# An Empirical Analysis of Ethereum Airdrop

**Xuxuan Zhou**[1] **Chris Liu**[1] **Jiangshan Yu**[2] **Xiapu Luo**[3] **Eric Lo**[1]

[1]**The Chinese University of Hong Kong** [2]**Monash University**

{xxzhou, cyliu, ericlo}@cse.cuhk.edu.hk j.yu.research@gmail.com

[3] **The Hong Kong Polytechnic University**

csxluo@comp.polyu.edu.hk

## Abstract

Cryptocurrencies usually conduct ICO (Initial Coin Offering) to raise funds and employ various methods to promote their tokens. Being one of the most important promoting strategies to gain popularity and awareness, airdrop refers to a process of distributing free tokens to many accounts. Airdrop has been widely used since its first launch by the token *OmiseGO*. However, little is known about this popular activity. To fill in the gap, in this paper, we conduct the first systematic study of airdrop on Ethereum. More precisely, we first propose new algorithms to extract airdrop data from Ethereum. Based on the data, we characterize airdrop events and profile users who get tokens through the airdrop. Our empirical analysis reveals many new observations and insights.

## 1 Introduction

Blockchain and cryptocurrencies have recently attracted a lot of attention. In particular, the public has showed great interest in the Initial Coin Offering (ICO), which is a new way for the development team to raise funds by selling the cryptocurrencies. Different ICOs raised over $31 billion US dollars between January 2016 and August 2019[Howell *et al.*, 2020]. One of the major challenges for them is to promote the projects and attract users to use their tokens[Fröwis and Böhme, 2019].

Being a widely-used promoting strategy for ICOs, Airdrop distributes free tokens to the users to raise their awareness. Usually, users need to complete some tasks or hold some other popular cryptocurrencies (e.g., Bitcoin, Ethereum, etc.) to get tokens. These tasks include following the project's social media such as Facebook or enter the project's Telegram channel, letting the users learn more about their projects. For example, the token OmiseGO [1] conducted the first token airdrop to attract the public's interest in OmiseGO. To some extent, such airdrop achieved great success as many people went to search for this project during the airdrop, thus increasing the brand awareness[Fröwis and Böhme, 2019]. Since then, the airdrop has been used by more and more teams to promote their projects.

However, little work has been done to understand the airdrop. It is meaningful to analyze this activity as many teams are still conducting the airdrops and the analysis could provide insights and guidelines to them. To fill the gap, in this paper, we conduct the *first* systematic analysis on airdrops by characterizing their behaviors, and profiling the users involved in airdrops. Although Airdrop could happen on many blockchain platforms (e.g., Bitcoin, Ethereum, etc.), we focus on Ethereum because it supports the majority of the ICO [Victor and Lüders, 2019].

We first collect the data relevant to airdrops from Ethereum. It is non-trivial to collect such data because they are mixed up with other token transfer activities. To solve this problem, we first extract the token transfer events from the Ethereum and build a token transfer network similar to [Victor and Lüders, 2019, Somin *et al.*, 2018]. Then, we design and implement new algorithms to identify the airdrop data from the token transfer network (Section 3). Then, we analyze the data to have a better understanding of airdrop from two aspects, including the airdrop campaigns, and the users involved in airdrops. In particular, we obtain many new observations and insights by answering seven research questions. For example, we found that airdrop has becoming more and more popular since the middle of 2017. Moreover, by examining the behavior patterns of users involved in the airdrops, we found that airdrop users use different strategies to attend the airdrop, such as controlling multiple accounts to get the airdrop tokens, based on which we classify them into different groups and analyze their properties.

To summarize, we make the following main contributions.

1. To the best of our knowledge, it is the *first* systematic analysis of Ethereum Airdrop. We design a new tool to identify and extract the data relevant to airdrop from Ethereum.
2. We characterize the airdrops from several aspects. The new observations and insights are useful to the newcomers for launching effective airdrops.
3. We profile the users involved in airdrops with many new observations. For example, greedy users use automated approaches to collect many tokens.

