Analysis of multi-input multi-output transactions in the Bitcoin network

Phetsouvanh, Silivanxay; Datta, Anwitaman; Oggier, Frédérique

2021

Phetsouvanh, S., Datta, A. & Oggier, F. (2021). Analysis of multi-input multi-output transactions in the Bitcoin network. *Concurrency and Computation: Practice and Experience*, 33(1), e5629-. <https://dx.doi.org/10.1002/cpe.5629>

<https://hdl.handle.net/10356/154816>

<https://doi.org/10.1002/cpe.5629>

This is the peer reviewed version of the following article: Phetsouvanh, S., Datta, A. & Oggier, F. (2021). Analysis of multi-input multi-output transactions in the Bitcoin network. *Concurrency and Computation: Practice and Experience*, 33(1), e5629-, which has been published in final form at <https://doi.org/10.1002/cpe.5629>. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Use of Self-Archived Versions.

Downloaded on 26 Apr 2023 21:45:24 SGT

ARTICLE TYPE

Analysis of Multi-input Multi-output Transactions in the Bitcoin Network

Silivanxay Phetsouvanh¹ | Anwitaman Datta^{*1} | Frédérique Oggier²

¹School of Computer Science and Engineering,
Nanyang Technological University, Singapore

²School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore

Correspondence

*Anwitaman Datta Email:
anwitaman@ntu.edu.sg

Present Address

School of Computer Science and Engineering
Blk N4 (office: 02A-18), North Spine Nanyang
Avenue, Nanyang Technological University
Singapore 639798

Abstract

Distinct transactions among different and unrelated users are combined together to create a single Bitcoin transaction (mixing transaction) to obfuscate the relationships among the actual participants (more specifically, the wallet addresses used for the transactions). We consider multi-input multi-output transactions with at least two inputs and three outputs as proxy, to analyze four characteristic periods of ~ 50 days each, representing periods before the introduction of mixing, in its early days, during its growth, and after the volume of such multi-input multi-output transactions became more or less stable. Structural properties and characteristics of the transaction and wallet address networks are computed and compared, through standard tools, but also via the introduction of two novel techniques that provide indicators of mixing-like behaviors: (i) an entropy characterization to detect abnormally uniform inputs and/or outputs, and (ii) a connected component analysis of subgraphs formed by only multi-input multi-output transactions (showing cascades of such transactions). The contributions of this exploratory Bitcoin network analysis paper can thus be seen as two-fold. At a macroscopic level, the growth and stabilization periods are shown to stand out with respect to most considered metrics, while at a microscopic level, chains of multi-input multi-output transactions, and transactions with outlier behavior in terms of input/output entropies are identified for further investigation.

KEYWORDS:

Bitcoin Network, Graph Analysis, Mixing, Exploratory Data Analysis

1 | INTRODUCTION

Bitcoin¹ is a digital currency which is realized using cryptographic protocols for security. All transactional activities are logged in a distributed ledger called blockchain. The ledger is public, so anyone can read the content of any transaction. Nevertheless, there is no notion of identity of users in the Bitcoin system, and the wallets holding and operating Bitcoins are controlled through public key cryptography. Each user may have one or several wallet addresses, which are used to make transactions, and only the addresses are visible in the blockchain, not the identity of the owner, thus providing its users some extent of anonymity.

The fact that the transactions log data is publicly available and that Bitcoin is a major cryptocurrency makes it an ideal candidate to study, whether one is interested in financial networks, community behaviors, or forensics. Indeed, a number of research studies have analyzed the Bitcoin network. Typical statistics coming from the theory of complex networks include degree distributions and in particular checking whether the network exhibits power laws. This has been done for the Bitcoin network¹ in^{2,3} but with different purposes:² studied Bitcoin as a financial network, with focus on its business and economic impact across different geographic areas, while³ considered whether the idea that "rich get richer" also holds for Bitcoin

¹There are in fact several kinds of networks that might be inferred from Bitcoin transactions. We shall elaborate on this subsequently as and when we discuss specifics. For now, we use the term Bitcoin network as a catch-all term to imply any/all of the various possible networks.

FIGURE 1 Bitcoin subgraphs (generated with the Bitcoin network analysis and visualization tool BiVa⁵).

(it was answered positively). Since users are anonymous, illegal activities have been taking place over the Bitcoin network, leading to the need of and creation of forensics tools. Features of Bitcoin forensics tools (e.g.^{4,5}) typically include clustering or aggregation of wallet addresses that are likely to belong to either the same user, or a group of collaborating users, as well as finding paths between addresses or identified users, apart from extracting basic statistics and provide visualization. Over time, Bitcoin has become more popular and has been growing. Therefore more recent studies also consider the architecture to handle Bitcoin analytics (see e.g.⁶), how the employed analytics techniques evolve over time (e.g.⁷) and how to best visualize a subset of the information to aide analysis (e.g.⁸).

This work is exploratory in nature, and considers Bitcoin analytics by zeroing into one specific phenomenon, namely that of multi-input multi-output (multi-input/output) transactions. Circa late 2012 early 2013, the Bitcoin network witnessed the introduction of mixing (also known as tumblers or laundry) services, which would combine multiple isolated transactions together to create a single Bitcoin transaction, so that the relationships among individual transaction inputs and outputs are obfuscated. Subsequently, decentralized mechanisms to do so obviating the need of any dedicated service provider have been designed, prominently CoinJoin⁹, as well as alternatives such as ZeroCoin¹⁰. The underlying motivation of this work is to study the introduction of mixing in the Bitcoin network, and whether the presence of consequent multi-input/output transactions changed noticeably its statistical properties. Since there is no absolute indicator on whether a specific transaction on the blockchain

was created because of mixing, we use multi-input/output transactions (we define this precisely later in Section 2) as a proxy that can be studied, and in the process look for traces of mixing.

To perform any analysis, Bitcoin transactions stored in the blockchain are first represented as a directed graph. There are several possible graph representations. The most complete one is arguably a graph comprising nodes of two kinds, namely transaction and wallet address nodes (we refer to this graph as being dual mode). There is a directed edge from an address node to a transaction node if this address serves as an input to the transaction, and similarly, there is a directed edge from a transaction node to an address node if the address appears in an output of the transaction. This is illustrated in Figure 1a. The example network shown here was extracted using a Bitcoin visualization and analytics tool called BiVA⁵: it is a subgraph of radius 3 around the wallet address labeled 'O', from a data set¹¹ used for a case study on Ashley Madison data breach victim extortion using the EGRET¹² Bitcoin forensics framework. Figure 1b lists the wallet addresses and transactions (identified by their hash) present in this subgraph.

The address network is a directed graph obtained by creating a directed edge between two addresses if they are connected to each other by any transaction node (one is the input wallet, the other the output wallet) in the dual mode network. This is easily understood when there is one input and one output involved in a transaction, or as long as there is only a single wallet involved in either the input or the output of a transaction. However, when there are more than one input and more than one output, all input/output pairs are linked to generate a wallet address graph. For example, transaction labeled 22 has two input wallet addresses 0 and 27, and two output wallet addresses, 23 and 25. Therefore, in the derived address network, there is a direct link from 0 to 23 and 25, and from 27 to 23 and 25. This manner of creating a wallet address graph creates many spurious links that may not exist in practice, particularly when mixing is occurring. Such an obfuscation of relationships is by design a consequence of Bitcoin mixing. Existing studies (e.g.,^{3,2,5}) use the said heuristic to create the wallet address graph from the transaction log, even though it leads to a noisy representation of the actual relationships that exist among wallet addresses. This approach may have been less detrimental for early studies (e.g.^{3,2}) which analyzed network statistics for periods when there was hardly, if any, mixing.

On the contrary, a noisy wallet address graph is a problem for deanonymization/forensics purposes (e.g.^{4,5}), whose objective is to trace the flow of Bitcoin not only among addresses, but among actual users. Another heuristic, called address aggregation, is then used to identify which (all) addresses may belong to the same person. There is no perfect solution to do so, but there are several heuristics for attributing different wallet addresses to the same user based on transaction patterns (see e.g.^{13,14}).

