LEDGER
ledgerjournal.org

ISSN 2379-5980 (online)
DOI 10.5195/LEDGER.20XX.XXX

RESEARCH ARTICLE

Heuristic-Based Address Clustering in Cardano Blockchain

Mostafa Chegenizadeh[*], Sina Rafati Niya[†], Claudio J. Tessone[‡]

Abstract. Blockchain technology has recently gained widespread popularity as a practical method of storing immutable data while preserving the privacy of users by anonymizing their real identities. This anonymization approach, however, significantly complicates the analysis of blockchain data. To address this problem, heuristic-based clustering algorithms as an effective way of linking all addresses controlled by the same entity have been presented in the literature. In this paper, considering the particular features of the Extended Unspent Transaction Outputs accounting model introduced by the Cardano blockchain, two new clustering heuristics are proposed for clustering the Cardano payment addresses. Applying these heuristics and employing the UnionFind algorithm, we efficiently cluster all the addresses that have appeared on the Cardano blockchain from September 2017 to January 2023, where each cluster represents a distinct unique entity. The results show that each medium-sized entity in the Cardano network owns and controls 9.67 payment addresses on average. The results also confirm that a power law distribution is fitted to the distribution of entity sizes recognized using our proposed heuristics.

KEY WORDS
1. Heuristics. 2. Address Clustering. 3. Cardano. 4. Extended UTXO.
5. UTXO-based Blockchain. 6. Blockchain Analytics

1. Introduction

Blockchain technology has attracted much attention in recent years as a revolutionary way of immutable data storage that has enabled a new generation of financial exchange platforms. In contrast to traditional banking systems, blockchain technology proposes a promising decentralized method enabling users to transfer value/assets to other users without the need for the presence of any centralized trusted third party to act as an intermediary in the network. In a blockchain network, each node is connected to and exchanges data with an arbitrary number of desired nodes, forming a peer-to-peer network. The main purpose of this decentralized network is to reach an agreement on a unique ledger, which is an ordered chain of transactional data. The key idea is that there is no central authority in a blockchain-based network. The transactions are

[*] M. Chegenizadeh (mostafa.chegenizadeh@uzh.ch) is a PhD Candidate in the Blockchain & Distributed Ledger Technologies Group at the Department of Informatics, University of Zurich, Switzerland.

[†] S. Rafati Niya (sina.rafatiniya@uzh.ch) is a Senior Research Associate in the Blockchain & Distributed Ledger Technologies Group at the Department of Informatics, University of Zurich, Switzerland.

[‡] C.J. Tessone (claudio.tessone@uzh.ch) is Professor of Blockchain and Distributed Ledger Technologies at the Department of Informatics, University of Zurich, Switzerland. He is also co-founder and Chairman of the UZH Blockchain Center.

generated by the network users and sent to the peers. The peers validate the transactions, put them together in blocks, store the blocks in their local ledger, and propagate the blocks to other peers. In this way, the transactions are broadcasted throughout the whole network, ensuring that a common record of transactions is shared among all peers.

The basic decentralized consensus protocol introduced by Satoshi Nakamoto in 2008 was called Proof of Work (PoW).[2] In PoW, each peer needs to complete a computationally heavy task before it can publish a block of transactions. The first peer that completes this task and publishes the block will be awarded according to the protocol. In fact, finding the appropriate parameters for a block is a time- and computation-intense task certifying some intensive work has been done by the peer, hence the name "Proof of Work". However, this heavy computation is used only to prove that some "work" is done, while the "work" itself is in some respects useless—at least avoidable—in the protocol. Thus, this "work" can be regarded as an inefficient utilization of time and energy. To address this problem, alternative consensus protocols have been introduced after PoW. One of the alternative protocols is called Proof of Stake (PoS).[23] In PoS, only stake owners, or users who have a certain amount of cryptocurrency in their possession, are in authority to validate new blocks. The next block validator is selected randomly with a higher probability of selecting a stake owner with a larger stake. This setting means that the selected validator does not need to perform heavy computations, but only validates the block's transactions.

Blockchains are supposed to determine the provenance and ownership of assets in the network. To this end, the existing blockchains use two different accounting models.[25] The first model is called Unspent Transaction Output (UTXO)-based model, which is similar to traditional cash trading. In this model, each transaction spends a finite number of inputs and generates some outputs with fixed values, where the inputs are unspent outputs of previous transactions, and the total amount of inputs must exceed the total amount of outputs. UTXOs are indeed transaction outputs that have not been spent yet. Each output is locked by a specific public address, which identifies a particular entity in the network allowed to spend it. Since the value of each UTXO is determined in the transaction generating it, a UTXO owner who intends to spend it cannot divide it but instead must spend the entire amount and receive a change value in return. Bitcoin,[2] which is a cryptocurrency with the largest market capacity,[46] exemplifies this UTXO-based accounting model. The second model is called the account-based model, which is closely similar to the banking system; where each user owns one or more accounts, and the transactions represent a transfer of an arbitrary amount of value from one account to another. In this setting, the blockchain can be considered as a state machine that records the current balances of all accounts, where each transaction changes the state of the blockchain by updating the balances: subtracting the input value from the sender's account and adding the output value to the receiver's account. Account-based model is exemplified by Ethereum,[20] which is the second-largest cryptocurrency in the world.[46] Ethereum is also the first blockchain platform that presented the concept of smart contracts.

In contrast to the banking system, the details of all transactions are publicly available in public blockchains, which can threaten the privacy of users. For this reason, in many UTXO-based blockchains (e.g., Cardano, Bitcoin, Litecoin, and Monacoin), each user/entity has been enabled to generate and control an arbitrary number of anonymous addresses. Furthermore, many crypto wallets (e.g., Daedalus, Yoroi, and Coinbase), by default, generate new change addresses for the users/entities after each transaction. Accordingly, it is reasonable for data analysts to assume that some blockchain users/entities are expected to own multiple addresses.

