Sensitivity analysis for a Bitcoin simulation model

By:
Yanan Gong, Kam-Pui Chow, Siu-Ming Yiu and Hing-Fung Ting

From the proceedings of
The Digital Forensic Research Conference
DFRWS APAC 2022
Sept 28-30, 2022

Forensic Science International: Digital Investigation 43 (2022) 301449

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

journal homepage: www.elsevier.com/locate/fsidi

DFRWS 2022 APAC - Proceedings of the Second Annual DFRWS APAC

Sensitivity analysis for a Bitcoin simulation model

Yanan Gong*, Kam Pui Chow**, Siu Ming Yiu, Hing Fung Ting

The University of Hong Kong, Hong Kong, China

ARTICLE INFO

Article history:

Keywords:
Sensitivity analysis
Bitcoin simulation model
Heuristic-based address clustering
Error rates

ABSTRACT

Bitcoin is a popular and widely traded cryptocurrency. The Bitcoin blockchain technology makes it easy for users to conduct pseudo-anonymous financial transactions. However, it also facilitates criminals to secrete their actual identities from law enforcement agencies. Heuristic-based address clustering is the subject regarding Bitcoin de-anonymization. But no heuristic algorithm has a known or potential error rate due to the lack of ground truth. This paper uses sensitivity analysis to validate and verify a constructed Bitcoin simulation model. The evaluation and validation processes examine the model behavior and model outputs from multiple simulation runs to demonstrate fidelity and credibility. The analysis results show no model uncertainties, and the simulation model is stable and can effectively simulate Bitcoin transactions. With a reasonable number of nodes and transaction volumes in the simulated network, the simulation model can be used to verify the effectiveness of two widely used heuristic-based address clustering algorithms and measure the corresponding error rates.

1. Introduction

Bitcoin, a well-known decentralized peer-to-peer (P2P) cryptocurrency, was first invented in 2008 and used in 2009 (Antonopoulos, 2014). Bitcoin blockchain technology has many vital features, such as anonymity and immutability, ensuring user privacy and data security (Peng et al., 2021). New developments in blockchain technology make it harder to trace the actual owner behind a particular Bitcoin address. Users can perform pseudo-anonymous financial transactions with ease. But this also facilitates the ability of criminals to hide their true identities from law enforcement agencies. In 2021, cryptocurrency-based crime reached a new high, with illegitimate addresses receiving $14 billion (Chainalysis Team, 2022).

Previous research on the Bitcoin de-anonymization for identifying real-world identities has been actively developed. Address clustering is a main research topic that divides addresses that may belong to the same user into the same group (called a cluster) (Harrigan and Fretter, 2016). Among them, one popular method is heuristic-based address clustering. However, the lack of ground-truth labels for identifying the actual owner behind each Bitcoin address makes it hard to determine the accurate linkage between addresses and assess the quality of clustering results. It is not admissible in court. For digital evidence, the Daubert standard defines how to determine the admissibility of an expert witness's scientific testimony, and it has five criteria (Daubert Standard; Legal Information Institute, 2022; Garrie, 2014; Luu and Imwinkelried, 2016). Because of the heuristic nature and the features of Bitcoin blockchain technology, the third condition has not been addressed. The determination of error rates can determine how the court can rely on such evidence. No heuristic-based address clustering algorithm has a known or potential error rate.

A simulation model represents the construction of a specific real system (Maria, 1997). Due to considerations like cost, privacy, and anonymity, configuring and deriving accurate error rates on the real Bitcoin network is impossible. Thus, a simulation model can help to study the behavior and properties of the real Bitcoin network. Model validity is crucial in modeling (Maria, 1997). Many reasons, such as incorrect parameter settings and improper assumptions, will lead to errors in a model, and sensitivity analysis can help parse model behavior and identify potential problems. Sensitivity analysis methods have been used across various disciplines and can aid in validating and verifying a model (Christopher Frey and Patil, 2002). Sensitivity analysis includes analytical examination and is an essential aspect of model development (Hamby, 1995). Appropriate sensitivity analysis reduces model uncertainties and increases confidence in the model results. Our research aims to perform sensitivity analysis for a constructed simulation model to ensure its credibility and validity. The evaluation and validation