The transaction network is a directed graph obtained by creating a directed edge between two transactions if they are connected via at least one wallet address. This graph does not suffer the uncertainty of wallet addresses, however there might be still discrepancies in node degrees. For example, if two wallet addresses are both outputs to one transaction, and inputs to another transaction, then they contribute to two to the respective transaction degrees in the dual mode network, but only to one in the transaction network.

In this paper, we consider transaction nodes both within the dual mode network and their respective transaction networks. We also study the derived wallet address graph, and the network derived by aggregating wallet addresses using some commonly used heuristics.

The transaction network characteristics will be studied in Section 3. Section 4 is dedicated to the wallet address networks, before and after aggregation, respectively in Subsection 4.1 and Subsection 4.2. The narrative through these sections follows our original question, namely, how multi-input/output transactions (used as a proxy for mixing) is influencing the statistics of the Bitcoin network. Section 4 also includes statistics reported by previous studies, both for validating the consistency of our study, as well as a point of comparison to determine how the network characteristics evolved over time. Our methodology consists of selecting four time periods of 49-50 days each, as representatives of different phases: a phase from the early days of Bitcoin, when no one was using mixing (8 December 2010 to 26 January 2011, days 700-749 in the history of Bitcoin), a phase just around/after the introduction of mixing services (6th April to 24 May 2013, days 1550-1598), a period where the fraction of multi-input/output transactions was significantly high than other periods (8 October to 25 November 2014, days 2100-2148), and a subsequent period (19 July to 6 September 2016, days 2750-2799), where the fraction has subsided to a lower but more steady value, even as the absolute number of such transactions also remain stable, and comparable to the period when the fraction of such transactions were highest. These periods were identified by studying the overall period spanning 3 Jan 2009 - 14 Apr 2018, and looking at daily transaction logs, which comprised of 3384 days (see Section 2 for the details). Each of these periods are analyzed and compared. The corresponding two-mode (transactions-wallet address) graph data for the four periods is available as CSV files¹⁵.

Our main contributions are found in Sections 3 and 4. After giving some basic statistics on multi-input/output transactions, the amount of Bitcoins they are involved in, and their in-/out-degrees as nodes in the dual mode network, we present in Section 3 an entropy (and normalized entropy) analysis that captures how uniform the Bitcoin amounts on their outgoing edges are. This can be used as a criterion to identify "mixing", whether voluntary (done by a mixing service or a decentralized mechanism) or naturally arising. This enables us to shortlist the most "interesting" nodes with respect to having high in-/out (normalized-)entropies. If users apply mixing for sensitive transactions, they may feel even safer if their transactions were to proceed through multiple hops of mixed transactions. We determine the conditional probabilities of (non-)multi-input/output transaction's inputs and outputs being used in/from other (non-)multi-input/output transactions. We then concentrate on multi-input/output transactions that

Input no \ Output no	1 output	2 outputs	$n_O \geq 3$ outputs
0 input	✓ mining	✓ mining	✓ mining
1 input	1 payment or a relay ✓ † (relay transaction)	2 payments or 1 payment with change ✓ † (peel transaction)	n_O payments or $n_O - c$ payments with c changes to own wallets ✓ (distribution)
$n_I \geq 2$ inputs	consolidation of fund either to own wallet or for payment ✓ † (sweep transaction)	payment with or without change † (peel transaction)	multi-input multi-output (mixing/distribution)

TABLE 1 A typology of Bitcoin transactions based on their input/output configurations and traceability of Bitcoin flows: checkmarks ✓ denote configurations such that the Bitcoin flow is traceable, while † refers to a definition introduced in¹⁷.

are tied up to each other. More precisely, we construct a subgraph of the transaction networks, whose nodes all are multi-input/output transactions. The connectivity of this subgraph contains information regarding the relationships among multi-input/output transactions, which we extract in terms of connected components and longest paths. Using as criteria in-/out-degrees, in-/out-degree (normalized-)entropies and belonging to the same connected components, we single out clusters of transactions which are outstanding with respect to mixing-like features. In Section 4, the wallet address network characteristics are compared before and after address agglomeration. We notice that the 3rd and 4th chosen periods stand out with respect to most of the characterizations considered for transactions, such as: extreme outliers in terms of transactions with high output number (in Subsection 3.1), amount spent in multi-input/output transactions and amount of large in- and out-degree nodes with high normalized entropies (in Subsection 3.2), and longest paths within multi-input/output subgraphs (in Subsection 3.3). Similar distinctive behavior for the 3rd and 4th period is also seen to be true for the address networks, in terms of several studied properties such as Gini or Pearson coefficients.

2 | BITCOIN MIXING

A Bitcoin transaction consists of a set of inputs, and a set of outputs: each transaction input contains a pointer to a previous transaction (output), with an associated wallet address (which will have the authority to spend the Bitcoins), while each output contains a Bitcoin amount, and the wallet address receiving the Bitcoins (more precisely, it contains a script that checks the public key and digital signature of the recipient.), see e.g.¹⁶ for details. The number of outputs is at least 1: it is not possible to have 0 output, however it is possible (but not usual) to have an output with 0 amount. It is possible however to have 0 inputs, this indicates a mining operation, irrespectively of the number of outputs.

How specific Bitcoins flow from individual inputs to individual outputs within one transaction may or not be deterministically inferred, depending on the input/output configuration. In¹⁷, a summary of nomenclature has been proposed for some of the configurations, which we refer to in Table 1, while specifying a typology of the transactions based on their input/output configurations and traceability of Bitcoin flows. Checkmarks ✓ in the table indicate that one can deterministically trace the individual Bitcoin's flow based on information locally available from only the specific transaction, while † represents the transactions where the heuristics from^{13,14} are applied for wallet address aggregation (discussed in detail in Subsection 4.2). For transactions with 2 or more inputs and more than 2 outputs, it may not always be possible to deterministically ascertain which specific input contributes to which specific output(s). Such transactions indicate several possibilities: A single entity controls all the input addresses, which is either making multiple payments, or just moving around the Bitcoins to other addresses belonging to itself, or making some payments, and moving remaining of the Bitcoins under its own control (all of these fall under the category of 'distribution' as per the terminology in¹⁷), or it may also be that multiple independent entities are combining their transactions (be it in a decentralized manner, or using a centralized facilitator) together to further obfuscate the flow of Bitcoins (commonly called 'mixing'). Often, though not necessarily, when there are two outputs, one of them is used to collect the change. Thus, we refer to all the transactions with 2 or more inputs and more than 2 outputs as multi-input multi-output (multi-input/output) transactions.² In general, it may not be possible to use just the information available from solely the transaction to associate specific inputs to particular outputs in multi-input/output transactions (though depending on the actual amounts involved, it might be possible in certain instances). Multi-input/output transactions may be created due to mixing when several distinct entities try to obfuscate the flow of their Bitcoins and thus real relationships among different wallet addresses. However, multi-input/output transactions may also be created without inputs from multiple entities. In the absence of any deterministic indicator of mixing, we consider multi-input/output transactions as a proxy (and

²If several inputs to a transaction are from a same wallet address, we count them together as '1'. Likewise for outputs.

FIGURE 2 Number of multi-input/output transactions in the blockchain (3 Jan 2009 - 14 Apr 2018) with zoom in of 3 periods of ~ 50 days.

FIGURE 3 Fraction of transactions that are multi-input multi-output.

an upper bound) for estimating mixing operations, and accordingly study these, and their effect on the properties of the networks that can be inferred from the Bitcoin transaction logs.

Figure 2a shows the number of multi-input/output transactions per day for the Bitcoin blockchain for the period from 3 Jan 2009 (UTC+0000) to April 14 2018 (UTC+0000). Note that the genesis block was mined on 3rd January 2009, but Bitcoin was publicly released on 9 January 2009, making 9 January the day 2 in our data. Since the total number of transactions tends to increase over time, we also report the ratio of multi-input/output transactions per total number of transactions in Figure 3a, per day. This shows that the percentage of multi-input/output transactions remains little over time (it was always less than 6%, and often much less than that for the studied period), even though the absolute numbers have been as high as close to 10000 in a day.