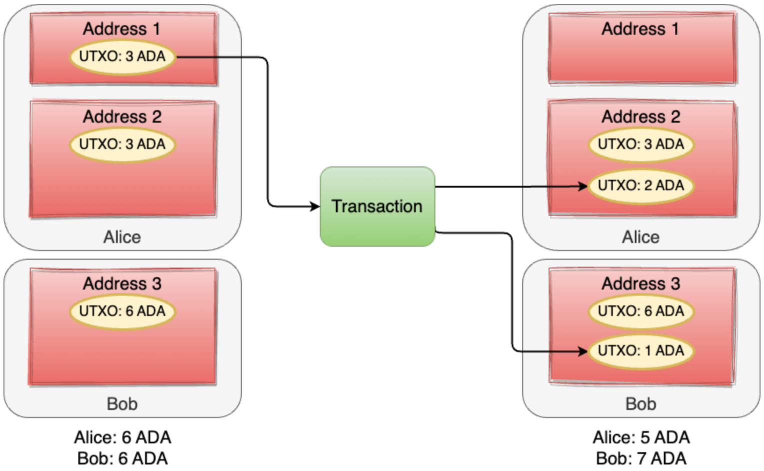

Fig. 1. Example scenario 1: value transfer among different addresses of the same entity

Considering that it is necessary for many financial analyses—for example, wealth distribution or asset velocity—to take into account the transactions between distinct entities rather than distinct addresses, address clustering would be the first step to conducting a statistical analysis on a UTXO-based blockchain in many cases. In other words, to analyze the network parameters of a UTXO-based blockchain, data analysts cannot treat each address as if it belongs to a distinct entity; instead, they need to cluster all addresses controlled by the same entity together and then perform their analyses based on those clusters.

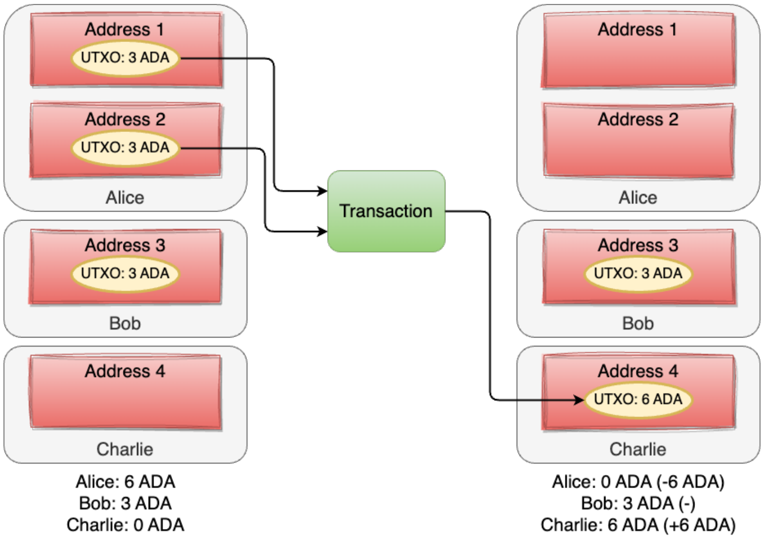

Fig. 2. Example scenario 2: Value transfer from different addresses of the same entity

Here are two scenarios that can provide insight into the significance of address clustering. As the first instance, when a certain amount of value is transferred from Address1 to Address2 through a transaction, while both addresses belong to the same entity, neither the distribution of wealth nor the velocity of assets must be affected in analyses by that part of the transaction because it represents no value transfer in the real world. Whereas without knowing that these addresses represent the same entity, the results of such analyses would not be accurate. As shown in Fig. 1, the transaction transfers 3 ADA in the network, while the real amount of transferred value among users is only 1 ADA. Another example would be when two UTXOs stored in two distinct addresses belonging to a single entity are spent. In this case, the corresponding impact on analyses must be completely different from the case when two similar UTXOs owned by two separate entities are consumed. Fig. 2 and Fig. 3 illustrate the difference between these two cases.

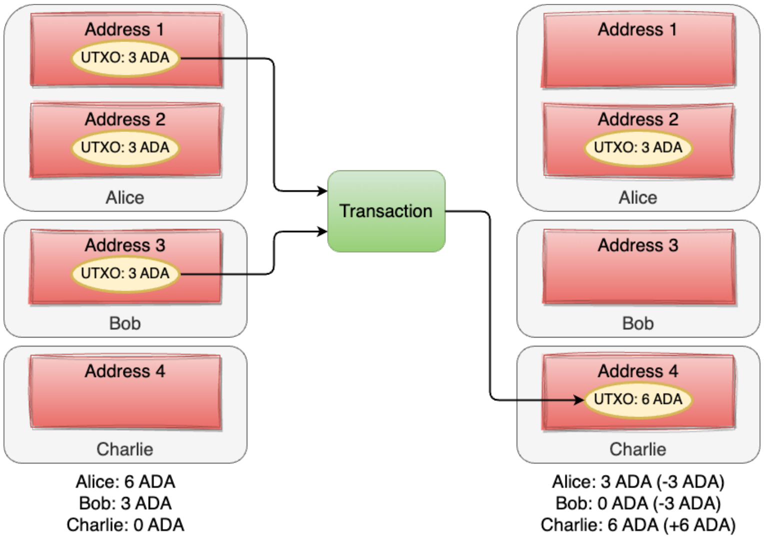

Fig. 3. Example scenario 2: Value transfer from different addresses of different entities

As can be conceived by the scenarios above, capturing a reliable address clustering has a key role to play in UTXO-based blockchain analytics. The effective heuristics for address clustering in UTXO-based blockchains have been discussed in a number of previous studies.[1,5–11,15] However, since Cardano has extended the UTXO-based accounting model by introducing the Extended Unspent Transaction Output (EUTXO)-based model[24] to support smart contracts, the already-existing heuristics developed for other blockchains are not directly applicable to Cardano. In this paper, we focus on Cardano address clustering and propose two heuristics for the EUTXO model. We apply our heuristics to the Cardano blockchain data to create clusters of addresses, with each cluster representing an entity controlling all addresses within the cluster.

