Transaction Addresses in all years and different years***



In Figure 15, Figure 16, and Figure 17, it can be observed that the DeFi Apps addresses of contract network, token network, and transaction network share the same trend in their degree centrality values in all and different years, where 2020 is the year with the highest degree centralities for some of the DeFi Apps addresses in all three networks while 2021 is the year with the second highest, along with 2019 being the third.

Figure 18: ***Coreness values of DeFi Apps Contract Addresses in all years and different years***

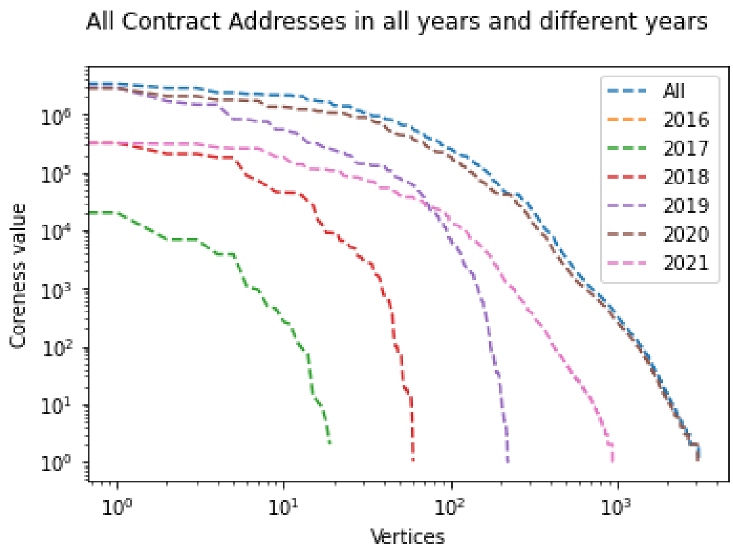


Figure 19: ***Coreness values of DeFi Apps Token Addresses in all years and different years***

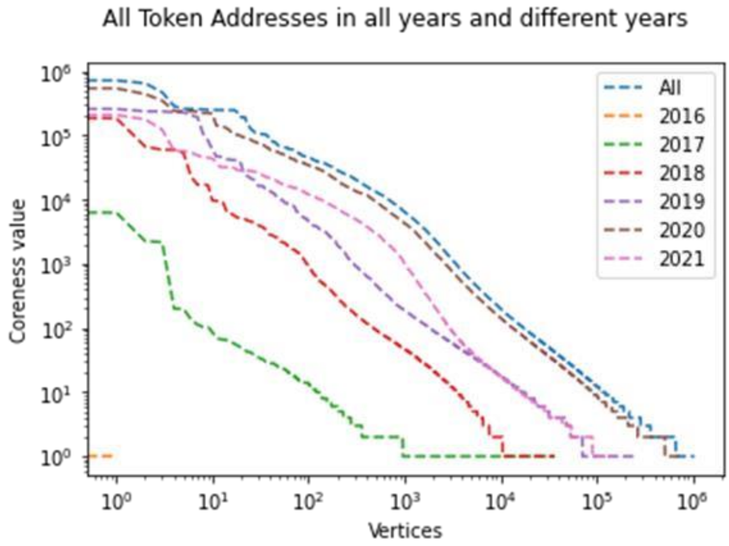
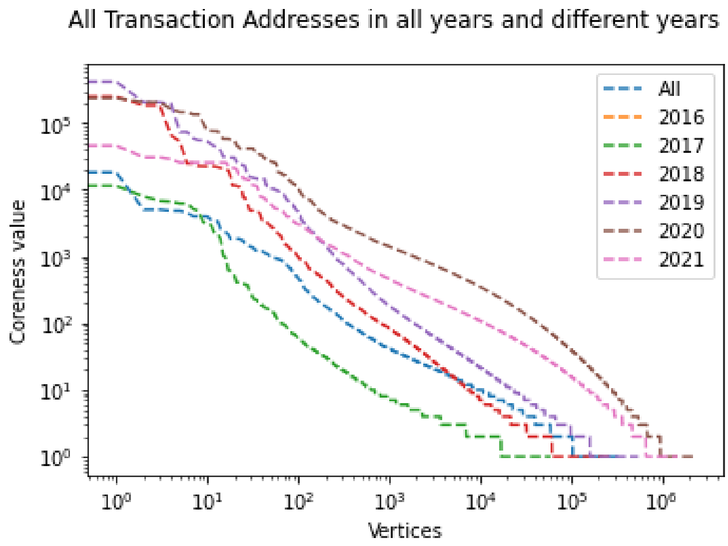


Figure 20: ***Coreness values of DeFi Apps Transaction Addresses in all years and different years***



The coreness, also known as the shell index, of any vertex of the network is k if it is a member of the k-core but not a member of the k+1-core, where the k-core of a graph is a maximal subgraph in which each vertex has at least degree k.