Mixing services (or tumblers) were introduced to mix one's Bitcoins with others', so as to obfuscate the connection between inputs and outputs, in exchange for a percentage fee. They artificially reproduce the phenomena explained above: instead of having naturally different inputs joining a transaction to send Bitcoin to different outputs, resulting in a lack of traceability, tumblers are collecting input transactions from different wallet addresses and possibly uniformizing amounts in such a way that one cannot trace anymore the link between inputs and outputs. Examples of such services include Bitcoin Laundry, BitMix, BestMixer.io, Bitcoin Blender, CryptoMixer, Bitcoin Fog, Blender.io, MixTum.io, to name a few. Third party free decentralized approaches have also been designed to mix coins, for instance, CoinJoin (reported on the Bitcoin Forum at <https://bitcointalk.org/?topic=279249> on August 22 2013). ZeroCoin became active in the late 2012/early 2013 (this is extrapolated based on the fact that the ZeroCoin paper appeared in May 2013 as per http://zerocoin.org/talks_and_press).

To get a sense of when mixing became popular, we show in Figure 4 the results given by Google trends for the searches "bitcoin tumbler" and "bitcoin mixer". The terms appeared already in 2011, but started to become popular in the late 2013, early 2014, with a peak of popularity at the end of 2018.

We now go back to the (normalized) number of multi-input/output transactions over time, and zoom into four periods:

FIGURE 4 The search occurrences of the words “bitcoin tumbler” and “bitcoin mixer” as per Google trends, which normalizes by the maximum number of searches during the period considered.

FIGURE 5 For the 4 periods considered, a summary of the transaction inputs/outputs (all transactions in the dual mode networks are included).

- From 8th December to 26 January 2011 (days 700-749): these are still early days of the Bitcoin network, with no instances of multi-input/output transactions. This is also a period before when we could locate any mention of mixing.
- From 6th April to 24 May 2013 (days 1550-1598): this period is within the very early days of mixing, close on the heels of early references to the first tumblers and the notion of mixing appearing in the public domain.
- From 8 October to 25 November 2014 (days 2100-2148): this is a period of popularity for the terms tumbler and mixer, during which the observed percentage of multi-input/output transactions was also particularly high, as can be seen on Figure 3b.
- From 19 July to 6 September 2016 (days 2750-2799): during this period, the absolute number of multi-input/output transactions is high (see Figure 2b), but the normalized number is moderate (see Figure 3b).

We chose periods of ~ 50 days for the periods to be long enough to exhibit meaningful statistical properties, while being not too long where the proportion of multi-input/output transactions with respect to overall transactions vary too much, in which case a subpart of the period might then have had a dominating influence on the statistics w.r.to other parts of the same period.

For each period, we used the dual mode network and also extracted both the corresponding transaction graph and wallet address graph. We note that the graph sizes vary significantly over time, because the number of transactions per unit time, and correspondingly the rate of growth of the blockchain has been increasing as Bitcoin has been becoming popular. Transactions and transaction graphs will be analyzed in the next section, followed by our studies of the wallet address graphs and agglomerated address graphs in Section 4.

FIGURE 6 A summary of the transaction inputs/outputs per period (all transactions in the dual mode networks are included).

3 | TRANSACTION ANALYSIS

We start our analysis by looking more closely at transaction parameters, that is at the transaction nodes within the dual mode graphs. In the previous section, we observed that the number of multi-input/output transactions does increase over different periods, but remains a small proportion of the overall transaction amount. We reiterate that not every multi-input/output transaction occurs because of mixing, but mixing leads to multi-input/output transactions.

3.1 | Inputs/Outputs Correlation Analysis

We thus look at the number of inputs/outputs of transactions, that is, the in-degree/out-degree of transaction nodes in the dual mode network. The statistics are summarized in Figures 5a and 5b in the form of boxplots. Boxplots display quartiles in boxes, around the median, and outliers are represented by round dots. The figures show an overwhelming range of outliers that make the boxes (that is, all non-outliers) collapse to one bar close to 0. This tells that the majority of transactions have very low in-/out-degrees, something that we already know given the low percentage of multi-input/output transactions. However, the three periods from the beginning of mixing (days 1550-1598, 2100-2148, 2750-2799) have all instances of high in-/out-degrees. The in-degrees are mostly up to ~ 750 for periods 2, 3 and 4, with 3 extreme outliers, one around 2000 in period 2 and two around respectively 1250 and 1500 in period 4. It is less homogeneous for the out-degrees. In fact, the 3rd period which has the highest ratio of multi-input/output transactions also has significantly higher out-degrees, with extreme outliers beyond 5000.

This naturally raises the questions of correlation between high in- and out-degrees. To get a sense of whether nodes tend to have both high degrees, we show the scatter plots (with in/out-degrees represented in x/y-axis respectively) in Figure 6, where we focus on multi-input/output transactions (therefore Period 1 is not displayed). There are a few immediate observations:

FIGURE 7 Transaction amounts (the Bitcoin (BTC) exchange rate is provided on the left for context).

(1) Though there were many nodes with very high in-degree, and many nodes with very high out-degree (as observed in 5), there were no nodes with both very high in-/out-degrees (such nodes, if present, would have populated the upper right corner of the scatter plots), and mostly, when one of the degrees is very high, the other degree is very low (typically 2 or 3).

(2) In fact, most nodes tend to have unbalanced degrees, nodes with both high degrees are sparse: for a threshold of 50 for both degrees, there are only 5 of them in Period 2, and 10 of them in Period 3. Period 4 has a larger number of 132, due to an accumulation of nodes with out-degrees around 60 (74 of them).

(3) This could suggest that if mixing is balancing the number of inputs and outputs, then it does so with relatively small numbers of inputs and outputs that do not stand out.

3.2 | Multi-output Spending Characterization

Now that we have a sense of the behavior of in-/out-degrees for transaction nodes, we look at what Bitcoin amounts have been dealt through these transactions.

The amount of transactions in Bitcoins for both multi-input/output and other transactions are shown in Figure 7. The y-axis is in logscale. It is not intuitively clear whether we should expect largest amounts in one category of amount, especially taking into account that the value of the Bitcoin currency has been fluctuating significantly over time. The boxplots are showing that actually, there are many more smaller amounts dealt with through multi-input/output transactions. The first thing to keep in mind is that we are looking at ranges of time during which the exchange rate of Bitcoin has varied considerably. We reproduce the data available from <https://coinmarketcap.com/currencies/bitcoin/historical-data/> (the exchange rate is in USD), using the highest value per day. There is a factor of roughly 7 between the lowest and the highest rates.

We compare in Figure 7 the amount in Bitcoins via multi-input/output transactions on 7b versus the amount in other transactions. The observations are arguably counter-intuitive:

(1) When the price of the Bitcoin increases, one would expect that the amount dealt with would reduce. For multi-input/output transactions, the median does decrease from Period 2 to 3, it barely reduces from Period 3 to 4, at the same time, the minimum does increase. For other transactions, the median hardly changes over Periods 2 to 4, in fact it increases a bit over Period 4. For the last two periods, one could hypothesize that users built confidence over time, and therefore increased the amount of money they converted into Bitcoin. For the first two periods, this is not likely to be explained by the number of input/outputs involved across time periods (see Figure 6).

(2) Multi-input/output transactions have by definition more inputs/outputs than other transactions. However the median for the overall amount of Bitcoins involved in multi-input/output transactions does not appear to be significantly higher than for other transactions.