In the following, a brief explanation of the basic concepts and preliminaries of the Cardano blockchain is presented in **Section 2**. Afterward, an overview of the literature is provided in **Section 3** by referring to relevant work proposed in the blockchain address clustering area. The approach followed in this work for address clustering is presented in **Section 4**. The implementation methods, as well as the analytics results, are discussed in **Section 5**. Finally, **Section 6** presents the conclusions.

2. Background and Preliminaries

2.1. ***Cardano Consensus Protocol***—Cardano consensus protocol, Ouroboros, is designed based on Proof of Stake.[22] Each user holding a stake in Cardano can earn passive rewards by participating in the block validation process. This participation can be achieved by either 1) establishing a new stake pool and pledging/delegating stake to this pool or 2) delegating stake to an already-existing pool. In order to delegate their stakes to a pool, users must first register a stake address, and then obtain a delegation certificate for that address. A stake pool typically consists of three components: 1) one stake pool operator (SPO), 2) one/multiple stake pool owners, and

3) one/multiple stake delegators. In the Cardano protocol, time is divided into epochs that last 432,000 slots, each lasting one second. In every time slot, one of the online registered pools is randomly chosen by the protocol to act as the slot leader, the pool eligible to produce the next block. Later on, validation rewards are calculated proportional to the amount of stakes pledged or delegated to the pool and are distributed by the protocol among the pool components. More specifically, to become redeemable and consumable, these rewards must be first withdrawn via transactions that convert them into spendable UTXOs.

2.2. ***Cardano Accounting Model***—Cardano basically falls in the category of UTXO-based blockchains. Cardano's native cryptocurrency is called ADA. Having said that, Cardano has introduced the EUTXO-based model, which extends the UTXO-based model to support smart contracts. A smart contract is a low-level code script that runs synchronously on multiple blockchain nodes.[3] In contrast to Bitcoin which does not support complex smart contracts but only simple scripts, Cardano's native programming language for smart contracts, Plutus, is a Turing-complete language written in Haskell and is capable of implementing any logic.[26] In Cardano's EUTXO model, each UTXO is locked by a payment address, which refers to 1) a payment key, 2) a simple script (e.g., multi-signature script), or 3) a Plutus smart contract.[27–29] A crucial point to note here is that the scripts and smart contracts can maintain assets in their own addresses, and these assets can be spent if and only if the conditions pre-defined in the script/contract are satisfied by the spending transaction.

2.3. ***Cardano Address Types***—Highlighting the features relevant to our clustering method, Cardano addresses can be categorized into two major categories as follows.[21]

- **Payment Addresses**: Only this type of address can own/spend a UTXO. A payment address can be either a Shelley address or a Byron address.
  - **Shelley**: A Shelley address has two main parts.
    - **payment part**: This part refers to 1) a payment key, 2) a simple script, or 3) a Plutus smart contract that owns funds stored in the Shelley address.
    - **delegation part**: This part may refer to a stake address that owns stake rights of funds stored in the Shelley address.
  - **Byron**: This is an old address format kept in the protocol only for backward compatibility.
- **Stake Addresses**: A stake address owns/controls staking rewards and refers to a stake key.

3. Related Work

There are different blockchain address clustering methods,[1,31] such as heuristic-based,[7–16] off-chain information-based,[32–34] and behavior-based methods,[35,36] the last of which includes data mining-based[37–41] and deep learning-based methods.[42–44] This section, however, describes only a number of heuristic-based studies that focus on UTXO-based blockchain address clustering and are more relevant to our work. Apart from these studies, several heuristics have also been explored for clustering and de-anonymizing the addresses in account-based blockchains such as Ethereum[17,18] and Ripple,[19] which are out of the scope of this paper.

First of all, Nakamoto hinted at the possibility that multiple input addresses of a transaction may be associated with the same person or entity.[2] Based on this assumption, Reid and Harrigan[7] later proposed the multi-input heuristic method as an effective identity detection method. In their paper, they also talked about the detection of change addresses as another heuristic; where the change address in a transaction indicates an output address that belongs to the sender and receives excess input. As part of their study, Androulaki et al.[8] analyzed Bitcoin's privacy provisions by applying multi-input and change address heuristics to recognize entities in the network. Their assumption was that new change addresses were used to receive transactions' changes. According to their findings, even when the users used new addresses for each transaction, almost 40% of them were still identifiable. Meiklejohn et al.,[9] assuming that change addresses only have one input, broadened the change heuristic to include transactions with three or more outputs. Ortega et al.[10] proposed a new change address heuristic by choosing the output address whose value had more decimals to be the change address. Their change address heuristic was indeed based on the assumption that real outputs usually have reduced decimals. Nick[11] selected the output address whose value was smaller than all inputs of a transaction as the optimal change address. In that study, it was assumed that crypto wallets typically do not spend unnecessary UTXOs.

Zhang et al.[12] modified the one-time change address heuristic by discarding the change addresses that were reused as non-change addresses after the change address was made. On the basis of multi-conditional recognition, Liu et al.[13] developed a one-time change address identification algorithm to cluster the Bitcoin addresses. In comparison to other heuristics, they found that their method can reliably detect almost 12% more one-time change addresses. He et al.[14] improved the change address heuristic and proposed a new heuristic based on the different number of output address transactions. They used six heuristics, including multi-input and change address heuristics, to recognize entities by address clustering. They also added conditional constraints to enhance the accuracy of the identified change addresses and hasten the algorithm's convergence.