[1]OmiseGO Homepage, `https://omisego.network/`. Last accessed 29 June, 2020

## 2 Background

### 2.1 Token

Many development teams leverage existing blockchain platforms to create new tokens for the sake of raising funds. More precisely, they usually develop smart contracts to issue their tokens, which will be bought by the investors through money or other cryptocurrencies (e.g., Bitcoin, Ether, etc.). According to the data in *coinmarketcap.com* [2], there are over 2000 tokens traded on the market, and about 89% of them are implemented as smart contracts running on Ethereum. Therefore, in this paper, we focus on the tokens running on Ethereum.

### 2.2 Token implementation and standards

To launch a token on Ethereum, developers will develop a smart contract to specify the token information and create functions for different token operations, such as transferring the tokens, etc. Although the token smart contract can be implemented in an arbitrary way, for the ease of inter-operations, some technical standards have been defined to guide the development of the token smart contract. For example, ERC-20 is the most widely used standard[Somin *et al.*, 2020] , which defines several functions and events

After the bytecode of a smart contract has been deployed to the Ethereum, to identify the relative functions or events, we should use their corresponding signatures, which are the first 4 bytes of the Keccak256 hash of their functions name or events name. As specified by ERC-20, whenever token transfer happens, the *Transfer* event must be triggered. Therefore, we leverage the *Transfer* event to trace the occurrence of tokens transfer. Token smart contracts that do not follow the standards or use a different *Transfer* event are not considered in this paper.

### 2.3 Airdrop

By distributing free tokens to other accounts, airdrop is one of the most important promoting strategies for tokens to raise the popularity and awareness of the project [Fröwis and Böhme, 2019]. It involves **airdropper accounts** that hold a large number of tokens and distribute them to the accounts of **airdrop user**.

As shown in Figure 1, there are two general kinds of airdrops: **Proactive Airdrop**[3] and **Reactive Airdrop** [4]. The former will actively distribute the tokens to the airdrop user accounts in a short period (usually in one or several days) after the airdrop users finish some tasks such as following the project's Twitter or entering their project's Telegram channel. The latter relies on a smart contract with an airdrop function. Once airdrop users invoke the airdrop function, the smart contract will transfer the tokens from the airdropper account to the airdrop user account.

[2]`https://coinmarketcap.com/`. Last accessed June 30, 2020

[3]`https://airdropalert.com/`. Last accessed June 30, 2020

[4]Airdrop via smart contract receiving 0 ETH. How is it done?, `https://reddit.com/r/ethdev/comments/7yx5yr`. Last accessed June 30, 2020

### 2.4 Airdrop users

Since a user can have more than one Ethereum account [Hileman and Rauchs, 2017], an airdrop user may use more than one account to join the airdrop in order to collect more tokens[5]. Hence, according to the number of Ethereum accounts used by an airdrop user, we can divide them into two groups, namely *individual airdrop user* who uses only one Ethereum account to join the airdrop and *controller* who controls more than one Ethereum account to participate in an airdrop. We call the Ethereum accounts controlled by a controller as *controlled accounts*. Moreover, a controller can be a **manual controller** or an **automatic controller** depending on whether the controller uses programs or smart contracts to manage its accounts automatically. According to the type of smart contracts used by **automatic controller**s to join the airdrop, we divide them into **self-destruct controller**[Chen *et al.*, 2020] and **program controller**. The former exploits self-destruct accounts to collect reactive airdrop tokens. Self-destruct account is a type of smart contract that performs the *SELF-DESTRUCT* operation to remove the smart contract from the Ethereum blockchain and send the remaining Ether to a designated account. The latter uses normal smart contracts to collect tokens from airdrop and transfers tokens among its Ethereum accounts.

## 3 Data Collection and Preparation

This section describes how we obtain the relevant data and evaluate the quality of such data.

### 3.1 Identifying token transfer

Airdrop is a kind of token transfer activity. According to Ethereum token standard, whenever a token transfer event happens, a *Transfer* event must be triggered and recorded on blockchain. Therefore, we extract all the *Transfer* events from Ethereum blockchain to collect token transfer data. Concretely, we instrument the *Geth* [Lee, 2019] client of Ethereum to obtain the blockchain data from the debut date of Ethereum to December 31st, 2018, and then extract the token *Transfer* event according to the signatures of ERC-20 [Victor and Lüders, 2019]

### 3.2 Extracting airdrop data

We design new methods to extract the airdrop data from the token transfer data.