* Corresponding author.
** Corresponding author.
E-mail address: u3556305@connect.hku.hk (Y. Gong).

https://doi.org/10.1016/j.fsidi.2022.301449

processes will verify whether the model behavior effectively simulates Bitcoin transactions and whether the model outputs have uncertainties when changing important model parameters and settings. The analysis results show that the simulation model is stable, and model outputs can be used to validate heuristic-based address clustering algorithms. The rest of the paper is organized as follows. Section 2 overviews related research. Section 3 briefly introduces the simulation model construction. Section 4 demonstrates the evaluation and validation processes. The last section concludes the paper and outlines the research directions for future studies.

2. Related work

Address clustering groups Bitcoin addresses belonging to the same user into the same cluster, which is related to Bitcoin de-anonymization. First, heuristic-based address clustering exploits the structural details of transactions (Zhang et al., 2020). Meiklejohn et al. (2013) demonstrate a multi-input heuristic method. It is also called the common spending heuristic in (Ermilov et al., 2017). While making transactions, the private keys are required to create signatures. Thus, all the input addresses may belong to the same user for multi-input transactions. The one-time change address heuristic algorithm is also put forward by Meiklejohn et al. (2013). For transactions having changes, the change address and all input addresses are likely to belong to a single user. Zhang et al. (2020) have improved the one-time change address heuristic to identify more change addresses precisely, called the address reuse-based change address detection heuristic. Nick (2015) has developed the optimal change heuristic derived from Bitcoin client behavior and the consumer heuristic regarding redeeming transactions. The second type of address clustering method uses information other than transactions' structural details in different ways. Biryukov et al. (2014) associate Bitcoin users' pseudonyms behind firewalls of ISPs or NATs with the public IP address of the host through eight octets of Bitcoin peers (entry nodes). Ermilov et al. (2017) collect and analyze public information that can be found on the Internet (off-chain information) to create key phrase-entity (tags), which are divided into six categories. Combine heuristics and the off-chain information to generate clusters. Kang et al. (2020) construct a custom Bitcoin client to collect data and conduct a statistical analysis to explore reliable mappings between Bitcoin addresses and IP addresses. Zhu et al. (2017) design a system for analyzing blockchain data and network traffic data to find the correlation between Bitcoin address and IP address.

A simulation model close to the real Bitcoin network can help understand the behavior of the Bitcoin system. Based on design structures, there are two broad types of simulators (Alsahan et al., 2020). The first type of simulator is the event-based simulator. In such a system, state variables change in discrete times (Banks, 2005). BlockSim, developed by Faria et al. (Faria and Correia, 2019), includes many fundamental simulation models common to blockchains. And users can flexibly expand the framework to assess deployment determinations and different setups. Shadow Bitcoin belongs to this simulation type, capable of scalability and direct execution of multi-threaded applications (Miller and Jansen, 2015). Fattahi et al. (2020) have implemented an improved version of BlockSim, called SIMBA, by including Merkle tree features for effective transaction authentication and consistency. The second one is the virtualization-based simulation, which utilizes light-weight virtualization approaches. Chen et al. (2017) have created a framework including the Docker platform and containers, which reaches a good tradeoff between expense and model performance. With the introduction of virtualization into blockchain testing, several logical nodes running a custom Bitcoin application can model a large-scale P2P network. Alsahan et al. (2020) have proposed the simulation framework with lightweight virtualization, which supports quick simulation of large-scale networks. This framework incorporates the Linux kernel traffic control (tc) tool and allows different network topologies.