In Figure 18 and Figure 19, the coreness values of the vertices in the token network and the contract network are the highest in all the years, 2021, and 2020 among different years, however, in Figure 20, the coreness values of the vertices in the transaction network are the highest in 2021, 2020, and 2019 among different years. It is interesting to see the corenesss values of the vertices in the transaction network in all the years are the 5th highest among different years. It could be interpreted that the vertices in the transaction network are more influential in the spreading dynamics in terms of individual years rather as a whole in terms of all the years.

6.5 ERC-compatible and incompatible addresses on the contract network

ERC is an acronym for Ethereum Request for Comments. ERC is an application-level standard for Ethereum and it can include token standards, name registries, library formats, and such. There are currently two types of ERC addresses on the contract network in the DeFi area on Ethereum and they are ERC-20 and ERC-721. ERC-20 is a common standard for creating tokens on the Ethereum blockchain, and it defines a set of rules for all the ERC20 tokens to interact seamlessly with one another. ERC-721 is a standard for non-fungible tokens on the Ethereum blockchain, where these tokens are unique, and they will not be the same as any other tokens.

I have acquired 3530 contract addresses belonging to the Ethereum DeFi in the end of 2020, in which there are 102 ERC-20 compatible addresses and 1 ERC-721 compatible addresses. For the active addresses of the 3530 addresses, there are only 3109 of these addresses in which 101 of them are ERC-20 compatible and no address is ERC-721 compatible. This can be interpreted that only a few of these ERC-20 and ERC-721 compatible addresses are created but not in use in the network and there are some ERC-20 incompatible addresses created and not in use too.

The time period for this analysis is from 2016 to 2021.

Therefore, it can be observed that the majority (more than 96%) of the contract addresses on Ethereum DeFi are ERC-20 incompatible and the contract address with the maximum degree is ERC-20 compatible as well.

Figure 21: ***Degree centrality values of the ERC-20 compatible and ERC-20 incompatible Contract addresses in all years***

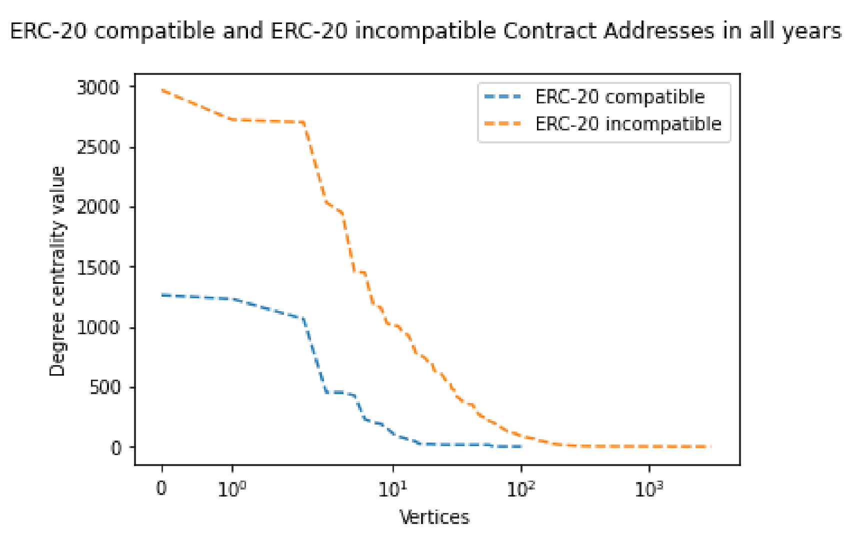
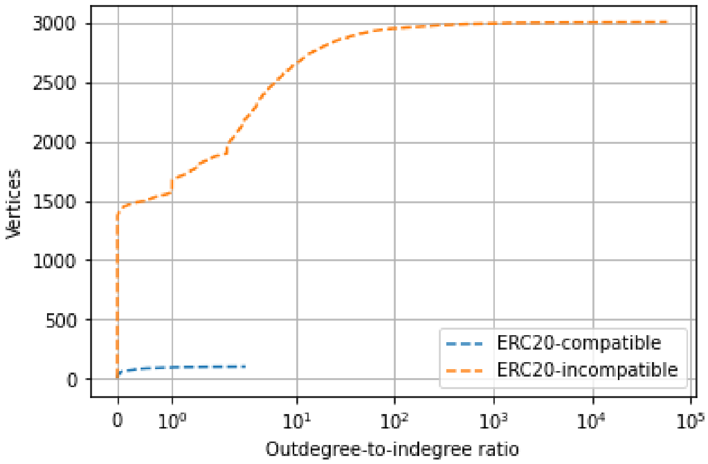