Combining the heuristics implemented in the BlockSci[45] C++ library, Campajola et al.[15] developed their own heuristic-based address clustering algorithms, most of which relied on the detection of change addresses in transactions. In their study, they employed logical combinations of five heuristics, such as multi-input and change address heuristics, in order to minimize the number of false positives. Adapting from these five heuristics, Rafati Niya et al.[30] for the first time, performed an address clustering to identify the addresses belonging to the same wallet in the Cardano blockchain. They also conducted an analysis of stake balance distribution, reward distribution, and wealth concentration in the Cardano blockchain.

However, according to our knowledge, none of the above-mentioned heuristics has yet been customized in the literature in order to fit the specific features of the EUTXO model used in the Cardano blockchain.

4. Approach

In this work, we study address clustering in the Cardano blockchain by applying a heuristic-based approach. Inspired by the heuristics that have already been developed for UTXO-based blockchains, two heuristics derived from the specific features of the EUTXO model are proposed in the following. The first heuristic is a modified version of the multi-input heuristic,[7] which is used for other UTXO-based blockchains like Bitcoin as outlined in **Section 3**. The second heuristic, however, is inspired by the staking and delegation mechanisms inherent in the Cardano consensus protocol and is unique to this protocol.

- **Heuristic 1 (modified multi-input heuristic)**: All Byron payment addresses, and also all Shelley payment addresses with a payment part referring to a payment key, appearing in the inputs list of a single transaction are assumed to belong to the same entity. This assumption stems from the fact that regular Cardano wallets, such as Daedalus, Yoroi, and Coinbase, support only generating transactions that spend UTXOs from wallets controlled by a single user. Particularly, note that Shelley payment addresses whose payment part refers to a simple script or a smart contract are excluded from this heuristic. This is because, when generating a valid transaction, there is no restriction on the selection of input scripts/contracts, i.e., entities can choose any previously created script/contract they wish as input—provided that the conditions specified in the script/contract are fulfilled by the transaction. It should also be noted that this heuristic is different from the multi-input heuristic,[7] which was proposed for address clustering in Bitcoin and considered all input addresses of a transaction to be owned by the same entity.
- **Heuristic 2 (staking heuristic)**: All Shelley payment addresses whose delegation part refers to the same stake key are assumed to belong to the same entity. This assumption arises from the fact that the entity holding the private key associated with a stake key can unconditionally withdraw the staking rewards stored in the corresponding stake address and send them to any desired payment address. Accordingly, in a normal situation and in the absence of high levels of off-chain trust, it is quite unlikely for one entity to transfer its stake rights to another.

Previous studies have also discussed a few other address clustering heuristics for UTXO-based blockchains, including "optimal change", "new address creation", "address reuse", and "peeling chain" heuristics.[1,8–15] Although these heuristics might be applicable to the Cardano blockchain as well, this paper focuses exclusively on the new heuristics above that are tailored specifically for Cardano.

5. Implementation and Results

In this section, we present the procedure and results of applying the heuristics proposed in **Section 4** to all the Cardano transactions from September 2017 to January 2023 in order to obtain the clustering result of all addresses that have appeared as inputs or outputs of transactions during this period of time.

5.1. ***Dataset, Hardware, and Algorithm***—To cluster the Cardano addresses, first, a Cardano node was set up, which consists of a component called Cardano DBsync. DBSync is the Cardano node's default indexer that collects and stores on-chain data in a PostgreSQL database.[47] After setting the node up, through joining three tables from the database—i.e., tx, tx_out, and tx_in—blockchain's historical transactions data was extracted and stored for further processing. Afterward, using PySpark library version 3.3.1, a Python script loaded this data into memory, allowing the inputs and outputs to be analyzed efficiently in accordance with the heuristics. The Python script was run on a Debian 11 machine with 64 AMD EPYC 3.40 GHz CPU cores and 512 GB memory. For further improvement of efficiency, the Union-Find algorithm[4] was adopted to link the payment addresses during the clustering process.

5.2. ***Clustering Results***—Through the history of Cardano until January 2023, the total number of unique payment addresses recorded on the blockchain as owners of UTXOs reaches 40,330,345, including 29,047,961 Shelley addresses and 11,282,384 Byron addresses. As well, the total number of stake addresses that have appeared on the blockchain as the delegation part of Shelley addresses is 3,868,049, of which 1,665,652 have been registered. This data is summarized in Table 1. The number of new addresses that have appeared on the blockchain over time is shown in Fig. 4.

Table 1. Cardano Address Types

Total Payment Addresses	40,330,345
Shelley Addresses	29,047,961
Byron Addresses	11,282,384
Total Stake Addresses	3,868,049
Registered Stake Addresses	1,665,652

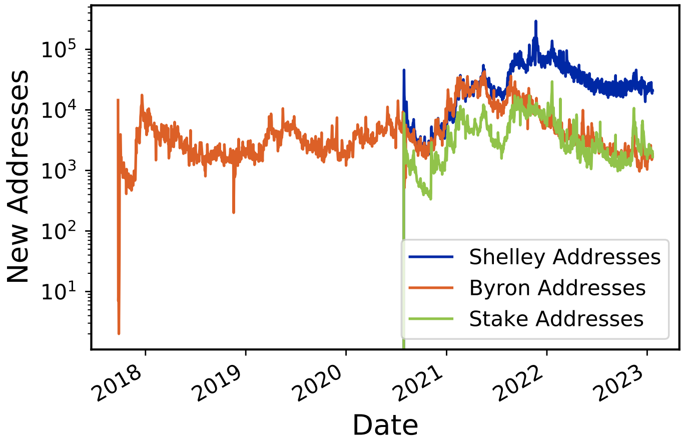

Fig. 4. Number of new addresses appeared on the blockchain per day

We applied **Heuristic 1** to the payment addresses, which resulted in 19,249,106 distinct clusters. Fig. 5 illustrates the distribution of addresses in each cluster.