**Proactive airdrop** Since Proactive airdrop usually distributes many tokens to other Ethereum accounts in a short period of time, fewe first identify such airdrop periods by capturing the bursts of token transfer. For example, Figure 2 shows the number of Transfer events per day of the token *OmiseGO* [6] from its creation to the end of 2017. Obviously, there is a burst of token transfer within a certain period, which is exactly the airdrop period of *OmiseGO*.

We design a new approach based on the burst detection algorithm [Kleinberg, 2003] to identify and extract airdrop

[5]Airdrop Hunters, `https://bitcointalk.org/index.php?topic=2844500.0`. Last accessed June 30, 2020

[6]OmiseGO Airdrop, `https://twitter.com/omise_go/status/907931559121924096`. Last accessed June 30, 2020

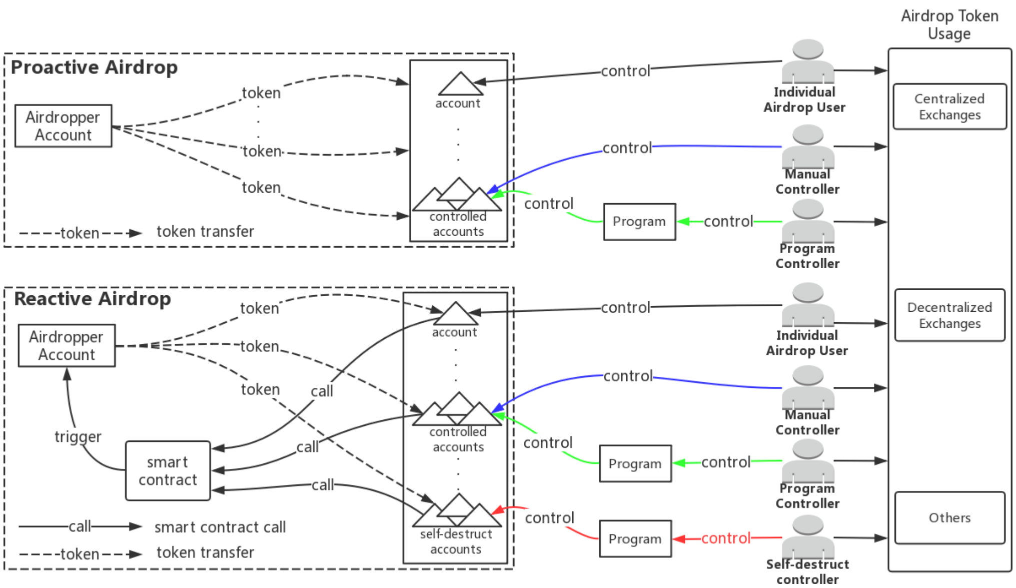



Figure 1: Roles and their interactions in Airdrop

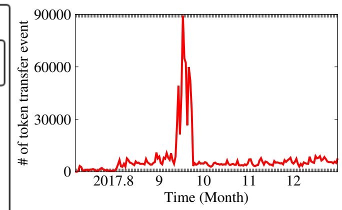



Figure 2: Number of Transfer events per day of OmiseGO

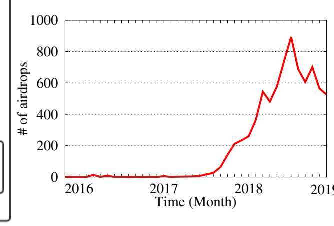



Figure 3: Airdrop time distribution

token transfer events. First, we identify all bursty periods of a token according to the number of token transfer events involving its public address. Next, for each account $X$ that has ever sent tokens to other accounts $Y$ in that period, we regard $X$ as a potential airdropper if it has never received any Ether from $Y$ before, because airdrop supposes to distribute tokens for free. After that, to filter out noise, we only keep those accounts that have sent tokens to more than $N$ different receivers, and have sent only one type of token in its entire lifetime. For those accounts, we count their token transfer events, and further prune the accounts whose number of token transfer events are lower than the average. Finally, we regard those accounts, whose token transfer events occupy more than $T\%$ of the total number of token transfer events in that period, as the airdropper accounts and their token transfer events as the airdrop data.