We next zoom further into the Bitcoin amounts and in-/out-degrees of transactions, by looking at the Bitcoin amounts transiting through each edge of each multi-input/output transaction. For a given multi-input/output transaction t , let b_e be the bitcoin amount on the edge e , where e could be either an incoming or outgoing edge of t . We then define for e an incoming edge

$$p_{e,in} = \frac{b_e}{\sum_{e' \in E_{t,in}} b_{e'}}$$

FIGURE 8 Entropy of transactions as per incoming/outgoing Bitcoin flows.

which is the Bitcoin amount on this edge, normalized by the total of Bitcoin amounts reaching t via the set $E_{t,in}$ of incoming edges of t , and similarly for outgoing edges:

$$p_{e,out} = \frac{b_e}{\sum_{e' \in E_{t,out}} b_{e'}}.$$

Since $p_{e,in}$, $p_{e,out}$ respectively satisfy the axioms of probability, namely, $0 \leq p_{e,in} \leq 1$, $0 \leq p_{e,out} \leq 1$, and

$$\sum_{e \in E_{t,in}} p_{e,in} = \sum_{e \in E_{t,in}} \frac{b_e}{\sum_{e' \in E_{t,in}} b_{e'}} = 1, \quad \sum_{e \in E_{t,out}} p_{e,out} = \sum_{e \in E_{t,out}} \frac{b_e}{\sum_{e' \in E_{t,out}} b_{e'}} = 1,$$

they are interpreted as probabilities over $E_{t,in}$ and $E_{t,out}$. We can then use (Shannon) entropy to determine how distinct the Bitcoin amounts are (the less distinct they are, the harder it would be to apply any statistical analysis to correlate relations between inputs and outputs, and the better the effect of obfuscation would be), given by:

$$H_{in}(t) = - \sum_{e \in E_{t,in}} p_{e,in} \log_2 p_{e,in}, \quad H_{out}(t) = - \sum_{e \in E_{t,out}} p_{e,out} \log_2 p_{e,out}.$$

We note that when $p_e = \frac{1}{|E|}$ for every $e \in E$ (whether e is an incoming or outgoing edge)

$$- \sum_{e \in E} p_e \log_2 p_e = \frac{1}{|E|} \sum_{e \in E} \log_2 |E| = \log_2 |E|,$$

	T ₂ (period 2)	T ₃ (period 3)	T ₄ (period 4)
whole graph	V ₂ = 2771274 E ₂ = 5638486	V ₃ = 3947992 E ₃ = 8790161	V ₄ = 11050062 E ₄ = 20864595
multi-input/ output nodes	88938	350431	300290
multi-input/ output graph	V ₂ = 81040 E ₂ = 121675	V ₃ = 324743 E ₃ = 1035632	V ₄ = 253808 E ₄ = 381035
no of cc	2837	6347	13708
largest cc	65251	301770	176181
longest path	774	64728	5343

(a) A summary of basic graph characteristics, including connected components (cc).

(b) A connected component from the multi-input/output graph, with its longest path highlighted in blue.

FIGURE 9 Transaction graphs & multi-input/output subgraphs.

and since the entropy measures uncertainty, the highest uncertainty, given the number of edges $|E|$, is reached when the distribution is uniform, in which case it is $\log_2 |E|$.

We display the pairs $(H_{in}(t), H_{out}(t))$ for every transaction t in the three usual periods in Figure 8. We note that some transactions have an output of 0. These outputs are not used in the entropy computations (more precisely, if $b_e = 0$, then $p_e = 0$ and as per the usual convention, $p_2 \log_2 p_2 = 0$). There are 119 transactions in Period 3 and 1934 transactions in Period 4 which have at least one such an output of 0.

Each of the three plots in Figure 8 follow the same convention: the pairs $(H_{in}(t), H_{out}(t))$ are shown inside each period, and circled are the transactions that were highlighted in Figure 6d for having both high in- and out-degrees. High degrees for Periods 2, 3, and 4 means degrees higher than 50, 50 and 61 respectively.

Given the number of edges, the highest possible entropy corresponds to a uniform distribution, which would be $\log_2(d^{in}(t))$ or $\log_2(d^{out}(t))$ for in-/out-degrees of $d^{in}(t)$ or $d^{out}(t)$ respectively. Thus, just having a high degree itself can lead to a high entropy in absolute. However, given a specific degree, we also want to investigate if there are certain transactions which have a high entropy subject to the maximum possible given the degree, because that is an indicator of how uniform the values are, and thus, how difficult it is to carry out any statistical analysis. So to say, we are looking for transactions t such that $(H_{in}(t), H_{out}(t))$ takes value close to $(\log_2(d^{in}(t)), \log_2(d^{out}(t)))$ where $d^{in}(t)$, $d^{out}(t)$ are respectively the in- and out-degree of t . We thus compute the normalized entropy (with respect to maximum possible entropy given the degrees) as

$$H_{in,norm}(t) = \frac{H_{in}(t)}{\log_2(d^{in}(t))}, \quad H_{out,norm}(t) = \frac{H_{out}(t)}{\log_2(d^{out}(t))}.$$

The color schemes in Figure 8 reflect the values of $H_{in,norm}, H_{out,norm}$. The darkest points are the pairs such that $(H_{in,norm}, H_{out,norm}) > (0.999, 0.999)$. The lighter points are pairs such that $(H_{in,norm}, H_{out,norm}) < (0.999, 0.999)$. The grid like pattern of these very high normalized entropy suggest that there are certain (generally low) specific in-/out-degrees that are used while keeping the amount of Bitcoin per edge equal, arguably to thwart statistical analysis, even as a much wider range of in-/out-degrees are in use in general. Given that a high absolute entropy in itself may make statistical analysis difficult, it is possible that many of the transactions which do not spread the amounts in a manner to maximize the entropy for a choice of degrees are also mixing operations. However, the transactions which maximize the normalized entropy (even with smaller absolute degrees and corresponding absolute entropy values) indicate a very deliberate and careful approach to obfuscate the flows, and suggest mixing.

3.3 | Multi-input Multi-output Connectivity Analysis

We next would like to study the connections among multi-input/output transactions. For that, we start by extracting transaction networks from dual mode networks, as explained in the introduction. This gives the 3 directed graphs T_2, T_3, T_4 , whose basic statistics are summarized in Figure 9a under the entry "whole graph". The nodes comprise both multi-input/output nodes and other nodes. The number of multi-input/output nodes is also reported in the Table. Out of each of these graphs T_2, T_3, T_4 , we then extract multi-input/multi-output subgraphs, by retaining all the multi-input/output nodes and their edges (and in particular those nodes that are not connected to other multi-input/output nodes become isolated and are removed from the subgraphs, which explains the differences in the table between numbers of multi-input/output nodes, and cardinalities of vertices in the multi-input/output graphs).

The connectivity of the newly created subgraphs is studied in order to understand how multi-input/output transactions are connected among each others. This is of interest in the mixing perspective, since one could expect that sensitive Bitcoin flows would go through several hops of

FIGURE 10 High degrees/high normalized entropy nodes within their connected components (and longest paths).

mixing. Two basic measures of connectivity are the number of connected components, and the size in nodes of the largest connected components. Both are tabulated in Figure 9a. Since the graphs are directed, we would like to emphasize that by connected components, we mean the connected components of the underlying undirected graphs, and not strongly connected components of the directed graph. Our interpretation of connected components captures multi-input/output transactions that are involved with each other, but not necessarily consecutively. Next, we look at directed paths within each such connected component obtained, in particular at the longest ones. An illustration of the process is available in Figure 9b. This subgraph is a connected component of a subgraph of T_2 seen as undirected. It contains 35 nodes. The blue nodes represent a longest path found in this subgraph, seen as directed. The lengths of the longest paths within the largest connected component are presented in Figure 9a. Period 3 stands out as usual for having a particularly long path within its largest component.

Extracting a multi-input/output transaction network out of a transaction network, then zooming into its connected components and its longest paths, gives different levels of granularity at which we study the connections among multi-input/output transactions. Later in Section 4 we further discuss the statistics of how multi-input/output transaction outputs are spent, or inputs derived in/from other multi-input/output transactions, to complete this study of cascading multi-input/output transactions.

We next would like to see how the transactions identified earlier as interesting in terms of degrees and/or normalized entropies behave through these different lenses.

FIGURE 11 The subgraph of Figure 10c inside the dual mode network. The last 2 characters of each transaction hash or wallet address are given inside each node for identification.

