Unraveling Blockchain based Crypto-currency System Supporting Oblivious Transactions: a Formalized Approach

Lin Chen, Lei Xu, Nolan Shah, Nour Diallo, Zhimin Gao, Yang Lu and Weidong Shi
University of Houston
Houston, Texas, United States
xuleimath@gmail.com, nolanshah212@gmail.com, chenlin198662@gmail.com,
noudiallo@gmail.com, mtion@hotmail.com, ylu17@uh.edu, wshi3@uh.edu

ABSTRACT

User privacy is an important issue in a blockchain based transaction system. Bitcoin, being one of the most widely used blockchain based transaction system, fails to provide enough protection on users' privacy. Many subsequent studies focus on establishing a system that hides the linkage between the identities (pseudonyms) of users and the transactions they carry out in order to provide a high level of anonymity. Examples include Zerocoin, Zerocash and so on. It thus becomes an interesting question whether such new transaction systems do provide enough protection on users' privacy. In this paper, we propose a novel and effective approach for de-anonymizing these transaction systems by leveraging information in the system that is not directly related, including the number of transactions made by each identity and time stamp of sending and receiving. Combining probability studies with optimization tools, we establish a model which allows us to determine, among all possible ways of linking between transactions and identities, the one that is most likely to be true. Subsequent transaction graph analysis could then be carried out, leading to the de-anonymization of the system. To solve the model, we provide exact algorithms based on mixed integer linear programming.

Our research also establishes interesting relationships between the de-anonymization problem and other problems studied in the literature of theoretical computer science, e.g., the graph matching problem and scheduling problem.

Keywords

blockchain, anonymization, privacy

1. INTRODUCTION

The widespread use of the internet-based shopping and banking in the last decades encourages the study of online payment systems as well as digital currencies. There are various well-known online payment systems, including Visa, Mastercard, Paypal, Moneygram and so on. One property that is shared by most of these payment systems is that they are centrally or quasi-centrally administrated in the sense that there exists a centally controlling authority who has access to the information of all transactions carried out within the system. One well-known exception is Bitcoin. Following decades of research on e-cash by the cryptographic community [Camenisch et al. 2005, Canard and Gouget 2007, Okamoto 1995], Bitcoin proves to be one of the most successful decentralized system since its first introduction in 2008 [Nakamoto 2008].

As we have mentioned, a crucial difference between Bitcoin and traditional e-cash systems is that it is fully decentralized. Instead of appointing a central authority (e.g., a bank), Bitcoin uses block chain, a public ledger to record transactions carried out between users. A crucial problem of the Bitcoin system that leads to the threat to users' privacy is that it is public information how transactions are linked to identities (pseudonyms) of users. Bitcoin allows a user to employ multiple pseudonyms in the system to prevent an adversary from discovering his/her true identity. However, a series of studies [Ron and Shamir 2013, Meiklejohn et al. 2013, Reid and Harrigan 2013, Barber et al. 2012] have shown that, based on the linkage between transactions and identities in the system, an adversary can employ transaction graph analyses to determine which identities are actually belonging to the same user, leading to the de-anonymization of the whole system. Indeed, many important parameters regarding to the structure of the Bitcoin transaction graph have been revealed by these studies, increasing the risk of users' privacy. In one example, researchers were even able to trace the spending of 25,000 bitcoins that were allegedly stolen [Reid and Harrigan 2013, Lee 2011].

A lot of work has been done towards better protection of users' privacy while retaining the decentralized feature of Bitcoin. Examples include Zerocoin [Miers et al. 2013], Zerocash [Sasson et al. 2014] and so on. A crucial feature of these new transaction systems is that the information of the linkage between identities and transactions is no longer public. As we will provide details in Section 2, in an oblivious crypto-currency transaction system like Zerocoin, an adversary can only obtain the information that a set of identities are involved in a set of transactions, however, given a spe-

BCC'17, April 02 2017, Abu Dhabi, United Arab Emirates

ISBN 978-1-4503-4974-1/17/04…$15.00

DOI: http://dx.doi.org/10.1145/3055518.3055528

cific transaction, he/she will not be able to trace directly which identities it is linked to.

We focus on such oblivious crypto-currency transaction systems in this paper. A natural question regarding to such systems is that, how much improvement with respect to users' privacy do they bring? Is it possible to de-anonymize the obfuscated transactions? Specifically, does the previous analyses on Bitcoin also imply something on these oblivious transaction systems?