**Reactive airdrop** Since reactive airdrop employs smart contract to distribute tokens, we can identify the airdrop data by examining the corresponding smart contract and the transactions. For example, *BroFistCoin*[7] is a token that had conducted a reactive airdrop. By identifying its smart contract and its token transfer activities, we found that its reactive airdrop requires airdrop users to send a nominal amount of Ether (denoted as $M$, which is usually $10^{-6}$ Ether) to get the token. Therefore, for each token, we extracted all its token transfer data and regard a transaction as airdrop transaction if it involves an input of less than $M$ Ether and an output of a token.

**Evaluation** Before conducting further study on the collected data, we evaluate the data quality. In particular, we first compare airdrop data extracted by our approach with the known airdrop information listed on two airdrop advertisement websites, including *airdropking.io* [8] and *airdrops.io* [9]. We remark that these two websites are not authentic as anyone can register their airdrop information there. In this evaluation, we set $N = 50$ and $T = 80$ to be some stringent values in order to eliminate the airdrops that are listed on airdrop websites but have been cancelled or deferred. For reactive airport, we set $M = 0.001$.

Table 1: Airdrop data extraction quality

| | Number of Airdrops listed | Found by program |
|---|---|---|
| airdropking.io | 752 | 65.2% |
| airdrops.io | 650 | 68.9% |

Table 1 shows that our approach can report 65.2% and 68.9% of the airdrop events listed on *airdropking.io* and *airdrops.io*, and our approaches find a total of 7,662 airdrops. For those that are not listed on *airdropking.io* or *airdrops.io*, we randomly pick 100 airdrops and manually check their information through search engines. The results show that 68 airdrops have officially announced about their airdrop events elsewhere (e.g., BitcoinTalk forum [10], Telegram).

However, 32 airdrops were missing from Google Search. After manually checking the token transfer history of those airdropper accounts, we discovered similar patterns in the distributing process of these 32 airdrops and the other 68 airdrops. Both groups of airdropper accounts distributed free tokens to a large number of accounts. Furthermore, some airdrops, although not found from Google Search, were completed by function explicitly named *airdrop*. This observation verifies and substantiates the algorithms we derived to collect the airdrops.

The following assumptions were made to reveal reasons for not finding their airdrop information from Google Search. First, we assumed that some of these airdrops were listed on other platforms, such as Wechat, which included information not accessible from Google Search. Second, some of the air-

[7] `https://brofistcoin.io` Last accessed June 30,2020
[8] `https://airdropking.io/`. Last accessed June 30, 2020
[9] `https://airdrops.io/`. Last accessed June 30, 2020
[10] `https://bitcointalk.org/`. Last accessed June 30, 2020

drops might be launched while still at testing phase. We decide to retain this type of airdrops to conduct data analysis for the following reasons. Even though these airdrops were not in production, the data itself was still invaluable. We could extract and study the information about the target accounts of these test airdrops. Similarly, we would be able to obtain information about the kind of airdrop methods that were chosen to use and so on. All aspects above could provide both the cryptocurrency customers and token development teams a detailed guide.

### 3.3 Classifying airdrop users

We propose a new algorithm (Algorithm 1) that leverages the domain knowledge to map the Ethereum accounts involved in the airdrop data to different types of users (e.g,. manual controller, program controller), As shown in Algorithm 1, we first identify the set of accounts controlled by a potential controller $Y$ based on the following two intuitions (Lines 3-4). First, $Y$ may use multiple accounts to collect tokens. Second, $Y$ will not use those accounts to carry out other transactions. For those that do not follow this intuition, we regard them as individual airdrop users (Line 6). Lines 9-15 further classify the potential controllers. In particular, if a potential controller $Y$ has more than one controlled accounts (Line 9), we label it as a self-destruct controller if it has a self-destruct smart contract (Lines 10-11). If a controller $Y$ has written a program (e.g., a script) to collect tokens using multiple accounts, the collection process should happen in a short timespan (e.g., 2 Ethereum blocks), so we label it as a program controller (Lines 12-13). Otherwise, that should be a manual controller who does the same manually and things should happen across a longer timespan (Line 15).

### 3.4 Identifying cryptocurrency exchange accounts

We also need to determine whether an Ethereum account belongs to a cryptocurrency exchange or not for further analysis. Since existing platforms like *etherscan* [11] and *bloxy* [12] list the addresses of almost all cryptocurrency exchanges, we identify 149 Ethereum account addresses as centralized exchanges and 425 addresses as decentralized exchanges.