Sensitivity analysis is usually used to investigate how changes in specific parameters in a model affect the generated results. The analysis results may indicate whether the constructed model is a suitable representative of the corresponding real system. Within a neighborhood of validity, whether the simulation results can be proven to be consistent with the behavior of the related real system. For developed simulation models, sensitivity analysis can assist researchers in uncovering the effects of parameters on output variability and which parameters interact with each other, and reducing output uncertainties, among other aspects (Chan et al., 1997). The model parameter is an internal configuration variable. Improper parameter settings can result in errors in a model. Sargent (2010) point out that sensitivity analysis can determine the effect of parameter variability to examine the model's output behavior. It is one of the validation approaches that are widely used in model verification and validation. Iterations may be needed to ensure the sensitive parameters are sufficiently accurate. Sometimes it may be that optimal values for some parameters cannot be determined in a simulation model; sensitivity analysis can help define intervals in which the parameters are expected to lie (Murray-Smith, 2015). Christopher Frey and Patil (Christopher Frey and Patil, 2002) present the identification and qualitative comparison of sensitivity analysis methodologies utilized in different disciplines. Depending on the applicability to different types of models to use suitable approaches, sensitivity analysis is helpful for the identification of crucial control points and the validation of a model. Saltelli (2002) goes through some instances in which sensitivity analysis has played an essential role in model-based analysis and describes the methods that satisfy these requirements.

3. Bitcoin simulation model

According to the modeling process in (Murray-Smith, 2015), modeling should start with the target purpose, the prior knowledge, and relevant requirements analysis. These aspects significantly affect the choice of modeling methods, model structure and parameters. For Bitcoin heuristic-based address clustering, the multi-input and one-time change heuristics are two broadly used and representative heuristics (Mun et al., 2020). This Bitcoin simulation model aims to simulate real-world Bitcoin transactions to validate these two primary heuristic methods and assess their error rates. Because of the diversity of research objectives and design structures, most existing simulation models and measurements pay more attention to network-level implementations and propagation efficiency. The structural details inside the simulated Bitcoin transactions are not the focus. Thus, an appropriate simulation model is required for the objective.

3.1. Prior knowledge

Prior knowledge about the real system can considerably impact the model generated and the modeling techniques used (Murray-Smith, 2015). Therefore, the real Bitcoin system should be investigated before modeling. The investigation results can provide insights into the real Bitcoin system and be used later to observe the model behavior and compare the model outputs. The two heuristics depend on the number of input or output addresses and address reuse (Meiklejohn et al., 2013). As a result, the prior knowledge covers transaction distribution and address reuse. Blockchain data

from when Bitcoin was first used until the end of 2021 is parsed. There are 716,548 blocks (block height: 0–716547) and 699,285,797 transactions.

3.1.1. Transaction distribution

Transactions are not categorized into standard and non-standard transactions based on scripts (Bistarelli et al., 2019). Instead, transactions are divided into four categories according to the numbers of inputs and outputs: transfer, multiple payments, consolidation, and complex transactions (Cotten, 2018).

The distributions of the four types of transactions are shown in Fig. 1. Multiple payments constitute the most significant type. Figs. 2 and 3 present the distribution of different numbers of inputs/outputs in all transactions. Transactions with only one input have the largest number, accounting for 73.01%. Also, transactions having two outputs count for the highest proportion, accounting for 75.20% of all transactions. Regarding the transaction volume in each block, blocks with 1–500 Bitcoin transactions make up around one-half of all blocks.

3.1.2. Address reuse

An address is new if it emerges as an output address the first time. Conversely, the address is old if it appears as the output multiple times in the whole blockchain. New addresses are encouraged in each new transaction for user privacy; however, it is still typical to get payments with old Bitcoin addresses (Gaihre et al., 2018). Fig. 4 shows the address reuse rates during each year. In the first four years, the address reuse rate grew. After that, the reuse rate declined and fluctuated around 10% in the following years. As a result, a ten percent address reuse rate will be set for simulation.