Our Contribution. The paper studies the problem of de-anonymization of an oblivious crypto-currency transaction system, e.g., Zerocoin or Zerocash, by leveraging information in the system that are not directly related, including the number of transactions an identity makes, the time stamp of the sending and receiving of coins and so on. The main contribution of this paper is to provide a model that allows an adversary to link transactions carried out in such a system to users' identities by utilizing these seemingly not relevant information. Once the linkage between transactions and identities are revealed, the adversary may apply transaction graph analyses to derive further information and de-anonymize an oblivious transaction system just like the de-anonymization of Bitcoin. We remark that we are not tracing coins in this paper. All the coins are the same, and we try to uncover the information whether two specific identities of the system have made transactions or not.

Note that a transaction could be viewed as an edge that connects two identities of users in the system. Once the information of such connections is hidden by the system, there could be a huge number of possible ways to connect all the identities. The major technical contribution of this paper is that, we propose a model which finds out the connection that is most likely to be true among all possibilities. We provide algorithms based on mixed integer linear programming to solve it. We also establish surprising relationships between our model and other problems studied in the literature of theoretical computer science, including the graph matching and scheduling problems.

The reminder of the paper is organized as follows: In Section 2 we give an abstract model of blockchain based transaction and describe the problem we are going to solve. We propose and study the probability model and the cardinality in Section 3 and Section 4, respectively. Section 5 discusses related work and we conclude the paper in Section 6.

2. ABSTRACT MODEL OF BLOCKCHAIN BASED TRANSACTION SYSTEM AND PROBLEM STATEMENT

In this section, we give an abstract model that captures the key features of a blockchain based transaction system, and then describe the problem of deanonymization.

2.1 Abstract Model of Blockchain based Transaction System

Blockchain based transaction system utilizes the blockchain as the book keeping mechanism. There are two main models for book keeping on blockchain:

- Per-output transaction model [Nakamoto 2008]: every transaction has a set of inputs and a set of outputs, and each input "spends" one output of a previous transaction;
- Per-address transaction model [Wood 2014, Nxt Community 2014]: every transaction has two addresses, and funds are moved from one address to another without indicating the specific previous transactions from which those funds should be taken.

These two models are equivalent, i.e., transactions recorded under one model can be easily converted to the other model, and we only consider the per-address transaction model in the rest of the work. We use the following abstract model to describe a blockchain based transaction system.

Definition 1 (blockchain based transaction system). *A blockchain transaction system consists of a sequence of records in the form:*

$$(id_{tran}, [id_{sender}], [id_{receiver}], [T_{val}], infor),$$

and the set of records satisfy the following requirements:

1. *Each participant of the system has an identity, and each transaction involved two sets of participants, the senders (vector $[id_{sender}]$) and the receivers (vector $[id_{receiver}]$). One of the sets (senders or receivers) can be empty, which means one transaction is divided into two records.*
2. *$[T_{val}]$ is a set that represents the transaction values.*
3. *Each record has a field info to store information related to transaction(s) (e.g., instructions to establish two parts of a divided transaction).*
4. *The set of records is a totally ordered set, which is reflected by the transaction identity id_{tran}.*
5. *Records are maintained on the blockchain through mining and consensus, which guarantees that honest nodes maintain same records and these records are public available.*

This model is general enough to cover most of existing blockchain based transaction systems like Bitcoin, Ethereum, and Nxt [Nxt Community 2014] ($|[id_{sender}]| = |[id_{receiver}]| = 1$). It also covers privacy preserving systems like Zerocoin [Miers et al. 2013] and Zerocash [Sasson et al. 2014] (a transaction is decomposed to two sub-transactions, one has $|[id_{sender}]| = 0$ and one has $|[id_{receiver}]| = 0$).