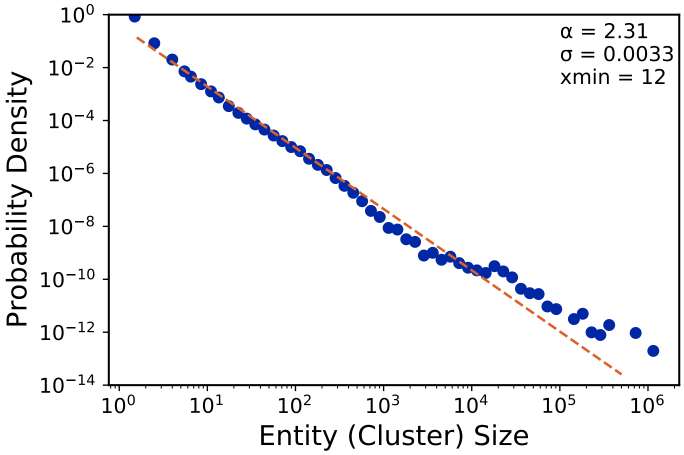

Fig. 5. Distribution of addresses per entities (clusters) determined by **Heuristic 1**

This clustering indicates that the average number of members (addresses) per cluster is 4.94, excluding 16,310,058 single-member clusters and 349 large clusters that contain more than 1000 members. More notably, there are 9 superclusters with more than 200,000 members. Likewise, applying **Heuristic 2**, 1,292,933 non-single-member clusters were detected. The distribution of cluster members according to this analysis is shown in Fig. 6. In this case, after excluding 528 large clusters with at least 1000 members, non-single-member clusters have an average size of 14.63.

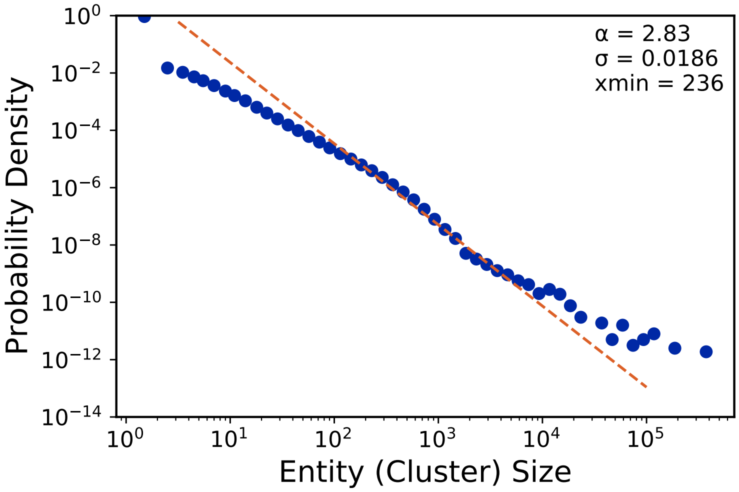

Fig. 6. Distribution of addresses per entities (clusters) determined by **Heuristic 2**

In the subsequent step, the clusters obtained by using the two heuristics were merged, providing a more comprehensive address clustering that is illustrated in Fig. 7. A summary of clustering results derived from **Heuristic 1** and **Heuristic 2** is presented in Table 2. In particular, the average size of medium-sized entities, that is, clusters with between 2 and 1000 members, is 9.67.

Our results show that a power law distribution can be fitted to the distribution of entity sizes recognized using **Heuristic 1** and **Heuristic 2**. A power law probability distribution is in the form of $p(x) \propto x^{-\alpha}$. The dashed lines in Fig. 5, Fig. 6, and Fig. 7 represent a fitted power law distribution obtained by the powerlaw Python package.[48] The power law distributions have been generated with a fitted parameter α and standard error σ for values equal to or more than *xmin*. It is worth mentioning that combining both heuristics yields a more power-law-like distribution with a smaller standard error and smaller *xmin*.

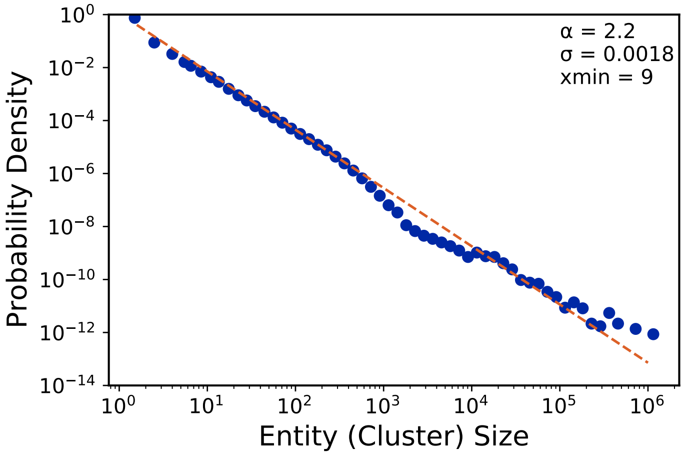

Fig. 7. Distribution of addresses per entities (clusters) determined by **Heuristic 1** and **Heuristic 2**

Table 2. Summary of Clustering Results

Clustering Result	Heur1	Heur2	Heur1 and Heur2
total # of clusters	19,249,106	18,529,342	8,805,791
average size of clusters[a]	4.94	14.63	9.67
single-member clusters	16,310,058	17,236,409	6,621,701
large clusters[b]	349	528	603
superclusters[c]	9	3	12

[a] excluding large and single-member clusters

[b] clusters with more than 1000 members [c] clusters with more than 200,000 members