3.2. Model structure

The model is based on Simchain (Simchain). In the real Bitcoin network, nodes can propagate, send and receive transactions and perform verification. The obtained ground truth should help validate the two widely used heuristics. Referring to the real bitcoin transaction structure (Antonopoulos, 2014), the simulated transaction details should cover information like txid, vin, vout, and used UTXOs. Therefore, in the Bitcoin simulation model, nodes are full nodes holding four functions: consensus, storage, wallet, and routing (Simchain). Bitcoin mixing services are for enhancing anonymity. When applying Bitcoin mixing, the linkages between input and output addresses are hard to derive (Wu et al., 2021). Thus, the model will generate transactions from multiple nodes to multiple nodes, similar to Bitcoin mixing services, for a more realistic simulation. Three log files (basicdata.log, bitcoin.log, and detaildata.log) record model logs and required simulation data. Details of the model structure and the generated simulation data can be found in (Gong et al., 2022).

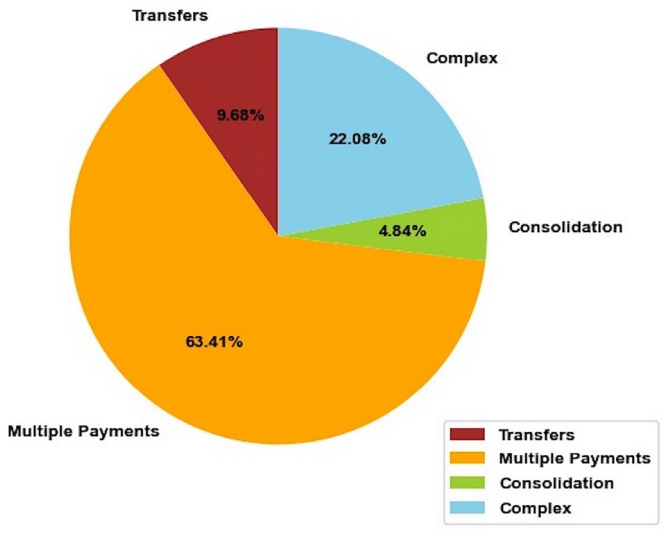

Fig. 1. Transaction Type Distribution (Bitcoin blockchain).

4. Evaluation and validation

Reasons like improper parameter settings and mistakes in experimental procedures can lead to uncertainties and errors in the models. After modeling, what is really examined when assessing the quality is whether the model is invalid under certain conditions. It is impossible to completely prove that a model is valid from all aspects (Murray-Smith, 2015). In addition, our understanding of the real system is never complete, and the ability of measurements and computations is limited. At best, we may be able to show that the developed model looks to be an acceptable representation of the real system for the specific objectives of the desired application (Murray-Smith, 2015).

4.1. Sensitivity analysis

Modeling includes parameter settings. The quality of a model should be questioned when the parameter variability within the predicted interval results in considerable variations in simulation results (Murray-Smith, 2015). Sensitivity analysis can deal with such situations. It can identify the critical parameters in the model and assess how model behavior and outputs are affected by parameter variability. Sensitivity analysis results are essential for model development and refinement and help improve the whole model's credibility.

4.1.1. Experimental setup

Generally, all input addresses of a multiple-input transaction should be from the same user, as related private keys are required to make the signature. With the development of mixing services, unrelated users can be grouped to create a single transaction. Therefore, what appears to be regular user transactions may utilize mixing services. In (Pakki et al., 2021), there are examples of transactions using mixers, and it can be seen that a transaction with a regular number of inputs and outputs (e.g., a transaction with one input and two outputs) can be related to a mixer. In the real Bitcoin network, it is not practical to identify the actual owners behind all addresses and calculate the proportions of mixing transactions in all Bitcoin transactions because of anonymity, etc.