2.2 Problem Statement

One important feature of blockchain based transaction system is anonymization, which can be defined in different ways under the abstract model given in Definition 1. Typical anonymization definition includes:

- Unlinkability between a transaction and real persons, i.e., by observing all transactions, an adversary cannot learn information about the real persons behind. A typical system that targets to achieve this feature is Bitcoin [Nakamoto 2008];
- Unlinkablity between a transaction and an identity in the system, i.e., by observing all information recorded on the blockchain, an adversary cannot determine the connections between different identities. Typical systems that are designed to fulfill this requirement include Zerocoin [Miers et al. 2013] and Zerocash [Sasson et al. 2014]. We also refer this type of blockchain based transaction system as *blockchain based oblivious transaction system.*

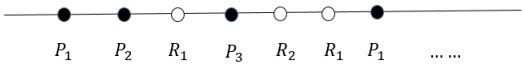

Figure 1: An illustration of the probability model

This paper targets at breaking blockchain based oblivious transaction system with information that can be extracted from the blockchain but are not directly related to the transaction. Note that Bitcoin and similar systems do not provide second type anonymization feature as identities involved in a specific transaction are in plain-text on the blockchain.

3. THE PROBABILITY MODEL

In an oblivious system like Zerocoin, anonymity is achieved via a decentralized mix. On a high level, a mix allows users to entrust a set of coins to a pool and meanwhile retrieve the same amount of coins after some time. By doing so, a transaction can be decomposed into two sub-transactions so as to prevent an adversary from obtaining the information which identities are involved in one specific transaction.

The following model gives a simplified illustration: Suppose there is a box for collecting coins, which is initially empty. There are n people who want to pay coins to another n people, and we call these $2n$ people as payers and receivers, respectively. Payers and receivers are denoted as $P_1, P_2, \cdots, P_n$ and $R_1, R_2, \cdots, R_n$. Specifically, each payer P_i wants to pay exactly one coin to a distinct receiver, and consequently each receiver R_j will receive one coin from exactly one of the payers. Instead of directly paying the coin, each payer proceedes to the box at a certain time t, drops a coin and informs his corresponding receiver that he has dropped a coin into the box. Later on, the receiver goes to the box and picks up a coin. Notice that coins are identical. Payers and receivers represent identities of users in the system.

We illustrate the model as a chain (see Fig. 1) where each black node denotes the event that a payer is dropping a coin, and a white node denotes the event that a receiver is picking up a coin. The chain is a simple representation of a blockchain, where events are listed on the chain according to the time it is added to the system.

Obviously such a model is a major simplification of a blockchain based oblivious transaction system. Especially, we assume there is only one transaction (a payment of one coin) between a payer and a receiver. We focus on such a simplified scenario in this section. The fact that there may be multiple transactions between a payer and a receiver will be addressed in the following sections.

Given a chain of events, the question is, can we infer from this chain which receiver R_j does a payer P_i pay to? That is, we aim to pair each payer P_i with his corresponding receiver R_j.

In general, we cannot pair payers with their receivers as this is the information that an oblivious transaction system, e.g., Zerocoin [Miers et al. 2013] or Zerocash [Sasson et al. 2014], tries to hide. Back to our model, the receiver can pick up his coin at any time later than his corresponding payer drops the coin, and thus P_1, for example, is possible to pay a coin to any R_j. However, in practice, P_1 is more likely to pay a coin to certain R_j's than others. There are many factors which influence the probability that two specific identities make transactions. One important factor is the time. Consider a transaction carried out between a payer and a receiver. It is likely that it is completed within a reasonable time. In other words, once a payer drops a coin into the box, his corresponding receiver is likely to pick up the coin within a reasonable interval on the chain.

We assume the probability information is known based on e.g., empirical experience or experiments carried on a small group of users. More precisely, for each P_i, we assume that $Pr(P_i, R_j)$, the probability that P_i is paying his coin to R_j, is known. Furthermore, we assume transactions between different payers and receivers are independent.

Let π be an arbitrary permutation of the integers $\{1, 2, \cdots, n\}$, which also denotes a pairing where P_i is paying his coin to $R_{\pi(i)}$. Based on our assumptions above, if π is indeed the true pairing, then the chain occurs with the probability:

$$\prod_{i=1}^{n} Pr(P_i, R_{\pi(i)}).$$

Obviously different pairings lead to different probabilities. Among all the possible pairings, we use the idea of maximum likelihood estimation in statistics to determine a pairing that is most likely to be true: Given that the chain is known, we find out the pairing that allows the chain to occur with the highest probability. That is, we solve the following optimization problem:

$$\max_{\pi \in \oplus} \prod_{i=1}^{n} Pr(P_i, R_{\pi(i)}),$$

where $\oplus$ is the set of all possible pairings.