[11] `https://etherscan.io`. Last accessed June 30, 2020

[12] `https://bloxy.info`. Last accessed June 30, 2020

**Algorithm 1:** Classifying airdrop user

1 **foreach** *airdrop receiver account $R$ in airdrop data* **do**
2     Check the token transfer data of $R$
3     **if** *$R$ has sent tokens to only one account $Y$ in its entire lifetime* **then**
4         we treat $R$ as a potential controlled account and $Y$ as a potential controller
5     **else**
6         label $R$ as an **individual airdrop user**
7 **end**
8 **foreach** *potential controller $Y$* **do**
9     **if** *$Y$ has more than one controlled accounts* **then**
10         **if** *$Y$'s controlled accounts are self-destruct smart contract* **then**
11             label $Y$ as **self-destruct controller**
12         **else if** *$Y$ received tokens from more than $n$ controlled accounts within 2 consecutive Ethereum blocks* **then**
13             label $Y$ as **program controller** because only a program can manipulate so many accounts on short notice
14         **else**
15             label $Y$ as **manual controller**
16     **else**
17         label $Y$ as **individual airdrop user**
18 **end**

## 4 Characterizing Airdrop Events

We characterize airdrop events by answering three research questions.

**RQ1: How many airdrops have been launched?**

**Motivation:** To give an overview of airdrops, we show the number of occurrences of airdrops and when they happened.

**Approach:** We identify the airdrops according to the algorithm in Section 3.2, and plot the number of airdrops happened between Nov. 2015 and Dec. 2018.

**Results:** We collected 7,662 airdrops that involved 4,079 tokens. Note that some tokens launched more than one airdrops. Since we have collected the token Transfer events in section 3.1, we could identify the token smart contracts that ever transferred the tokens. For every such smart contract, we treat it as a token and thus find 85,454 distinct tokens according to the unique addresses from the token transfer data. Around 4.8% of tokens have conducted the airdrops. Fig.3 shows the number of airdrops occurred between Nov. 2015 and Dec. 2018. We can see that the number of airdrops increases over the whole period. The trend of the airdrop begins from July 2017, reaches a peak in the middle of 2018, and still stays at a high level through the rest of 2018. In other words, the airdrop activities have become popular since July 2017, especially in 2018.

**Conclusion:** We observed 7,662 airdrops launched, and it becomes more and more popular after the first airdrop.

**RQ2: When will the token development teams launch the airdrops?**

**Motivation:** To infer the purposes of the airdrop, we examine when the token development teams launch the airdrops.

**Approach:** We determine when the token development teams launch the airdrop after they create the tokens.

**Results:** Fig. 4 shows the distribution of the time gap between the token creation time and airdrop begin time. Each point (x,y) denotes that y percent of the airdrops was launched within x days after the creation of their token contracts. Fig 4 illustrates that 20% of the airdrops happened within 10 days after token contracts were created. In other words, the majority of the token development teams did not launch an airdrop immediately after they created the token contract. We can see from the graph that 49% of the airdrops were launched in the period from 10 days to 100 days after the creation of the token contract. Besides, it is notable that there are still 31% of the

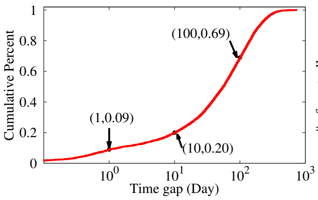



Figure 4: Distribution of the time gap between the token contract creation time and airdrop begin time

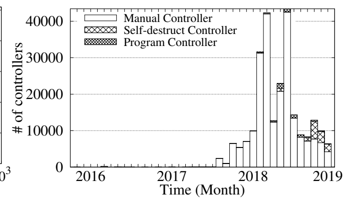
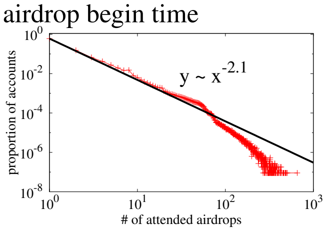
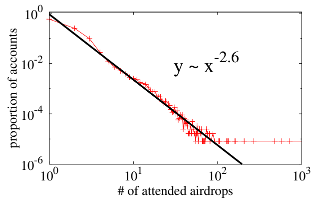


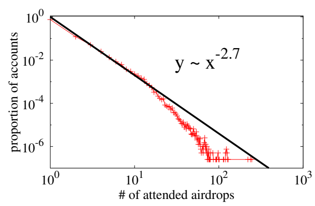

Figure 5: Controllers' time distribution




(a) Individual airdrop user




(b) Controllers




(c) Controlled accounts

Figure 6: Distribution of the number of airdrops joined by airdrop users and controlled accounts

airdrops happened 100 days after the creation time of airdrop tokens.