In the constructed simulation model, functions to simulate mixing services are integrated, but the settings regarding the percentage of mixing transactions are uncertain. For transaction types with multiple inputs/outputs, i.e., Multiple Payments, Consolidation, and Complex, there are two different situations. For example, a consolidation transaction is assumed to have two inputs and one output. These two input addresses may come from the same user or two different users with mixing. Thus this transaction may be from one node to one node or from multiple nodes to one node, i.e., the one-to-one or one-to-many setting. Please note that there may be the case that the user sends Bitcoin to his/her another address, i.e., the sender and receiver are the same user. This situation is already considered in these two settings. All input addresses should belong to the same user from the aspect of the multi-input heuristic. When two inputs are from two different users, i.e., the one-to-many case, it causes a false positive. Therefore, the probability distributions of

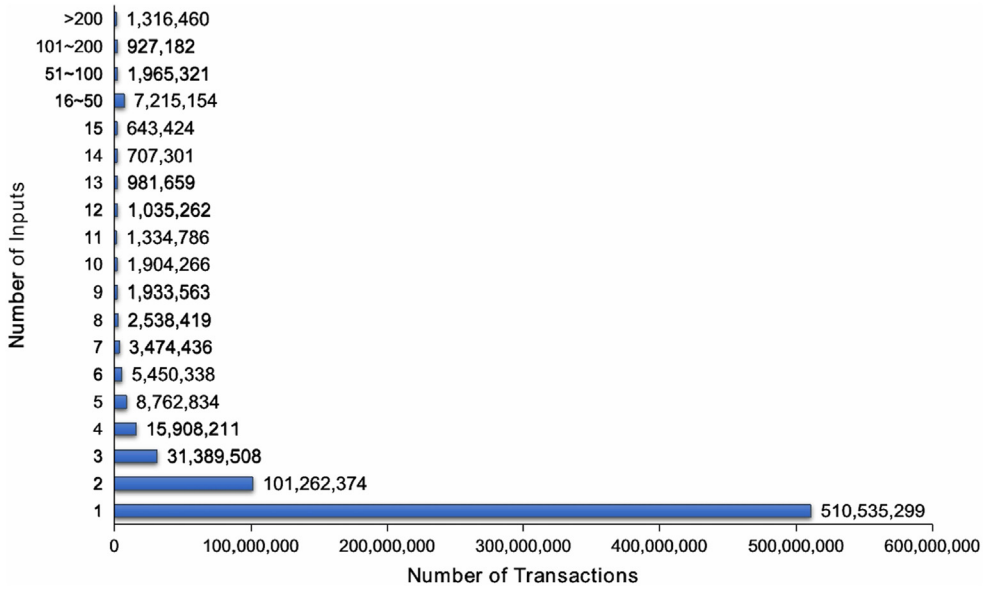

Fig. 2. Distribution of the numbers of inputs (Bitcoin blockchain).