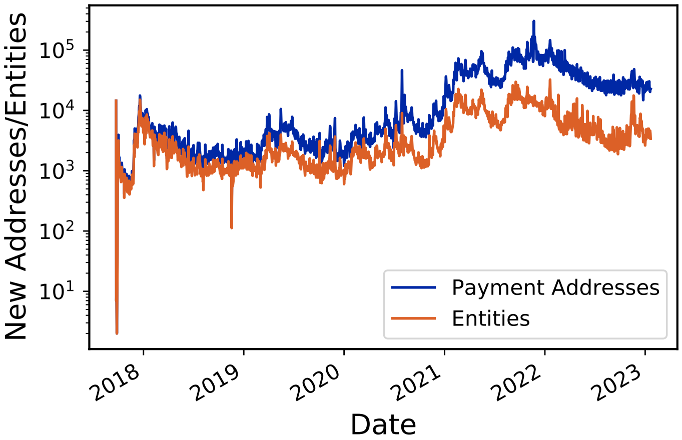

Fig. 8. Number of new payment addresses and new entities (clusters) determined by **Heuristic 1** and **Heuristic 2** appeared on the blockchain per day

Based on this clustering, the number of new entities that have joined the blockchain network over time is calculated and compared with the number of new addresses, the result of which is shown in Fig. 8.

In Fig. 9, the number of active payment addresses per day as well as the number of active entities per day are illustrated. In this figure, if a payment address has appeared in the inputs list of at least one transaction on a given day, that address and consequently its corresponding entity have been considered active.

Finally, based on the address clusters indicating the entities in the network, we calculated the distribution of non-fungible tokens (NFTs) and fungible tokens (FTs) minted by each entity, the results of which are presented in Fig. 10 and Fig. 11. According to this analysis, 602,656 entities out of a total of 8,805,791 entities have contributed to NFT minting in the Cardano network. This number equals 22,872 for the entities that have participated in the generation of FTs.

5.3. ***Discussion and Limitation***—Although the proposed heuristics generally produce accurate results, they may also produce false positive links: assuming that two addresses are connected together, or owned by the same entity, while in reality, they are not. For instance, considering **Heuristic 1**, we have assumed that all UTXOs locked by payment keys listed as inputs of a single transaction are owned by the entity that has generated and signed the transaction via their crypto wallet. However, there are some tools, such as "cardano-cli," that are capable of

ISSN 2379-5980 (online)
DOI 10.5195/LEDGER.20XX.XXX

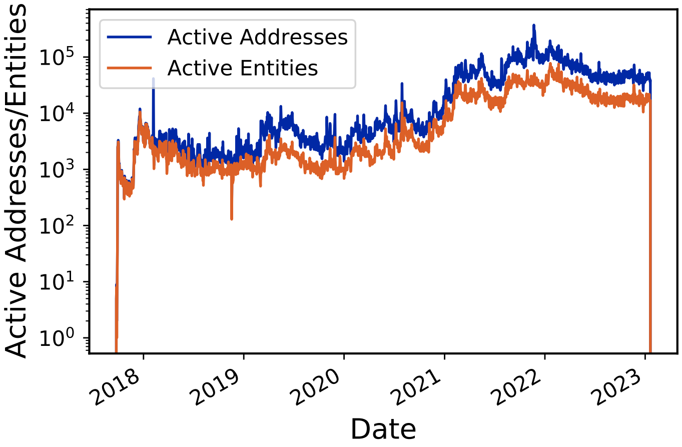

Fig. 9. Number of active addresses and active entities (clusters) determined by **Heuristic 1** and **Heuristic 2** per day

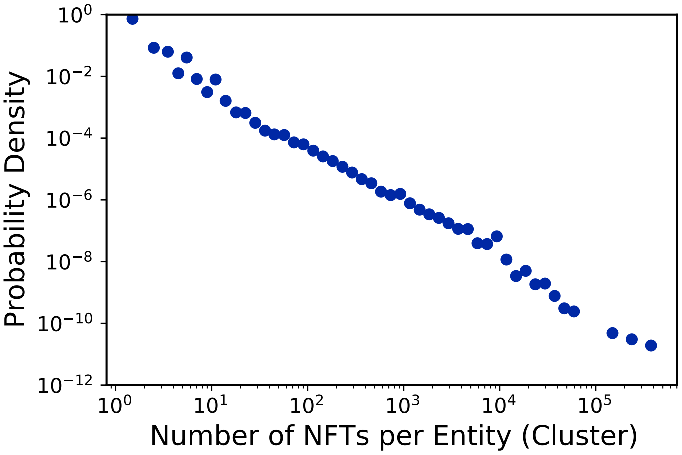

Fig. 10. Distribution of NFTs minted by entities (clusters) determined by **Heuristic 1** and **Heuristic 2** and contributed to NFT minting

generating complex transactions that carry the signatures of multiple users. The input list of such transactions can contain payment keys from different users. Consequently, these transactions may insert false positives into our clustering outputs. Such false positives are probably responsible for the creation of the superclusters. As with **Heuristic 1**, **Heuristic 2** may also lead to false positives. Indeed, it is technically possible for users to store their assets in Franken addresses, which are Shelley addresses whose delegation part refers to another user's stake address. By doing so, the users can pledge additional stakes to a pool without having to register an additional pool owner on the blockchain, although this requires off-chain trust between the two parties.

In order to improve the accuracy of the proposed address clustering algorithm, these false positive links should be detected and subsequently removed from the output of the algorithm in future work.

6. Conclusion

Cardano has extended the UTXO-based accounting model by introducing the EUTXO model to support smart contracts. Because of this extension, the already-existing heuristics developed for address clustering in other UTXO-based blockchains are not directly applicable to Cardano.