**Conclusion:** The data shows that most of the development teams did not launch an airdrop right after creating the token smart contract.

## 5 Profiling Airdrop user

We characterize airdrop users to gain a better understanding of this community.

**RQ3: How many airdrops did different airdrop users and controlled accounts participate in?**

**Motivation:** By analyzing the number of airdrops that different airdrop users participate in, we could infer their passion for this activity. We also examine how many airdrops the controlled accounts join to determine whether or not the controllers use the same set of controlled accounts to join different airdrops.

**Approach:** By correlating the results obtained, we plot the distribution of the number of airdrops that different airdrop users and controlled accounts joined.

**Results:** Fig. 6 illustrates the distribution of the number of airdrops that individual airdrop user, controller, and controlled account participated in. We can see that all of them follow the power law, meaning that there are a few airdrop users and controlled accounts attend many airdrops whereas many airdrop users and controlled accounts participate a small amount of airdrops. We also plot the power function $y \sim x^{-\alpha}$ that best fits the distribution. Fig. 6(a) and Fig. 6(b) indicate that although the controllers control more accounts than individual airdrop users to get the airdrop tokens, most of the controllers only attend one or several airdrop tokens. Fig. 6(c) shows that instead of controlling the same accounts to participate in different airdrops, controllers use different controlled accounts for different airdrops.

**Conclusion:** Most of the individual airdrop users, controllers and controlled accounts join a small amount of airdrops, and the distribution of the number of airdrops attended follow the distribution of power law.

**RQ4: How did the airdrop users use the tokens collected from the airdrops?**

**Motivation:** By exploring how the airdrop users use the airdrop tokens, we could learn whether the airdrop will be useful to the token circulation. For example, if most of the tokens are no used, the development teams may re-consider whether or not to conduct an airdrop and cryptocurrency customers may decide whether or not to attend the airdrop.

**Approach:** For each airdrop user group, we will calculate its airdrop token usage separately, and we classify the usage of the airdrop tokens into five categories: (1) Send to centralized exchanges (CEX) (2) Send to decentralized exchanges (DEX) (3) No trade (4) No exchanges (5) Send to the unknown

*No trade* means that the tokens have been launched on the exchanges, but the airdrop users keep them in the accounts without trading the tokens. *No exchanges* indicates that the tokens have not been launched on any exchanges and the airdrop users could not trade it on exchanges. *Send to the unknown* denotes that the airdrop users sent the tokens to the Ethereum accounts that do not belong to the exchanges. We only consider the Ethereum accounts that the airdrop users directly traded with to get a correct token usage.

Since each airdrop delivers a different volume of tokens, we normalize the token usage using the total number of tokens delivered for each airdrop. For example, assume token A delivers a total of 100 free tokens and 80 of them are sent to CEX and 20 of them are sent to DEX. Furthermore, there is another token B delivers a total of 5K tokens of which 2K of them are sent to CEX and 3K of them are sent to DEX. Then, we regard CEX get $(80/100+2000/5000)/2*100 = 60\%$ of usage and DEX get $((20/100+3000/5000))/2*100 = 40\%$ of usage.

**Results:** Table 2 shows the airdrop token usage of different user groups. The first column is about the four user groups. Column 2 to 6 list the airdrop usages that we described above.

For the individual airdrop user, only a small percent of the airdrop tokens sent to this group will be traded directly on the exchanges (both centralized exchanges and decentralized exchanges), which accounts for 2.2%. 11.5% of their tokens were traded to other accounts that do not belong to the ex-

Table 2: Airdrop Token Usage of Different Airdrop Users

| User Groups | To CEX | To DEX | No Trade | No Exchange | Unknown |
|---|---|---|---|---|---|
| Individual Airdrop User | 0.6% | 1.6% | 44.2% | 42.1% | 11.5% |
| Manual Controller | 8.0% | 7.3% | 23.9% | 33.8% | 27.0% |
| Self-destruct Controller | 0% | 2.5% | 18.7% | 71.3% | 7.4% |
| Program Controller | 2.8% | 5.9% | 17.0% | 28.9% | 45.4% |

changes. Maybe they sent the tokens to their other accounts before sending to exchanges. However, 44.2% of the airdrop tokens stay in the accounts without any trading activities. The reasons for *No Trade* could be complicated. Maybe the token value is too small for them to trade, or they want to keep the tokens for investment.