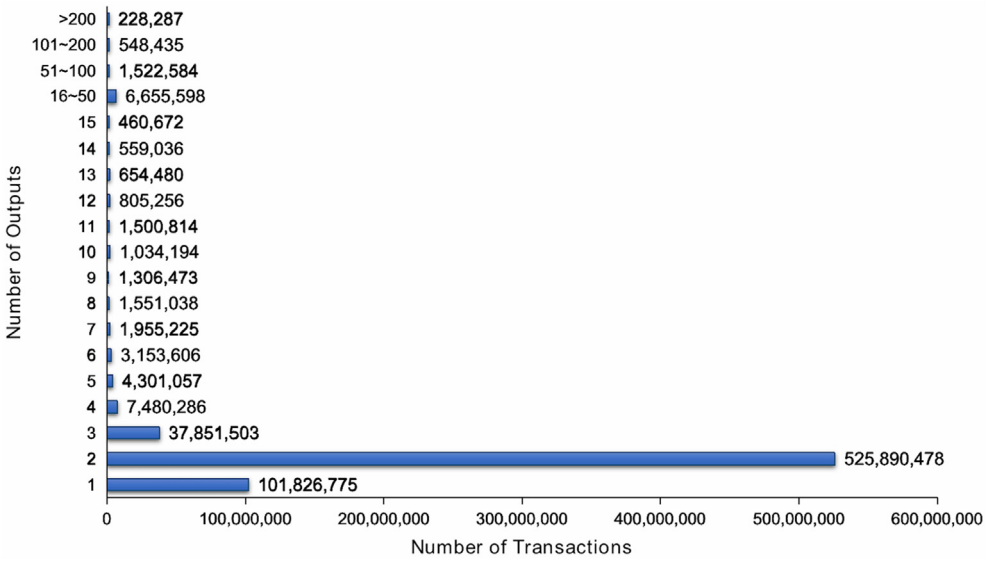

Fig. 3. Distribution of the numbers of outputs (Bitcoin blockchain).

these two settings may influence the error rates of heuristic algorithms.

In the real Bitcoin system, the percentage of mixing transactions is unknown. Since no reference is derived from the real system, the probability distributions for these two settings in the simulation model are also uncertain. The number of nodes in the simulated network and the total number of transactions generated can be set when simulating a Bitcoin network. As the real-world blockchain is dynamic and changing, the total transaction volume can be set, but the model will generate a transaction volume approximately equal to the set value. There is no fixed absolute generated transaction volume to simulate the real Bitcoin system as close as possible. Therefore, the sensitivity analysis needs to be performed on these three aspects, i.e., the probability distributions of the two settings, the number of nodes in the simulated network, and the total number of transactions generated. The effect of parameter changes on the simulation model can be studied by repeating simulation runs and directly comparing the model behavior before and after the changes (Murray-Smith, 2015).

The experiments can be divided into three sets: (i) repeated simulation runs with initial settings, (ii) simulation runs for different probability distribution settings for three transaction types having multiple inputs/outputs, and (iii) simulation runs for different numbers of nodes and generated transactions in the simulated network. Each node starts with a balance of 1,000 BTC in the simulated network. The transaction fee is 0.001% of the total input amount. As the number of miners in the real Bitcoin network is variable, the number of miners in the simulated network will be random. The winning miner receives a 6 BTC mining incentive. The number of transactions produced per block follows the probability distribution in the real blockchain. The initial probability distributions are 50%/50% in the simulation model for the one-to-one and one-to-many settings.

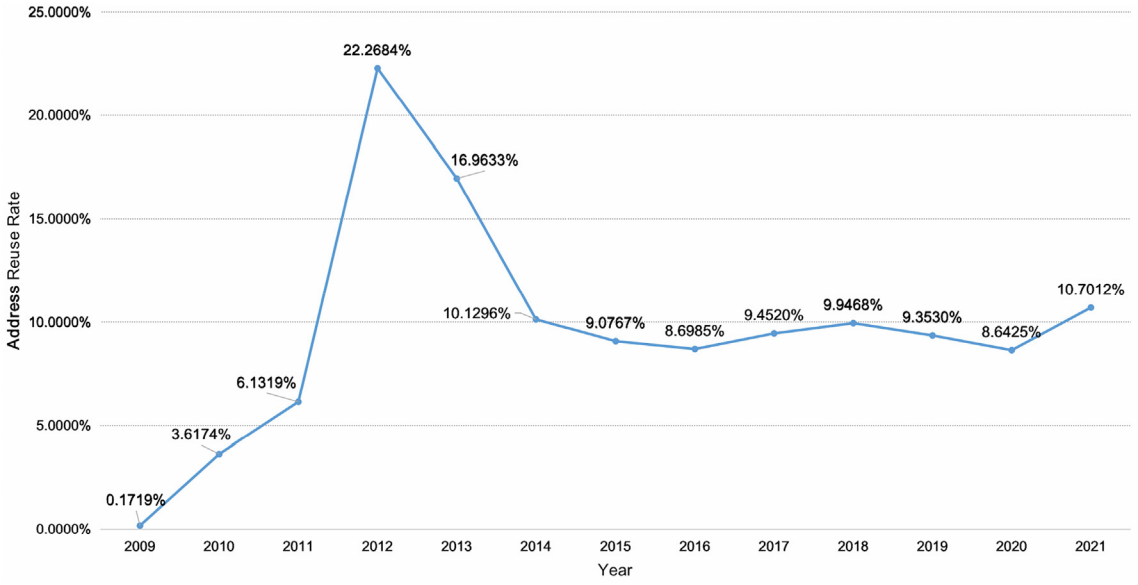

Fig. 4. Address reuse rates (Bitcoin blockchain).