For Period 2, we consider transactions t such that either $d^{\text{in}}(t), d^{\text{out}}(t) > 30$ (high in-/out-degrees), or $d^{\text{in}}(t), d^{\text{out}}(t) > 3$ but with $H_{\text{in}, \text{norm}}(t), H_{\text{out}, \text{norm}} > 0.99$ (very high normalized entropies). There are 24 of them, and 14 of them belong to the largest connected component. There are 2 components that contain 2 of them, the other 6 transactions are each in one component, which also means that only 8 of the 2837 components contain transactions satisfying these conditions. We plot the two components containing 2 outstanding transactions in Figure 10a and 10b. These are small connected components of respectively 7 and 20 nodes. We emphasize that the high degrees and normalized entropies are computed in the dual mode network, they measure the number of Bitcoin wallet addresses and the distribution of Bitcoin amounts associated with the transactions, while in this representation we only show the connection among transactions. We note that one component contains two nodes of high degrees, the other contains two nodes of high entropies. The 14 nodes of interest in the largest component are not plotted as such, since the component contains 65251 nodes. We thus looked at whether these 14 nodes are directly connected among each other, and we found that only 3 of them are connected (forming a directed path of length 3).

For Period 3, we consider transactions t such that either $d^{\text{in}}(t), d^{\text{out}}(t) > 30$, or $d^{\text{in}}(t), d^{\text{out}}(t) > 5$ but with $H_{\text{in},\text{norm}}(t), H_{\text{out},\text{norm}} > 0.99$. We increased the degree requirement for high normalized entropy nodes, since we have many more nodes than for Period 2. There are 763 of them, and since we have a giant component of 301770 nodes, it is probably not surprising that most of them are on it. What is more surprising is that in fact 756 out of 763 are in the giant component (out of which 492 belong to the longest path of the giant component), 3 are in a different component, and we have 4 nodes each in separate components. We show in Figure 10c the component containing 3 nodes of interest, all of them turn out to be high normalized entropy nodes. We also see that the longest path covers all the nodes in the whole component. This creates an interesting cluster of multi-input/output transactions, where we observe Bitcoin amounts flowing from one multi-input/output transaction to another for up to 21 hops, including 3 high normalized entropy nodes that are at the beginning of the chain. Furthermore, we notice multiple flow confluence artifacts that have been hypothesized as indicators of coordination^{5,12}. Now for the nodes of interest in the giant component, we again look at how many are actually directly connected to each other. We already observed they are on the same path, but this path contains other nodes which are not our focus.

FIGURE 12 Cumulative time to spend a transaction output: NM means no multi-input/output, M means multi-input/output transaction. The three categories ('All' being inclusive of both) are normalized w.r.to the total number of transactions of said category (i.e, the Y-axis indicates the fraction of transactions of each category).

The situation is a bit different for Period 4. We consider transactions t such that either $d^{\text{in}}(t), d^{\text{out}}(t) > 30$, or $d^{\text{in}}(t), d^{\text{out}}(t) > 50$ with $H_{\text{in},\text{norm}}(t), H_{\text{out},\text{norm}} > 0.99$, but note that this time we look at degrees more than 50 for high normalized entropy nodes (if we had chosen a lower threshold, many more nodes would be shortlisted, making the following illustrative discussions less tractable). This yields 454 nodes, but unlike for the previous periods where the nodes were concentrated over a few components, this time they are spread over more than 50 connected components (Period 4 does have 13708 components, there are 6347 for Period 3 and 2837 for Period 2. Furthermore, we have chosen a much larger threshold, which too leads to such spread across components.). Out of these 454, 242 nodes are found in the largest component. Nodes ranging in numbers from 1 to 15 are found in the other components, with the exception of one component which contains 28. It is displayed in Figure 10e. It is an interesting component because it forms a single directed path of 43 transactions, out of which 28 are of high degrees. We notice that several of them are either following each others, or at distance 2 from each others. For the 242 nodes inside the giant component, we plot as we did for Period 3 the graph of their connections in Figure 10f.

The method described above identified a series of transaction clusters which are of interest, namely multi-input/output transactions with high degrees and high normalized entropies, and which are interacting with each others. Once identified, they could be studied individually for further details, such as looking at confluence of flows to establish relationships among wallet addresses coordinating activities amongst themselves^{5,12}. In Figure 11 we display for example the subgraph of Figure 10c inside the dual mode network. This provides a visualization that enables us to see how many edges are in between each pair of transaction nodes, and which are the wallet addresses involved.

3.4 | Time Patterns

We conclude this section by exploring the spending time patterns involved in multi-input/output transactions. In Figure 12, we look at the time (in minutes) taken for Bitcoin amounts to be spent. A priori we expected that Bitcoin transactions that are being mixed would be spent much faster than usual transactions. Out of the 4 periods we are looking at, period 3 again stands out: during this period, Bitcoin amounts spent via multi-input/output transactions are spent faster than for other transactions. This is unlike the other periods, where amounts in multi-input/output transactions are typically spent more than or equally slowly as other transactions.

graph	vertices	edges	clustering coefficient
G ₁ (period 1)	V ₁ = 55295	E ₁ = 76501	$\mathcal{C}(G_1) = 0.03855$
G ₂ (period 2)	V ₂ = 2572361	E ₂ = 9628675	$\mathcal{C}(G_2) = 0.10818$
G ₃ (period 3)	V ₃ = 6643301	E ₃ = 43412201	$\mathcal{C}(G_3) = 0.03304$
G ₄ (period 4)	V ₄ = 12924336	E ₄ = 45487349	$\mathcal{C}(G_4) = 0.05039$

TABLE 2 Size of the wallet address graphs and their clustering coefficients (treated as undirected graphs).

4 | ADDRESS GRAPHS ANALYSIS

4.1 | Analysis of the Original Wallet Address Graph

We next consider the wallet address graphs for the four periods, respectively called G_1, G_2, G_3, G_4 . Recall that the wallet address graph is created by adding a directed link from each address which is an input to a transaction to each address which is an output of that transaction. The sizes of these four graphs are found in Table 2.

To give some perspective on the overall growth of the Bitcoin network,³ reported 13'086'528 wallet nodes with 44'0320'115 edges for the period from January 2009, when the Bitcoin network came online, until mid-2013. The overall Bitcoin address graph for that period was studied in³, where in particular the in-/out-degree distributions of the nodes were analyzed in terms of power-law. We start by repeating this power-law analysis for the 4 graphs G_1, G_2, G_3, G_4 of interest.

We recall that a power-law distribution is a probability distribution given by

$$p(x) = Cx^{-\alpha}, \quad x \geq x_{\min}$$

where C is a normalization constant, such that, in the discrete case, $\sum_{x \geq x_{\min}} p(x) = C \sum_{x \geq x_{\min}} x^{-\alpha} = 1$. For the continuous case, $\int_{x_{\min}}^{\infty} p(x) dx = C \int_{x_{\min}}^{\infty} x^{-\alpha} dx = C \frac{x_{\min}^{-\alpha+1}}{-\alpha+1}$, assuming that $\alpha > 1$ (if $\alpha \leq 1$, the exponent $-\alpha+1$ is positive and this quantity is infinite), from which $C = (\alpha-1)x_{\min}^{\alpha-1}$.

The m th moment of a random variable X distributed according to $p(x)$ is given by $\langle X^m \rangle = \sum_{x \geq x_{\min}} x^m p(x)$, or $\langle X^m \rangle = \int_{x_{\min}}^{\infty} x^m p(x) dx = C \frac{x_{\min}^{-\alpha+1+m}}{-\alpha+m-1}$ assuming that $\alpha > m+1$, which simplifies to $x_{\min}^m \frac{\alpha-1}{\alpha-m-1}$. The mean is given by $m = 1$.