Figure 22: ***Outdegree-to-indegree ratio of the ERC-20 compatible and ERC-20 incompatible Contract addresses in all years***

ERC-20 compatible and ERC-20 incompatible Contract Addresses in all years



In Figure 21, the overall ERC-20 incompatible contract addresses have higher degree centrality values than the overall ERC-20 compatible contract addresses. This shows that the ERC-20 incompatible contract addresses are more central than the ERC-20 compatible ones, and therefore, the ERC-20 incompatible contract addresses are more important than the compatible ones in the network.

In Figure 22, half of the ERC-20 compatible and the ERC-20 incompatible contract addresses have more outdegrees than indegrees and half of those addresses have more indegrees than outdegrees. It can be generally observed that the ERC-20 incompatible contract addresses have more high outdegree-to-indegree ratios than the compatible addresses. Interestingly, this shows that there are some of the ERC-20 incompatible contract addresses are mainly responsible to call other contracts to execute functions for many times.

Overall, it can be concluded that the ERC-20 incompatible contracts addresses are more active and dominant than the ERC-20 compatible contract addresses on Ethereum DeFi. It is interesting that the Ethereum DeFi network does not use more ERC-20 compatible contract addresses to run its operations and this may mean that the Ethereum DeFi network has or is developing its own ERC standards to serve the needs of the market.

7. Conclusion

In this paper, I have investigated network properties, local graph properties, and global graph properties of the four blockchain networks on Ethereum DeFi, such as trace network, contract network, transaction network, and token network. Then I have analysed contract network, transaction network, and token network on a year-by-year and all-years level and the ERC compatibility in contract addresses has also been studied.

I find that the four Ethereum blockchain networks in the field of decentralized finance are weakly connected and the contract network and the trace network are comparatively the most centralized in terms of the DeFi Apps smart contracts. These DeFi networks are of more than or equal to two disconnected components and generally, the vertices of these networks are of lowquality. In these four DeFi networks, their addresses' interactions are not balanced across the networks and the networks' addresses follow the power-law distribution where, a certain small subset of addresses has some of the highest outdegree-to-indegree ratios or some of the highest indegree-to-outdegree ratios, while most addresses are the targets of recipient more than they are the targets of sender.

In all four DeFi blockchain networks, the contract network has the most tightly connected community structure, whereas the other networks do not have such tightly connected community structures.

The DeFi Apps smart contract addresses in the networks have unilateral transactions with the Ethereum DeFi users, as seen largely from the few bidirectional edges in the trace network and the transaction network.

As the density and the reciprocity of the DeFi token network's directed graph is comparatively higher, we can conclude that the token exchange on Ethereum DeFi is vehement and in these exchanges, different kinds of tokens are traded back and forth from a specific set of addresses.

Many interesting observations are made for the DeFi contract network. As the self-loops in the contract network is the highest in percentage comparing to other networks, the contract addresses call themselves many times. The high density of the DeFi contract network shows that it is the most tightly connected. In the interaction among the contract addresses in the DeFi contract network, the DeFi Apps smart contracts generally receive instructions from other contracts more than send instructions, and particularly, there is a small number of DeFi Apps smart contracts are only used to be called on by other smart contracts about 100,000 times.

Moreover, the smart contracts in the DeFi contract network tend to call on the same or the similar contracts to themselves.

In the DeFi contract network regarding the ERC compliance standards, it can be observed that non-fungible tokens have yet to be adopted in Ethereum DeFi and the ERC-20 standard does not seem to be used widely in the DeFi networks. This can imply that the Ethereum DeFi network may have separated itself from the general Ethereum blockchain network and it is going to progress as an industry developing its own ERC standards.

The year-by-year analyses tell us that all four DeFi blockchain networks have the most activities in 2020, and that there appears to be a trend where the Ethereum DeFi networks are becoming more popular year-by-year.