We observe that, the optimization problem above can be reduced to the problem of finding a maximum weighted matching in a bipartite graph, which is a fundamental problem in computer science. We describe the maximum bipartite matching problem as follows.

Maximum Weighted Bipartite Matching: Given a bipartite graph $G = (U \cup V, E)$ where $U = \{u_1, u_2, \cdots, u_n\}$, $V = \{v_1, v_2, \cdots, v_n\}$, $E \subseteq \{e_{ij} | 1 \leq i \leq n, 1 \leq j \leq n\}$. For each edge e_{ij} between u_i and v_j, there is a weight w_{ij} associated. The goal is to find a matching M of the maximum total edge weights, where a matching is subset of E such that no edges share a common vertex. Figure 2 is an example of a bipartite graph

We can transform the probability model into the maximum weighted bipartite matching problem. Given the probability model, we establish a bipartite graph $G = (U \cup V, E)$ where each vertex of U and V correspond to a payer P_i and receiver R_j, respectively. Abusing the notation a bit, we also denote vertices of U and V as P_i and R_j. There is an edge $e_{ij} \in E$ between vertex P_i and R_j if and only if $Pr(P_i, R_j) > 0$, with a weight of $-\ln Pr(P_i, R_j)$. Notice that $\ln Pr(P_i, R_j) \leq 0$ as $Pr(P_i, R_j) \leq 1$, therefore the weight of every edge is positive. See Fig. 2 as an illustration of the transformation.

Consider an arbitrary matching in G, where P_i is matched with $R_{\phi(i)}$ with some permutation ϕ on the integers $\{1, 2, \cdots, n\}$. The weight of this matching is

$$-\sum_{i=1}^{n} \ln Pr(P_i, R_{\phi(i)}) = -\ln \prod_{i=1}^{n} Pr(P_i, R_{\phi(i)}).$$

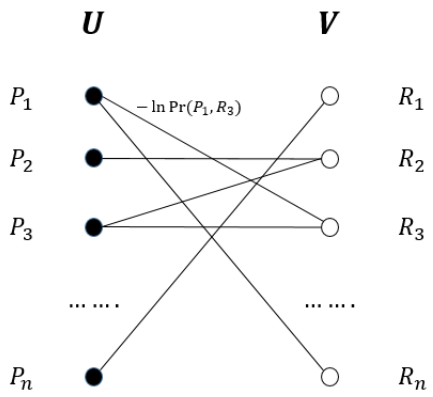

Figure 2: An illustration of the reduction

Therefore, the matching of the maximum weight in the bipartite graph G implies directly a pairing with the largest probability.

We remark that, the maximum weighted bipartite matching problem is a well-known fundamental problem and admit several algorithms. In some literature, it is also known as the assignment problem. Specifically, it can be solved in $O(n^3)$ time by the Hungarian algorithm [Kuhn 2010].

4. THE CARDINALITY MODEL

In the probability model, we make use of the information that the probability a transaction is carried out between a payer and a receiver follows a certain pattern. As we mentioned before, the probability model simplifies an oblivious transaction system by assuming that each payer is only making one transaction with some receiver. In this section, we take into account of the fact that a payer may make multiple transactions with several receivers. We propose the cardinality model, which tries to identify the pairing of payers and receivers based on such information.

The cardinality model: There are m payers and n receivers. Each payer P_i does a_i transactions; each receiver R_j does b_j transactions. Match the transactions between payers and receivers such that the transactions of each payer P_i is carried out with at most c_i different receivers.

The motivation of the cardinality model is based on the fact that, in practice, a payer may carry out multiple transactions, however, it is likely that most of these transactions are done with a few fixed receivers. Indeed, if we consider a payer who makes 100 different transactions, it is more likely that he makes them with a small number of, rather than 100 different receivers. Such an intuition is actually supported by quantitative analysis performed in the Bitcoin system. Indeed, Reid and Harrigan [Reid and Harrigan 2013] constructs a transaction graph based on the dataset from Bitcoin system which contains 1253054 vertices and 4929950 edges, where every vertex corresponds to a user identified by a unique public-key, and every edge implies that there is a transaction between these two vertices. It is not difficult to see that in such a transaction graph, on average each payer is actually making transactions with 3.93 receivers. Therefore, it is reasonable to assume that most c_i's are bounded by a small integer, say, 4. Of course the above data is obtained from the Bitcoin system, which is not an oblivious transaction system, however, we notice that users are not likely to exhibit completely different habits when making transactions within different systems. Generally, if a set of users only make transactions with a small number of other users in the Bitcoin system, then the same set of users are likely to follow this habit in other systems like Zerocoin or Zerocash, that is, their identities are making transactions with a small number of other identities in the system. That being said, we remark that our model does not necessarily restrict c_i's to be small. Indeed, the algorithms provided in this section works for arbitrary c_i.