For the rest of the controllers, we could see that the self-destruct controllers have a similar usage as the individual airdrop users, but their *No Exchange* part is extremely high. The manual controllers and program controllers vary a lot compared with the other two user groups. First, Table 2 shows that the manual controllers and program controllers spend more of their received tokens on the exchanges than individual airdrop users and self-destruct controllers. Besides, we can also see that most of the airdrop user groups spend more of their received tokens on the decentralized exchanges than centralized exchanges. One possible reason is that more airdrop tokens were traded on the decentralized exchanges instead of the centralized exchanges. Among the 4,079 tokens collected by our algorithms, 1,847 tokens were traded on decentralized exchanges while 1,022 tokens were traded on centralized exchanges. Note that a token can be traded on centralized exchanges and decentralized exchanges.

On the one hand, we also observe that the ratios of *No Trade* and *No Exchange* in the manual controllers and program controllers are less than that in individual airdrop users and self-destruct controllers. On the other hand, the manual controllers and program controllers spend more of their received tokens on the *Unknown* part. We could conclude that compared to the individual airdrop users and self-destruct controllers, manual controllers and program controllers have more passion for trading the airdrop tokens and also, they prefer to get the airdrop tokens that could be traded on exchanges. Moreover, instead of trading the tokens directly on the exchanges, they may send the tokens to other unknown accounts before the tokens go to the exchanges.

**Conclusion:** Overall, the airdrop token usage of *No trade* and *No Exchange* take up the majority of the token usage in different airdrop user groups. Besides, for all of the airdrop user groups, only a small percent of the airdrop tokens were traded directly on the exchanges. Last but not least, the manual controllers and program controllers could have more passion for trading tokens on exchanges, and they could also select those airdrops that the airdrop tokens could be traded on exchanges.

## 6 Related work

### 6.1 Initial coin offering analysis

Initial Coin Offering is an activity for development teams to raise funds by selling the tokens to the investors. ICO has been intensively analyzed in these research works [Zetzsche *et al.*, 2017, Adhami *et al.*, 2018, Fenu *et al.*, 2018, Fisch, 2019] from different perspectives. Zetzsche et al. analyzed the ICO from the law and regulatory aspects and highlighted the risk of ICO, while the rest of three works focused on the factors of a successful ICO. What's more, Rhue et al. [Rhue, 2018] pointed out that reputation scores in the platforms were not reliable information to decide whether an ICO is to succeed, and investors should be careful about that.

### 6.2 Airdrops

As far as we know, there are little works related to airdrop studies. There is a work conducted by Martin Harrigan et al. [Harrigan *et al.*, 2018], which is the privacy concern about an airdrop conducted on Clam blockchain. The authors reminded that the sharing of addresses between blockchains is a privacy risk. Michael Fröwis and Rainer Böhme [Fröwis and Böhme, 2019] estimate the operational cost of different kinds of airdrops. The authors mainly analyzed two kinds of airdrops: push-style airdrops and pull-style airdrops, which corresponding to our reactive airdrops and proactive airdrops. They found that from the perspective of the distributors (airdroppers), push-style airdrops cost more gas than pull-style airdrops. However, pull-style airdrops impose a disproportional cost on airdrop users. Both of these works did not focus on the overall of the Ethereum airdrops, and also, the features of the airdrop users and our work could fill in this gap.

### 6.3 Token analysis

Recently, there are two works [Somin *et al.*, 2018, Victor and Lüders, 2019] related to the token analysis on the Ethereum. Somin et al. 's work is the first network analysis of the ERC20 tokens in the Ethereum. They constructed the network by treating every Ethereum account as a node and the trading activities between the nodes as the edges, and they demonstrated that the network displays strong power-law properties. While Friedhelm Victor et al. 's work has a more complete and comprehensive study on the ERC-20 token network. The authors analyzed the network properties of the ERC-20 token network, such as the degree distributions, density and components, clustering coefficients and so on, and they also analyzed the network flow of the ERC-20 network and indicated that the main use case of many tokens appeared to be trading. However, both of these works focus on the overall network properties of the Ethereum token network and do not further analyze the dedicated token activities such as ICO or airdrop, and our research could fill this gap.