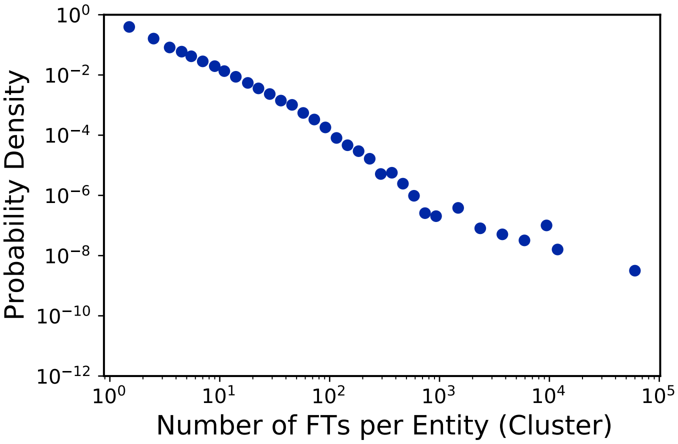

Fig. 11. Distribution of FTs minted by entities (clusters) determined by **Heuristic 1** and **Heuristic 2** and contributed to FT minting

In this paper, we proposed and implemented two new heuristics based on the specific features of Cardano. We also presented the clustering results of Cardano addresses based on the proposed heuristics. The results of this paper can be used as a basis for further blockchain analysis in Cardano concerning wealth distribution and asset velocity. The results show that a power law distribution can be fitted to the distribution of entity sizes recognized using our proposed heuristics. Nevertheless, the proposed heuristics result in the formation of a few superclusters, indicating that the final clustering results still contain false positives. In future work, it would be advantageous to improve the accuracy of this heuristic-based address clustering method by detecting and eliminating false positives.

Acknowledgements

We acknowledge financial support from Cardano Foundation.

Author Contributions

MCH reviewed the literature, performed the experiments, and wrote the initial draft of the manuscript. SRN collected the data, assisted in structuring the scientific writing of the paper, and proofread the manuscript. CJT conceptualized the study, proposed benchmark models, and proofread the manuscript throughout the writing process.

Notes and References

[1] Wu, J., Liu, J., Zhao, Y. and Zheng, Z., 2021. Analysis of cryptocurrency transactions from a network perspective: An overview. Journal of Network and Computer Applications, 190, p.103139.

[2] Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system. Decentralized business review, p.21260.

[3] Zou, W., Lo, D., Kochhar, P.S., Le, X.B.D., Xia, X., Feng, Y., Chen, Z. and Xu, B., 2019. Smart contract development: Challenges and opportunities. IEEE Transactions on Software Engineering, 47(10), pp.2084-2106.

[4] Wenzel Jakob, 2020. Lock-free parallel disjoint set data structure. URL: https://github.com/wjakob/dset.

[5] Ron, D. and Shamir, A., 2013. Quantitative analysis of the full bitcoin transaction graph. In Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers 17 (pp. 6-24). Springer Berlin Heidelberg.

[6] Harrigan, M. and Fretter, C., 2016, July. The unreasonable effectiveness of address clustering. In 2016 intl ieee conferences on ubiquitous intelligence & computing, advanced and trusted computing, scalable computing and communications, cloud and big data computing, internet of people, and smart world congress (uic/atc/scalcom/cbdcom/iop/smartworld) (pp. 368-373). IEEE.

[7] Reid, F. and Harrigan, M., 2013. An analysis of anonymity in the bitcoin system (pp. 197-223). Springer New York.

[8] Androulaki, E., Karame, G.O., Roeschlin, M., Scherer, T. and Capkun, S., 2013. Evaluating user privacy in bitcoin. In Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers 17 (pp. 34-51). Springer Berlin Heidelberg.

[9] Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M. and Savage, S., 2013, October. A fistful of bitcoins: characterizing payments among men with no names. In Proceedings of the 2013 conference on Internet measurement conference (pp. 127-140).

[10] Ortega, M.S., 2013. The bitcoin transaction graph—anonymity (Doctoral dissertation, Master's thesis, Universitat Oberta de Catalunya).

[11] Nick, J.D., 2015. Data-driven de-anonymization in bitcoin (Master's thesis, ETH-Zürich).

[12] Zhang, Y., Wang, J. and Luo, J., 2020. Heuristic-based address clustering in bitcoin. IEEE Access, 8, pp.210582-210591.

[13] Liu, F., Li, Z., Jia, K., Xiang, P., Zhou, A., Qi, J. and Li, Z., 2023. Bitcoin Address Clustering Based on Change Address Improvement. IEEE Transactions on Computational Social Systems.

[14] He, X., He, K., Lin, S., Yang, J. and Mao, H., 2022. Bitcoin address clustering method based on multiple heuristic conditions. IET Blockchain, 2(2), pp.44-56.

[15] Campajola, C., Cristodaro, R., De Collibus, F.M., Yan, T., Vallarano, N. and Tessone, C.J., 2022. The evolution of centralisation on cryptocurrency platforms. arXiv preprint arXiv:2206.05081.

[16] Remy, C., Rym, B. and Matthieu, L., 2018. Tracking bitcoin users activity using community detection on a network of weak signals. In Complex Networks Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications) (pp. 166-177). Springer International Publishing.

[17] Klusman, R. and Dijkhuizen, T., 2018. Deanonymisation in ethereum using existing methods for bitcoin. https://www.os3.nl/_media/2017-2018/courses/rp1/p61_report.pdf

[18] Victor, F., 2020. Address clustering heuristics for Ethereum. In Financial Cryptography and Data Security: 24th International Conference, FC 2020, Kota Kinabalu, Malaysia, February 10–14, 2020 Revised Selected Papers 24 (pp. 617-633). Springer International Publishing.

[19] Moreno-Sanchez, P., Zafar, M.B. and Kate, A., 2016. Listening to whispers of ripple: Linking wallets and deanonymizing transactions in the ripple network. Proc. Priv. Enhancing Technol., 2016(4), pp.436-453.

[20] Wood, G., 2014. Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper, 151(2014), pp.1-32.

[21] Benkort, M., 2020. CIP 19 - Cardano Addresses. URL: https://cips.cardano.org/cips/cip19.

[22] Kiayias, A., Russell, A., David, B. and Oliynykov, R., 2017, July. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Advances in Cryptology–CRYPTO 2017: 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20–24, 2017, Proceedings, Part I (pp. 357-388). Cham: Springer International Publishing.