Variants of the cardinality model: Analogously, we may ask to match transactions between payers and receivers such that the transactions of each receiver R_j is carried out with at most c'_j different payers, which is essentially the same due to the symmetry between payers and receivers in our model. A harder problem is to match transactions between payers and receivers such that the requirements on each payer and each receiver are simultaneous satisfied.

Relationship with the scheduling problem: We observe that, our cardinality model is related with the scheduling problem, which is a fundamental problem in computer science and receives much study in literature. For a nice introduction and overview to the scheduling problems, we refer the readers to [Leung 2004]. One specific scheduling problem that is closely related to our cardinality model is the identical machine scheduling problem with cardinality constraints. In this problem, there are m identical machines and n independent jobs, each of processing time p_j. Job splitting is allowed, i.e., a job could be split into several parts and scheduled independently on different machines. The goal is to assign jobs to machines such that each machine i processes at most c_i jobs, and the total processing time of jobs on machine i is bounded by a_i. Our cardinality model could be viewed as a scheduling problem where each P_i represents a machine, and R_j represents a job of processing time b_j. To the best of our knowledge, our cardinality model is not studied in the literature of scheduling problems. The most relevant previous work [Chen et al. 2016] considers the scheduling problem where job splitting is not allowed and $a_i = a$. However, allowing or disallowing job splitting makes a significant difference. The method in [Chen et al. 2016] cannot be applied to our model and a new approach is required.

Mixed integer linear programming formulation. Let $x_{ij} \in \{0, 1\}$ denotes whether payer P_i makes transactions with receiver R_j. Among the b_j transactions, let $0 \leq y_{ij} \leq 1$ be the fraction of them that are made between P_i and R_j, that is, P_i and R_j make $b_j \cdot y_{ij}$ transactions. We establish the following MILP for the cardinality model.

$$\sum_{j=1}^{n} x_{ij} \le c_i \qquad \forall 1 \le i \le m \qquad (1a)$$

$$\sum_{j=1}^{n} b_j y_{ij} = a_i \qquad \forall 1 \le i \le m \qquad (1b)$$

$$\sum_{i=1}^{m} b_j y_{ij} = b_j \qquad \forall 1 \le j \le n \qquad (1c)$$

$$0 \le y_{ij} \le x_{ij} \qquad \forall 1 \le i \le m, 1 \le j \le n \qquad (1d)$$

$$z_{ij} = b_j y_{ij} \qquad \forall 1 \le i \le m, 1 \le j \le n \qquad (1e)$$

$$x_{ij} \in \{0,1\}, z_{ij} \in \mathbb{N} \qquad \forall 1 \le i \le m, 1 \le j \le n \qquad (1f)$$

We explain each constraint. Constraint (1a) implies that each payer P_i is making transactions with at most c_i receivers. Constraint (1b) and (1c) imply that each P_i and R_j make a_i and b_j transactions in total, respectively. Constraint (1d) implies that if y_{ij} is positive, then P_i must have made transactions with R_j, and therefore $x_{ij} = 1$. We add constraint (1e) to ensure that $b_j y_{ij}$ is an integer.

We remark that constraint (1c) is actually equivalent as $\sum_{j=1}^{n} y_{ij} = 1$. Furthermore, if we further impose the cardinality constraints on the receivers, say, each receiver R_j is making transactions with at most c'_j payers, then we can simply add the following constraint into the above mixed integer linear programming:

$$\sum_{i=1}^{m} x_{ij} \le c'_j \qquad \forall 1 \le j \le n$$

Solving mixed integer linear programming. In general, there is no efficient algorithm that runs in polynomial time for mixed integer linear programming. In theory, the best known algorithm is due to Kannan [Kannan 1987] that runs in $d^{O(d)}|I|^{O(1)}$ time, where d is the number of integer variables and $|I|$ is the length of the input. The mixed integer linear programming constructed in this section consists $2mn$ integer variables, which leads to a running time of roughly $(mn)^{O(mn)}$. Note that this is the bound on the running time of the algorithm in the worst case. In practice, many heuristics works well for mixed integer linear programming, e.g., the branch and bound method [Bader et al. 2005]. There are also several solvers available for solving general mixed integer linear programming, e.g., SCIP and Gurobi.