Experimental group I. The simulation network includes 100 nodes and generates around 5,000 transactions. It can be seen as the initial control data for the model. Repeat simulation runs ten times to get ten sets of experimental data.

Experimental group II. Change the probability distribution settings by 5% each time. For three transaction types having multiple inputs/outputs, each has four sets of probability change trials (i.e., 60%/40%, 55%/45%, 45%/55%, 40%/60%) and one set of initial probability experimental results (50%/50%). In total, there are thirteen sets of experimental data.

Experimental group III. Set the total number of generated transactions to be 5,000, then simulate the Bitcoin network with 50/100/150/200 nodes separately. Set 100 simulated nodes and create 3,000/5,000/8,000/10,000 transactions individually in the simulated network. A total of eight sets of experimental results are generated.

4.1.2. Evaluation analysis

By applying the two heuristics separately and in combination to the simulation results, there are three heuristic methods (i.e., multi-input (MI) heuristic, one-time change (OTC) heuristic, and the multi-input and one-time change (MI + OTC) heuristics). The corresponding error rates can be obtained from real and heuristic clusters. H_i and R_i represent a heuristic and the corresponding real clusters, respectively. To get the matching accuracy, the number of correctly clustered addresses appearing in both clusters divides the number of addresses in the real cluster.

$$Accuracy_i = \frac{|H_i \cap R_i|}{|R_i|}. \tag{1}$$

The error rate can be derived from the results of cluster matching.

$$Error\ Rate_i = 1 - Accuracy_i. \tag{2}$$

The average error rate of a heuristic algorithm can be calculated if N is the total number of heuristic clusters.

$$Average\ Error\ Rate = \frac{1}{N}\sum_{i=1}^{N} Error\ Rate_i. \tag{3}$$

Details and explanations of the error rate calculation are in (Gong et al., 2022).

Experimental group I. Table 1 and Fig. 5 show the simulation results. Experimental group I tested whether the model could achieve a stable mode under the same conditions. Fig. 5 shows that although the error rates fluctuate slightly, values stay within a reasonable range, and the results are relatively stable. The real blockchain is constantly changing. In reality, various users create transactions that make up the whole Bitcoin blockchain. The transaction behavior of users is not static either. For example, user A transfers funds to an address of user B, and then user A may not make transactions or create transactions with more users the next day. More precisely, in two blocks with the same number of transactions, the distributions of the four transaction types and the number of inputs/outputs may not be entirely the same. Even if the same number of nodes are set in the model and the same total number of transactions are generated, the blockchain patterns and generated data may not be the same in each simulation. Consequently, the simulation results may not necessarily yield the same error rate values. The error rate values are acceptable to keep relatively stable within a reasonable range even with minor fluctuations. The simulation model can generate relatively stable experimental results under the initial settings.

The average error rate of the one-time change heuristic is 90.3612%, and the average error rate for the multi-input heuristic is 46.5040%. A combination of multi-input and one-time change heuristics reaches the lowest average error rate, 41.2528%.

Table 1
Simulation results for experimental group I.

Group	Num of TX	Address reuse rate
Group I	4908	9.7498%
Group II	4916	10.1265%
Group III	4942	9.5222%
Group IV	4887	9.8225%
Group V	4892	9.7498%
Group VI	4899	9.8425%
Group VII	4888	9.7402%
Group VIII	4907	9.9387%
Group IX	4912	9.5980%
Group X	4914	9.6877%