A power-law (or scale-free) directed graph on n vertices is a graph whose in-degrees $d_1^{\text{in}}, \dots, d_n^{\text{in}}$ and out-degrees $d_1^{\text{out}}, \dots, d_n^{\text{out}}$ are realizations of the random variables $D^{\text{in}}, D^{\text{out}}$ which follow the respective power laws:

$$\begin{aligned} p_{\text{in}}(x) &= C_{\text{in}} x^{-\alpha_{\text{in}}}, \quad x \geq x_{\min}^{\text{in}}, \\ p_{\text{out}}(x) &= C_{\text{out}} x^{-\alpha_{\text{out}}}, \quad x \geq x_{\min}^{\text{out}}. \end{aligned}$$

In³, it was computed over one month periods that from late 2011 until mid 2013, the in-degree and out-degree distributions do not change significantly (it is claimed that the network measures converge to their typical value by mid-2011) and are approximated by power-laws $p_{\text{in}}(x) \sim x^{-2.18}$ and $p_{\text{out}}(x) \sim x^{-2.06}$.

We provide a power-law fitting for both in- and out-degrees of the graphs G_1, G_2, G_3, G_4 as shown in Figure 13 for $x_{\min} = 1$. We notice that for G_1 , $\alpha_{\text{in}} = 2.1960196$ which is indeed consistent with the computations of³, though there is already some variation for $\alpha_{\text{out}} = 1.8922812 < 2.06$ (period 1 is before the stabilization period of mid-2011). For G_2 and G_4 , both exponents are in the interval 1.63 to 1.85, while G_3 stands out with the lowest exponents for both in-degrees and out-degrees.

The global clustering coefficient $\mathcal{C}(G)$ of a graph G is determined as the ratio of the total number of closed triplets (where three nodes form a closed triangle in an undirected graph) to the total number of closed and open triplets (when three nodes are connected among themselves with two undirected edges). In Table 2, we report the clustering coefficients of the undirected version of the four wallet address graphs. In³, the observed clustering coefficient was around 0.15 in the early phase (for which the authors speculated 'a few enthusiasts trying out the Bitcoin system by moving money between their own addresses'), but subsequently it was observed to stabilize to 0.05 when Bitcoin came to be in more mainstream usage (and it was reported that a random network with similar degree sequence would have significantly lower clustering coefficient of around 0.0037). We observe the clustering coefficients for the four periods to be similarly much larger than what was mentioned for random networks, but it also fluctuates over a larger range between $\approx 0.033 - 0.108$ (which includes the 0.05 value mentioned in³). From the reading of³ it is ambiguous whether the reported clustering coefficients were for the entire cumulative network, or were determined independently for separate time windows, so we cannot comment on whether the behavior has changed again over time since their study. But, the significant difference of the clustering coefficient across the four studied periods, and the particular observed values convey two interesting information. First, in the early phase of multi-input/output transactions (Period 2), we see a clustering coefficient higher than 0.1, suggesting that even if many people were likely using mixing,

FIGURE 13 Fitting of power law for the 4 periods.

they did so somewhat indiscriminately, so that there were many directly identifiable interactions among their wallet addresses. In contrast, in Period 3 (when the relative frequency of multi-input/output transactions were in fact highest), we observe the lowest clustering coefficient, suggesting that those carrying out such transactions were extra cautious. This aligns well with the speculation that the relative frequency of transactions were higher precisely because many people were involved in careful mixing. In Period 4, the clustering coefficient rises again, close to 0.05, which can be attributed to the larger number of multi-input/output transactions in absolute, and also possibly for this number to settle back to the value 0.05 observed in³. Next, we compute the Gini coefficient, which is a statistical measure to determine inequalities in a distribution. Every wallet address graph G_i has $|V_i|$ vertices. For each set V_i of vertices, we compute the in-degree and out-degree of each vertex, and list all the in-degrees d_j^{in} by increasing order, so that $d_j^{\text{in}} \leq d_{j+1}^{\text{in}}$, and similarly all the out-degrees d_j^{out} so that $d_j^{\text{out}} \leq d_{j+1}^{\text{out}}$.

For every graph G_i , $i = 1, \dots, 4$, we compute the Gini coefficients

$$\mathcal{G}^{\text{in}}(G_i) = \frac{2 \sum_{j=1}^{|V_i|} j d_j^{\text{in}}}{|V_i| \sum_{j=1}^{|V_i|} d_j^{\text{in}}} - \frac{|V_i| + 1}{|V_i|}, \quad \mathcal{G}^{\text{out}}(G_i) = \frac{2 \sum_{j=1}^{|V_i|} j d_j^{\text{out}}}{|V_i| \sum_{j=1}^{|V_i|} d_j^{\text{out}}} - \frac{|V_i| + 1}{|V_i|}$$

for the in-degree and out-degree distribution respectively.

We note that $\mathcal{G}(G_i) = 0$ happens exactly when $d_j = d$ for all j , meaning that all nodes have the same in-/out-degrees. On the other hand, $\mathcal{G}(G_i) = 1$ corresponds to having one node connecting to all other nodes, and all the other nodes having degree 0 (this would be a directed star network). A low Gini coefficient is then interpreted as having a homogeneous in-/out-degree distribution, while a high Gini coefficient (close to 1) is a sign of a graph with very heterogeneous degrees. The Gini coefficients for the address graphs are reported in Table 3. They show that the graph G_1 is the most homogeneous, especially in terms of out-degrees. This means that in the first period, wallet addresses are all having interactions with overall similar numbers of partner addresses, a behavior that changes over time. From period 2 on, it seems that small and large Bitcoin players are more prominent. Period 3 is again distinguished with the most skewed in-degree distribution, though the out-degree distribution is similar to that in other periods.

For comparison,³ reported $\mathcal{G}^{\text{in}} \approx 0.629$ and $\mathcal{G}^{\text{out}} \approx 0.521$ for the period between mid-2011 to mid-2013. With respect to that period, the latter periods (2-4) that we study indicate that the skew in the degree distributions among wallet addresses have increased. This amplification in skew can in part be attributed to the presence of the multi-input/output transactions, which leads to nodes with large number of in-/out-degrees (particularly given the manner the address network is constructed).

When the in-/out-degrees follow a power-law distribution with exponent $\alpha \geq 2$, it is known³ that the corresponding Gini coefficient is $\frac{1}{2\alpha-3}$. Thus $\alpha = 2$ which is the lowest possible value of α gives a Gini coefficient of 1, the most "unfair" extreme. When α grows $\frac{1}{2\alpha-3}$ tends to 0, thus

$\mathcal{G}^{in}(G_1)$	0.5337604	$\mathcal{G}^{out}(G_1)$	0.3346766	G_1	$r = -0.0559628$
$\mathcal{G}^{in}(G_2)$	0.6244982	$\mathcal{G}^{out}(G_2)$	0.6214836	G_2	$r = -0.0578501$
$\mathcal{G}^{in}(G_3)$	0.7048792	$\mathcal{G}^{out}(G_3)$	0.6457142	G_3	$r = -0.0100263$
$\mathcal{G}^{in}(G_4)$	0.6668730	$\mathcal{G}^{out}(G_4)$	0.6315119	G_4	$r = -0.0275603$

TABLE 3 Gini coefficients for the in-/out-degrees of the wallet address networks, and Pearson coefficients for the in-/out-degrees of the wallet address networks.

Period	Combinations in-out	$M_o \rightarrow M_i$	$M_i \leftarrow M_o$	$nM_o \rightarrow M_i$	$nM_i \leftarrow M_o$
Period 2		0.411	0.464	0.041	0.040
Period 3		0.484	0.589	0.074	0.091
Period 4		0.321	0.366	0.038	0.042

TABLE 4 Statistics on co-mingling of (non-)multi-input/output transactions.

converging towards a uniform distribution. This shows that small values of α yield more heterogeneous degree distributions. However only α_{in} for G_1 is slightly more than 2, and it thus can be used to explain a very heterogeneous in-degree behavior for that case.