5. RELATED WORK

In this section, we provide a brief overview on e-cash systems and side channel attacks, which are relevant to our research.

There is a long history of studies on e-cash systems. Chaum [Chaum 1983] was the first to obtain an anonymous e-cash system. Chaum's scheme is anonymous but centralized. Whenever a coin is minted or a transaction between users are carried out, a central trusted authority, e.g., a bank, is involved. Chaum's scheme was further improved by Sander and Ta-Shma [Sander and Ta-Shma 1999].

Subsequent studies focus on establishing an anonymous and decentralized system. One of the most successful example is Bitcoin. It relies on two important features to achieve decentralization, a peer to peer network for broadcasting new transactions, and a public ledger, known as blockchain, to record all transactions carried out within the system. However, Bitcoin fails to achieve user anonymity. The blockchain in the Bitcoin system keeps the information of transactions carried out between identities (pseudonyms). Although a user may employ multiple identities to enhance privacy, a series of studies [Ron and Shamir 2013, Meiklejohn et al. 2013, Reid and Harrigan 2013, Barber et al. 2012]have been carried out and show that it is indeed possible to retrieve the information of a user by analysis on the transaction graphs. Therefore, Bitcoin only provides pseudonymity. It fails to achieve the same level of anonymity as the traditional centralized e-cash system.

A lot of researches are directed towards increasing the anonymity of the Bitcoin system. The major question is how to obfuscate the transaction history so as to prevent an adversary from obtaining the information regarding to the linkage between transactions and identities. A system that hides such information is called an oblivious system in this paper. An oblivious system could be achieved via the help of mixes. A mix allows a user to entrust a set of coins to a pool. Briefly speaking, a user can put coins into the pool and can retrieve the the same amount of coins after some time. The pool serves as a bank and is operated by a central party. The invention of a mix appears to drag the whole research back to centralized anonymous e-cash systems, however, a decentralized mix is possible to be constructed via zero-knowledge proofs. Zerocoin [Miers et al. 2013] is the first decentralized blockchain based transaction system that successfully implement such ideas to derive a decentralized mix. It, however, also has some problems, especially that the verification procedure involved is much time-consuming and only supports coins of fixed denomination. Such problems are addressed in subsequent researches, e.g., Ben-Sasson et al. [Sasson et al. 2014] designed Zerocash to overcome these issues.

The approach used to de-anonymize Bitcoin [Ron and Shamir 2013, Meiklejohn et al. 2013, Reid and Harrigan 2013, Barber et al. 2012]relies on transaction graph analysis, and are hence inapplicable to oblivious systems. In this sense, oblivious transaction systems like Zerocash or Zerocoin do achieve a high level of anonymity. However, it is possible to leverage other parameters to de-anonymize oblivious systems, and we are the first to address this problem by taking into account of the information in the system that are not directly related with the linkage between identities and transactions but may still lead to the revealing of the linkage.

In many context, attacking a system based on information gained from the physical implementation of the system, rather than brute force or theoretical weaknesses in the algorithms in use is called side channel attack. More precisely, such an attack is based on side channel information which is not the plaintext to be encrypted, but rather some source of information collected outside the system [Standaert 2010]. This is related to this paper to some extent, but we remark that our research does not fall into such a category as the source of information used in our model is on the chain and our approach doesn't use any off-chain data for de-anonymizing oblivious transaction. It is an interesting question whether our model could be improved by taking into account of sources of off-chain information. Indeed, prior side-channel research doesn't focus on oblivious transactions and applying side-channel information to un-

cover the anonymized transaction link between sender and receiver.

6. CONCLUSION

We consider the problem of de-anonymizing a blockchain based oblivious crypto-currency transaction system. We propose the probability model and the cardinality model to address this problem. We show that the probability model can be reduced to the maximum bipartite matching problem and admits an $O(n^3)$ time algorithm, while the cardinality model can be formulated as a mixed integer linear programming which could be solved by Kannan's algorithm [Kannan 1987] as well as other heuristics that run faster in practice.