8. Limitations and discussion

Due to the equipment limitation, some insights are missed as the year-by-year analyses on the trace network cannot be done and some computational graph analytics cannot be performed on the networks. Moreover, the paper's specific sampling methods may have missed some addresses on Ethereum that may otherwise connected the vertices in the sampled networks, and as a result, it may cause the computational graph analytics to be slightly deviated from the true results.

9. References/Bibliography

[1] Chen, Y., & Bellavitis, C. (2020). *Blockchain disruption and decentralized finance: The rise of decentralized business models*. Journal of Business Venturing Insights, 13. doi:10.1016/j.jbvi.2019.e00151

[2] D. (2020). 2020 Q2 Dapp Market Report by DappReview. Retrieved October 19, 2020, from https://dapp.review/article/274/2020-Q2-Dapp-Market-Report-by-DappReview

[3] D. (2020). *2020 Q2 Dapp Market Report* by DappReview. [image] Available at: https://dapp.review/article/274/2020-Q2-Dapp-Market-Report-by-DappReview [Retrieved October 19, 2020,].

[4] Grigo, J. (2020). *Decentralized Finance (DeFi) – A new Fintech Revolution?* Bitkom, 15-16.

[5] Lee, X. T., Khan, A., Gupta, S. S., Ong, Y. H., & Liu, X. (2020). *Measurements, Analyses, and Insights on the Entire Ethereum Blockchain Network*. Proceedings of The Web Conference 2020. doi:10.1145/3366423.3380103

[6] Chen, W., Zhang, T., Chen, Z., Zheng, Z., & Lu, Y. (2020). *Traveling the token world: A graph analysis of Ethereum ERC20 token ecosystem*. Proceedings of The Web Conference 2020. doi:10.1145/3366423.3380215

[7] Jia, N., Kong, Q., & Huang, H. (2020). How Similar Are Smart Contracts on the Ethereum? Communications in Computer and Information Science Blockchain and Trustworthy Systems, 403-414. doi:10.1007/978-981-15-9213-3_32

[8] Bistarelli, S., Mazzante, G., Micheletti, M., Mostarda, L., & Tiezzi, F. (2019). Analysis of Ethereum Smart Contracts and Opcodes. Advanced Information Networking and Applications Advances in Intelligent Systems and Computing, 546-558. doi:10.1007/978-3030-15032-7_46

[9] Pinna, A., Ibba, S., Baralla, G., Tonelli, R., & Marchesi, M. (2019). A Massive Analysis of Ethereum Smart Contracts Empirical Study and Code Metrics. IEEE Access, 7, 7819478213. doi:10.1109/access.2019.2921936

[10] Oliva, G. A., Hassan, A. E., & Jiang, Z. M. (2020). An exploratory study of smart contracts in the Ethereum blockchain platform. Empirical Software Engineering, 25(3), 1864-
1904. doi:10.1007/s10664-019-09796-5

[11] Wang, Z., Jin, H., Dai, W., Choo, K. R., & Zou, D. (2020). Ethereum smart contract security research: Survey and future research opportunities. Frontiers of Computer Science, 15(2). doi:10.1007/s11704-020-9284-9

[12] Gupta, B. C., Kumar, N., Handa, A., & Shukla, S. K. (2020). An Insecurity Study of Ethereum Smart Contracts. Security, Privacy, and Applied Cryptography Engineering Lecture Notes in Computer Science, 188-207. doi:10.1007/978-3-030-66626-2_10

[13] Victor, F., & Lüders, B. K. (2019). Measuring Ethereum-Based ERC20 Token Networks.
Financial Cryptography and Data Security Lecture Notes in Computer Science, 113-129. doi:10.1007/978-3-030-32101-7_8

[14] S. Somin, G. Gordon, and Y. Altshuler. 2018. Social Signals in the Ethereum Trading Network. CoRR abs/1805.12097 (2018). arXiv:1805.12097 http://arxiv.org/abs/1805.12097

[15] Somin, S., Gordon, G., & Altshuler, Y. (2018). Network Analysis of ERC20 Tokens Trading on Ethereum Blockchain. Unifying Themes in Complex Systems IX Springer Proceedings in Complexity, 439-450. doi:10.1007/978-3-319-96661-8_45

[16] Meegan, Xavier. (2020). Identifying Key Non-Financial Risks in Decentralised Finance on Ethereum Blockchain. 10.13140/RG.2.2.27175.57769.