The Pearson correlation coefficient r of the out-/in-degrees of connected node pairs is given by

$$r = \frac{\sum_{i=1}^{|V|} (d_i^{out} - \overline{d^{out}})(d_i^{in} - \overline{d^{in}})}{|V|\sigma_{in}\sigma_{out}}, \quad \overline{d^{out}} = \sum_{i=1}^{|V|} d_i^{out}/|V|, \quad \overline{d^{in}} = \sum_{i=1}^{|V|} d_i^{in}/|V|, \quad \sigma_{in}^2 = \sum_{i=1}^{|V|} (d_i^{in} - \overline{d^{in}})^2/|V|.$$

We see (Table 3) that the Pearson coefficient is negative and very small - which conforms with the results observed in³ where the values mostly ranged between -0.01 and -0.05, reasonably similar to our study. Thus, presence of mixing does not seem to change the overall characteristic of the graph. A negative value typically indicates a disassortative network, however the very small absolute value (close to zero) suggests that the out-/in-degrees are for all practical purposes uncorrelated.

Finally, for the wallet address networks, we want to study how the inputs/outputs of multi-input/output and non-multi-input/output transactions co-mingle. To that end, for the three relevant periods, we report in Table 4 the fraction of multi-input/output transactions' outputs (and respectively non-multi-input/output transactions' outputs) that are used as input for other multi-input/output transactions $M_o \rightarrow M_i$ (and respectively $nM_o \rightarrow M_i$) and the fraction of multi-input/output transaction inputs (and respectively non-multi-input/output transactions' inputs) that used outputs of multi-input/output transactions $M_i \leftarrow M_o$ (and respectively, $nM_i \leftarrow M_o$). 'Round half up' convention has been used for presenting these numbers. Corresponding values for the complementary events, for instance, the fraction of multi-input/output transactions' outputs that are used as inputs for non-multi-input/output transactions $M_o \rightarrow nM_i = 1 - (M_o \rightarrow M_i)$ can be inferred from the four reported values. From the first two columns in the table, we infer that a significant number of multi-input/output transactions outputs go into other multi-input/output transactions, and likewise, a significant portion of their inputs come from multi-input/output transactions. This statistics suggests the presence of cascades of multi-input/output transactions, something we observed from the subgraph analysis of multi-input/output transaction networks in Section 3.

In absolute, we observed 88938, 350431 and 30290 multi-input/output transactions for the Periods 2, 3 and 4 respectively. Only a small fraction of all transactions are multi-input/output transactions (as can be inferred from the total number of transactions for the period, tabulated in Figure 9a), and as was depicted explicitly previously in Figure 3. This explains the very low values for the last two columns in Table 4, i.e. non-multi-input/output transactions are predominant, thus their inputs (and outputs) are predominantly from (and to) other non-multi-input/output transactions.

4.2 | Analysis of the Network of Aggregated Wallet Addresses

A same entity may control several Bitcoin wallet addresses. It may also consolidate the Bitcoins in these addresses, use them jointly to pay, collect changes from payment, and so forth. These interactions among multiple wallets controlled by the same entity can be leveraged to aggregate them together. Some of the common heuristics^{13,14,17} for address aggregation (which we apply for the current study³) are as follows. For transactions

³The cited related works have certain minor variations amongst themselves. We thus specify the particular set of heuristics used in this work.

FIGURE 14 Histogram of the size of groups (x-axis) of aggregated wallet addresses.

with multiple inputs and one or two outputs, the addresses corresponding the multiple inputs are aggregated together. Furthermore, for such transactions, if there is only one (distinct address) output then the corresponding address is clubbed with the input addresses, while if there are two (distinct address) outputs, then the address receiving lower of the two amounts is clubbed with the input addresses. If the two outputs are of equal amount, but only one of them is assigned to a hitherto unused address, then it is chosen to be clubbed with the input addresses. For single input single output transactions, the involved addresses are also grouped together, and for single input 2 output transactions, the identical heuristics described above for clubbing input and output addresses apply. Groups so formed across transactions, if they have any common addresses among themselves, are merged together.

Such Bitcoin wallet address aggregation heuristics are inaccurate (for instance, in¹⁴, such an address aggregation algorithm identified more than 33 address groups that each contained more than 100,000 up to a million wallet addresses, and an aggregated group of more than a million wallet addresses), and are based on assumptions and simplifications of Bitcoin wallet usage, which are not always true. The aforementioned heuristics totally ignore the multi-input/output transactions, and the scope of these heuristics is thus confined to aggregating addresses from non-multi-input/output transactions. But even for the family of transactions which fit the scope, in general, usage of such heuristics do not yield full recall

FIGURE 15 Graphs derived by address aggregation

(they do not identify all the addresses that belong to same entity), nor are they precise (they may aggregate addresses that in fact belong to different entities). In our previous study^{5,12} we have noticed that, aware of such heuristics being used for address aggregation, people deliberately carry out transactions (e.g., both the output addresses belong to the same user, but the aggregation heuristic follows the trail of the smaller output amount, ignoring the address receiving the larger amount) to muddle the results from such heuristics.

Nevertheless, in the absence of any ground truth or significantly better well established address aggregation techniques, and given the profuse use of these heuristics in the literature, we apply the above described heuristics for each of the four periods, to carry out address aggregation of the wallet addresses, and construct the network among these groups of aggregated addresses AG_1 , AG_2 , AG_3 , AG_4 , and study the properties of these networks.

For Period 1, of the total 55295 wallet addresses (see Table 2), 43341 unique addresses were involved in transactions which fit the scope of the heuristics, and resulted in 5576 groups (this can be seen as a compression by a factor of approximately 7.77). For Period 2, 909662 addresses were clubbed into 384398 groups (compression by a factor of 2.36). For Period 3, 1702204 addresses yielded 697351 groups (compression by a factor of 2.44), while the numbers for Period 4 were 11374733 addresses leading to 524096 groups (compression by a factor of 21.7). This huge compression may be interpreted in several ways, the two most common possibilities are: (i) Individuals used many more wallet addresses on an average, yet they were not particularly cautious not to use them in a manner to avoid the basic aggregation heuristics. (ii) Distinct entities were (much more often, for whichever reason) wrongly grouped together by the heuristics.

Histograms of the size distributions of the aggregated address groups for the four periods are shown in Figure 14. The sizes of the largest group for the four periods were respectively 21093, 579, 504 and 9453034, which corresponded to 0.4864, 0.0006, 5.87×10^{-7} and 0.831 fraction of the nodes that were considered for aggregation by the heuristics.

The presence of one very large group in Period 4, and also in Period 1, which club together very large fractions of all the nodes considered for aggregation suggest that the latter (nodes being wrongly grouped together) is the principal cause. This in turn also means that even though such heuristics are popularly used, they in fact have little practical value. Particularly for Period 4, the creation of such a very large group may have come about simply because of the very large number of transactions, and possibly large number of distinct users, with a larger likelihood of real world interactions, some of which led to wrong groupings, but leading to a snow-balling effect to lead to humongous grouping. In a previous study of aggregated address groups¹⁴, similar humongous clusters were observed, and the authors attributed its presence due to similar wrong merger of many distinct users, apparently predominantly of different users using Mt. Gox, because of some peculiarities on how Mt. Gox handled users' private keys. It is unclear from the reading of¹⁴ whether the large cluster observed was solely because of Mt. Gox exchange, or it also included other causes. Likewise, the possible presence of wrong mergers in the formation of the other identified groups was not specifically discussed. Nevertheless, such

$\mathcal{G}^{\text{in}}(\text{AG}_1)$	0.7333942	$\mathcal{G}^{\text{out}}(\text{AG}_1)$	0.4671081	G_1	$r = -0.1547348$
$\mathcal{G}^{\text{in}}(\text{AG}_2)$	0.6408136	$\mathcal{G}^{\text{out}}(\text{AG}_2)$	0.6611083	G_2	$r = -0.0541277$
$\mathcal{G}^{\text{in}}(\text{AG}_3)$	0.6901881	$\mathcal{G}^{\text{out}}(\text{AG}_3)$	0.6331952	G_3	$r = -0.0097702$
$\mathcal{G}^{\text{in}}(\text{AG}_4)$	0.6376942	$\mathcal{G}^{\text{out}}(\text{AG}_4)$	0.7520560	G_4	$r = -0.1624468$

TABLE 5 Gini and Pearson coefficients for the in-/out-degrees of the aggregated address networks.

wrong mergers were already observed in the literature, and we notice unsurprisingly (given that there are fundamental shortcomings that lead to false positives in the heuristics), in our study that such behavior persists.