An important open problem is whether we can derive a more efficient algorithm, especially a polynomial time algorithm to solve the cardinality model exactly or approximately, if approximation is acceptable. Another interesting open problem is that, besides the probability and cardinality information we utilized in this paper, is there any other information that allows an adversary to derive the linkage between transactions and identities?

7. REFERENCES

[Bader et al. 2005] David A Bader, William E Hart, and Cynthia A Phillips. 2005. Parallel algorithm design for branch and bound. In Tutorials on Emerging Methodologies and Applications in Operations Research. Springer, 5–1.

[Barber et al. 2012] Simon Barber, Xavier Boyen, Elaine Shi, and Ersin Uzun. 2012. Bitter to better - how to make bitcoin a better currency. In International Conference on Financial Cryptography and Data Security. Springer, 399–414.

[Camenisch et al. 2005] Jan Camenisch, Susan Hohenberger, and Anna Lysyanskaya. 2005. Compact e-cash. In Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 302–321.

[Canard and Gouget 2007] Sébastien Canard and Aline Gouget. 2007. Divisible e-cash systems can be truly anonymous. In Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 482–497.

[Chaum 1983] David Chaum. 1983. Blind signatures for untraceable payments. In Advances in cryptology. Springer, 199–203.

[Chen et al. 2016] Lin Chen, Klaus Jansen, Wenchang Luo, and Guochuan Zhang. 2016. An Efficient PTAS for Parallel Machine Scheduling with Capacity Constraints. In Combinatorial Optimization and Applications - 10th International Conference, COCOA 2016, Hong Kong, China, December 16-18, 2016, Proceedings. 608–623. DOI: http://dx.doi.org/10.1007/978-3-319-48749-6_44

[Kannan 1987] Ravi Kannan. 1987. Minkowski's convex body theorem and integer programming. Mathematics of operations research 12, 3 (1987), 415–440.

[Kuhn 2010] Harold W Kuhn. 2010. The Hungarian method for the assignment problem. In 50 Years of Integer Programming 1958-2008. Springer, 29–47.

[Lee 2011] TB Lee. 2011. A risky currency? Alleged 500,000 Bitcoin heist raises questions. (2011).

[Leung 2004] Joseph YT Leung. 2004. Handbook of scheduling: algorithms, models, and performance analysis. CRC Press.

[Meiklejohn et al. 2013] Sarah Meiklejohn, Marjori Pomarole, Grant Jordan, Kirill Levchenko, Damon McCoy, Geoffrey M Voelker, and Stefan Savage. 2013. A fistful of bitcoins: characterizing payments among men with no names. In Proceedings of the 2013 conference on Internet measurement conference. ACM, 127–140.

[Miers et al. 2013] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. 2013. Zerocoin: Anonymous distributed e-cash from bitcoin. In Security and Privacy (SP), 2013 IEEE Symposium on. IEEE, 397–411.

[Nakamoto 2008] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. (2008).

[Nxt Community 2014] Nxt Community. 2014. NXT Whitepaper. (2014).

[Okamoto 1995] Tatsuaki Okamoto. 1995. An efficient divisible electronic cash scheme. In Annual International Cryptology Conference. Springer, 438–451.

[Reid and Harrigan 2013] Fergal Reid and Martin Harrigan. 2013. An analysis of anonymity in the bitcoin system. In Security and privacy in social networks. Springer, 197–223.

[Ron and Shamir 2013] Dorit Ron and Adi Shamir. 2013. Quantitative analysis of the full bitcoin transaction graph. In International Conference on Financial Cryptography and Data Security. Springer, 6–24.

[Sander and Ta-Shma 1999] Tomas Sander and Amnon Ta-Shma. 1999. Auditable, anonymous electronic cash. In Annual International Cryptology Conference. Springer, 555–572.

[Sasson et al. 2014] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. 2014. Zerocash: Decentralized anonymous payments from bitcoin. In 2014 IEEE Symposium on Security and Privacy. IEEE, 459–474.

[Standaert 2010] François-Xavier Standaert. 2010. Introduction to side-channel attacks. In Secure Integrated Circuits and Systems. Springer, 27–42.

[Wood 2014] Gavin Wood. 2014. Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper (2014).