## 7 Conclusion

In this paper, we conduct the first systematic analysis on the Ethereum airdrop, which includes the airdrop features, and airdrop users features. We find that the Ethereum airdrops have been popular since 2017 and it was still prevalent in 2018, and we also find that the development team did not launch an airdrop right after creating the token smart contract. Finally, we analyze the airdrop users features and find that the number of airdrops attended by the individual airdrop users and controllers follow a power law distribution, which means that most of them attend little airdrops. Besides, we also observe that the majority of the airdrop tokens sent to the airdrop user groups are no trade or no launch on any exchanges, and only a small percent of the tokens would be traded directly on the exchanges. In conclusion, our overall results help to gain a better understanding of the airdrops and give a more detailed guide to the token development team and cryptocurrency users.

## References


[Adhami *et al.*, 2018] Saman Adhami, Giancarlo Giudici, and Stefano Martinazzi. Why do businesses go crypto? an empirical analysis of initial coin offerings. *Journal of Economics and Business*, 100:64–75, 2018.

[Chen *et al.*, 2020] Jiachi Chen, Xin Xia, David Lo, and John Grundy. Why do smart contracts self-destruct? investigating the selfdestruct function on ethereum. *arXiv preprint arXiv:2005.07908*, 2020.

[Fenu *et al.*, 2018] Gianni Fenu, Lodovica Marchesi, Michele Marchesi, and Roberto Tonelli. The ico phenomenon and its relationships with ethereum smart contract environment. In *2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*, pages 26–32. IEEE, 2018.

[Fisch, 2019] Christian Fisch. Initial coin offerings (icos) to finance new ventures. *Journal of Business Venturing*, 34(1):1–22, 2019.

[Fröwis and Böhme, 2019] Michael Fröwis and Rainer Böhme. The operational cost of ethereum airdrops. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, pages 255–270. Springer, 2019.

[Harrigan *et al.*, 2018] Martin Harrigan, Lei Shi, and Jacob Illum. Airdrops and privacy: a case study in cross-blockchain analysis. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pages 63–70. IEEE, 2018.

[Hileman and Rauchs, 2017] Garrick Hileman and Michel Rauchs. Global cryptocurrency benchmarking study. *Cambridge Centre for Alternative Finance*, 33:33–113, 2017.

[Howell *et al.*, 2020] Sabrina T Howell, Marina Niessner, and David Yermack. Initial coin offerings: Financing growth with cryptocurrency token sales. *The Review of Financial Studies*, 33(9):3925–3974, 2020.

[Kleinberg, 2003] Jon Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

[Lee, 2019] Wei-Meng Lee. Connecting to the ethereum blockchain. In *Beginning Ethereum Smart Contracts Programming*, pages 61–69. Springer, 2019.

[Rhue, 2018] Lauren Rhue. Trust is all you need: an empirical exploration of initial coin offerings (icos) and ico reputation scores. *Available at SSRN 3179723*, 2018.

[Somin *et al.*, 2018] Shahar Somin, Goren Gordon, and Yaniv Altshuler. Network analysis of erc20 tokens trading on ethereum blockchain. In *International Conference on Complex Systems*, pages 439–450. Springer, 2018.

[Somin *et al.*, 2020] Shahar Somin, Goren Gordon, Alex Pentland, Erez Shmueli, and Yaniv Altshuler. Erc20 transactions over ethereum blockchain: Network analysis and predictions. *arXiv preprint arXiv:2004.08201*, 2020.

[Victor and Lüders, 2019] Friedhelm Victor and Bianca Katharina Lüders. Measuring ethereum-based erc20 token networks. In *International Conference on Financial Cryptography and Data Security*, pages 113–129. Springer, 2019.

[Zetzsche *et al.*, 2017] Dirk A Zetzsche, Ross P Buckley, Douglas W Arner, and Linus Föhr. The ico gold rush: It's a scam, it's a bubble, it's a super challenge for regulators. *University of Luxembourg Law Working Paper*, (11):17–83, 2017.