We emphasize that we do not have the ground truth and the plausible explanation is just a hypothesis. We defer the systematic study of the underlying causes to understand the observations we make here to the future (and to researchers who may be better equipped to study the hypothesized snow-balling phenomenon). In¹⁴, a possible mitigation has been proposed to avoid wrong mergers - namely, significant changes in merged group size can be used as an indicator of wrong mergers. Based on our search of the literature, we could not locate any study of application of this heuristic, let alone its effectiveness validated.

We derive the network among the aggregated addresses, from the corresponding original address networks (by treating all addresses within individual groups as equivalent). In Figure 15 we show the power-law characteristics of the networks so derived. Though the address aggregation heuristics were applied only on non-multi-input/output transactions, once equivalent addresses are considered to be same, we also see a drop in the number of multi-input/output transactions. For the three relevant periods, the number of remaining multi-input/output transactions were 75527 (down from 88938 for Period 2), 333486 (down from 350431 for period 3) and 120757 (down from 300290 for period 4). There are two interesting observations to be made here. First, in period 3, where the relative frequency of multi-input/output transactions was highest among the considered periods, only a small fraction ≈ 0.048 of the multi-input/output transactions were reduced (as compared to a reduction by a fraction of 0.15 in Period 2, even though both had roughly comparable compression due to address aggregation). This suggests that most of the multi-input/output transactions in Period 3 were carried out more cautiously, without reusing older addresses together. In contrast, Period 4 witnessed a drastic reduction by a fraction of 0.597, which we attribute to the large compression of a factor of more than 20 noted above.

Compared to the Gini coefficients observed for the wallet address networks, we see (in Table 5) different trends of increase (e.g., in Period 1 and 2) as well as slight decrease (e.g., in Period 3), while for Period 4 we see increase in out-degree and decrease in in-degree Gini coefficients, for the corresponding aggregated address networks. More pertinently, the numbers across the periods are closer to each other for the aggregated address networks, and the moderately high values indicate that the degree distributions are heterogeneous. This conforms and can be explained with our observation that diverse number of wallet addresses were aggregated to form groups of different sizes (see Figure 14). Such agglomeration suggests that many of the low degree nodes (which considered separately lead to greater homogeneity) disappear in the aggregated network, thus creating a more heterogeneous population.

There are two interesting observations to be made from the Pearson coefficients observed (in Table 5) across the periods for the aggregated address networks. Period 3 stands out in having the lowest absolute value (it was also so for the wallet address network). But unlike Periods 2 and 3, where the values are close across the wallet and aggregated networks, we see a significant increase in the absolute values for Periods 1 and 4, and there is a greater clarity in them being disassociative (given the negative values).

5 | CONCLUSIONS & FUTURE WORK

We studied mixing-like behaviors in the Bitcoin network by looking at multi-input/output transactions over 4 representative periods. The data used for our study is available as CSV files¹⁵. This exploratory analysis of the Bitcoin network has three broad contributions:

- (i) We reapply some prior analysis (e.g., Gini, clustering, and Pearson coefficients, or looking at the degree distribution in terms of power law) which were conducted with old data sets, and we study new data sets which have multi-input/output transactions, to compare whether and how the macroscopic network properties have changed over time; we also provide a comparison of wallet addresses before and after agglomeration.
- (ii) We study new macroscopic (e.g. probability of a (non-)multi-input/output transaction to be followed by a (non-)multi-input/output transaction) as well as microscopic properties (e.g. (normalized-)entropy and connected components comprising exclusively multi-input/output transactions) not investigated in previous studies; and in the process we identify indicators that highlight outliers in new manners (e.g., high (normalized-)entropy in transactions, graph comprising cascades of multi-input/output transactions, paths within such graphs), that can be used to identify and isolate a portion of the network to be scrutinized in further detail. Such mechanisms for identifying and isolating nodes or regions of interest within the larger Bitcoin network is interesting and useful, particularly because of the humongous volume of data the whole network comprises, making it impractical for certain purposes (such as forensics, detection of money laundering) to study in its entirety.

(iii) Out of the 4 periods studied, we showed that periods 3 and 4, where mixing is likely to be more frequently present, exhibited different behaviors with respect to several measures. We observed that mixing behavior has often been very cautious, e.g., when mixing is balancing the number of inputs and outputs, then it does so with relatively small numbers of inputs and outputs so that they do not immediately stand out based on the transaction fan-in/outs. We also pin-pointed transactions which maximize the normalized entropy indicate a very deliberate and careful approach to obfuscate the flows, and suggest mixing.

This work was exploratory in nature, and as explained above, we have identified several promising indicators. The natural next step for our future work is to develop tools to zoom into and study the subnetworks of interest, and develop actionable intelligence from the same, for instance, design better mechanisms for wallet address aggregation (as seen in this study, existing heuristics are too flawed to be of any practical value), identify with greater confidence whether specific (set of) transactions are mixing, and thus produce practical tools to support forensics and anti-money laundering efforts.

References

1. Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. 2008.
2. Lischke M, Fabian B. Analyzing the Bitcoin Network: The First Four Years. *Future Internet* 2016; 8(7).
3. Kondor D, Pósfai M, Csabai I, Vattay G. Do the Rich Get Richer? An Empirical Analysis of the Bitcoin Transaction Network. *PLoS ONE* 2014; 9(2).
4. Spagnuolo M, Federico M, Stefano Z. Bitiodine: Extracting intelligence from the bitcoin network. *Intl. Conf. on Financial Cryptography & Data Security* 2014.
5. Oggier F, Phetsouvanh S, Datta A. BiVA: Bitcoin Network Visualization & Analysis. *IEEE International Conference on Data Mining (demo track)* 2018.
6. Bartoletti M, Lande S, Pompianu L, Bracciali A. A general framework for blockchain analytics. *Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers* 2017.
7. Akcora C, Kantarcioglu M, Gel Y. Blockchain Data Analytics. *IEEE International Conference on Data Mining (ICDM)* 2018.
8. Bistarelli S, Santini F. Go with the -Bitcoin- Flow, with Visual Analytics. *Intl. Conf. on Availability, Reliability & Security (ARES)* 2017.
9. CoinJoin: Bitcoin privacy for the real world. <https://bitcointalk.org/?topic=279249>; 2013.
10. Miers I, Garman C, Green M, Rubin A. Zerocoin: Anonymous Distributed E-Cash from Bitcoin. *IEEE Symposium on Security and Privacy* 2013.
11. S. Phetsouvanh FO, Datta A. A 4571 node 4762 edge directed weighted Bitcoin address subgraph. <https://doi.org/10.21979/N9/FHBR2E>, DR-NTU (Data); 2019.
12. S. Phetsouvanh FO, Datta A. EGRET: Extortion Graph Exploration Techniques in the Bitcoin Network. *IEEE International Conference on Data Mining Workshops (ICDMW)* 2018.
13. Nick J. Data-Driven De-Anonymization in Bitcoin. ETH Master Thesis; 2015.
14. Harrigan M, Fretter C. The Unreasonable Effectiveness of Address Clustering. Tech. Rep. arXiv:1605.06369, arXiv; 2016.
15. Silivanxay P, Datta A, Oggier F. Four Bitcoin subgraphs for analysis of multi-input multi-output transactions. DR-NTU (Data); 2019.
16. Narayanan A, Bonneau J, Felten E, Miller A, Goldfeder S. *Bitcoin and Cryptocurrency Technologies*. Princeton University Press . 2016.
17. Chang T, Svetinovic D. Improving Bitcoin Ownership Identification Using Transaction Patterns Analysis. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS* 2018.

How to cite this article: S. Phetsouvanh, A. Datta, F. Oggier, (2019), Analysis of Multi-input Multi-output Transactions in the Bitcoin Network, , .