[23] King, S. and Nadal, S., 2012. Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. self-published paper, August, 19(1).

[24] Chakravarty, M.M., Chapman, J., MacKenzie, K., Melkonian, O., Peyton Jones, M. and Wadler, P., 2020. The extended UTXO model. In Financial Cryptography and Data Security: FC 2020 International Workshops, AsiaUSEC, CoDeFi, VOTING, and WTSC, Kota Kinabalu, Malaysia, February 14, 2020, Revised Selected Papers 24 (pp. 525-539). Springer International Publishing.

[25] Zahnentferner, J., 2018. Chimeric ledgers: Translating and unifying utxo-based and account-based cryptocurrencies. Cryptology ePrint Archive.

[26] Chakravarty, M., Kireev, R., MacKenzie, K., McHale, V., Müller, J., Nemish, A., Nester, C., Jones, M.P., Thompsona, S., Valentine, R. and Wadler, P., 2019. Functional blockchain contracts.

[27] Zahnentferner, J., 2018. An abstract model of UTxO-based cryptocurrencies with scripts. Cryptology ePrint Archive.

[28] IOHK. Understanding the Extended UTXO model. URL: https://docs.cardano.org/learn/eutxo-explainer.

[29] IOHK. Simple Scripts. URL: https://github.com/input-output-hk/cardano-node/blob/master/doc/reference/simple-scripts.md.

[30] Rafati Niya, S., Mesić, I., Anagnostou, G., Brunini, G. and Tessone, C.J., 2023. A First Analytics Approach to Cardano. In 2023 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). IEEE.

[31] Xueshuo, X., Jiming, W., Junyi, Y., Yaozheng, F., Ye, L., Tao, L. and Guiling, W., 2021. AWAP: Adaptive weighted attribute propagation enhanced community detection model for bitcoin de-anonymization. Applied Soft Computing, 109, p.107507.

[32] Ermilov, D., Panov, M. and Yanovich, Y., 2017, December. Automatic bitcoin address clustering. In 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 461-466). IEEE.

[33] Biryukov, A., Khovratovich, D. and Pustogarov, I., 2014, November. Deanonymisation of clients in bitcoin p2p network. In Proceedings of the 2014 ACM SIGSAC conference on computer and communications security (pp. 15-29).

[34] Fleder, M., Kester, M.S. and Pillai, S., 2015. Bitcoin transaction graph analysis. arXiv preprint arXiv:1502.01657.

[35] Monaco, J.V., 2015, May. Identifying bitcoin users by transaction behavior. In Biometric and surveillance technology for human and activity identification XII (Vol. 9457, pp. 25-39). SPIE.

[36] Zhang, Z., Zhou, T. and Xie, Z., 2018. Bitscope: Scaling bitcoin address de-anonymization using multi-resolution clustering. In Proceedings of the 51st Hawaii International Conference on System Sciences (pp. 1-11).

[37] Bartoletti, M., Pes, B. and Serusi, S., 2018, June. Data mining for detecting bitcoin ponzi schemes. In 2018 Crypto Valley Conference on Blockchain Technology (CVCBT) (pp. 75-84). IEEE.

[38] Huang, B., Liu, Z., Chen, J., Liu, A., Liu, Q. and He, Q., 2017. Behavior pattern clustering in blockchain networks. Multimedia Tools and Applications, 76, pp.20099-20110.

[39] Jourdan, M., Blandin, S., Wynter, L. and Deshpande, P., 2019. A probabilistic model of the bitcoin blockchain. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 0-0).

[40] Lin, Y.J., Wu, P.W., Hsu, C.H., Tu, I.P. and Liao, S.W., 2019, May. An evaluation of bitcoin address classification based on transaction history summarization. In 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) (pp. 302-310). IEEE.

[41] Toyoda, K., Mathiopoulos, P.T. and Ohtsuki, T., 2019. A novel methodology for hyip operators' bitcoin addresses identification. IEEE Access, 7, pp.74835-74848.

[42] Shao, W., Li, H., Chen, M., Jia, C., Liu, C. and Wang, Z., 2018. Identifying bitcoin users using deep neural network. In Algorithms and Architectures for Parallel Processing: 18th International Conference, ICA3PP 2018, Guangzhou, China, November 15-17, 2018, Proceedings, Part IV 18 (pp. 178-192). Springer International Publishing.

[43] Tang, H., Jiao, Y., Huang, B., Lin, C., Goyal, S. and Wang, B., 2018. Learning to classify blockchain peers according to their behavior sequences. IEEE Access, 6, pp.71208-71215.

[44] Liang, J., Li, L., Chen, W. and Zeng, D., 2019, July. Targeted addresses identification for bitcoin with network representation learning. In 2019 IEEE International Conference on Intelligence and Security Informatics (ISI) (pp. 158-160). IEEE.

[45] Kalodner, H., Möser, M., Lee, K., Goldfeder, S., Plattner, M., Chator, A. and Narayanan, A., 2020. Blocksci: Design and applications of a blockchain analysis platform. In 29th USENIX Security Symposium.

[46] Bruhn, P. and Ernst, D., 2022. Assessing the Risk Characteristics of the Cryptocurrency Market: A GARCH-EVT-Copula Approach. Journal of Risk and Financial Management, 15(8), p.346.

[47] IOHK. Schema Documentation for cardano-db-sync. URL: https://github.com/input-output-hk/cardano-db-sync/blob/master/doc/schema.md.

[48] Alstott, J., Bullmore, E. and Plenz, D., 2014. powerlaw: a Python package for analysis of heavy-tailed distributions. PloS one, 9(1), p.e85777.

Ledger is published by the University Library System of the University of Pittsburgh as part of its D-Scribe Digital Publishing Program and is cosponsored by the University of Pittsburgh Press.