星云链技术白皮书

Nebulas Inc.

August 8, 2017

Abstract

白皮书的摘要，让大家看完摘要后，就对星云链要做的事情一目了然。
TODO 是蓝色
注释是红色
引用的例子，如下 GNU pthread [25], PARSEC [6]

Contents

1 简介

1.1 区块链技术简介

简单介绍一下区块链技术的概要，和最流行的开源区块链技术特点。

1.2 商业和技术挑战

当前公有链遇到的技术挑战：智能合约无法被简单搜索；用户以及智能合约的价值贡献无法被评估和计算；PoW 共识速度慢，消耗大量电能，PoS 共识使得拥有大量代币资产的用户获得记账资格，形成马太效应，带来另外的不公平忧虑；目前的激励机制仅仅奖励矿工，有一定的局限性，不能鼓励优秀的智能合约开发者；智能合约一经部署，无法升级，不适应普通的企业应用需求；区块链核心协议如果需要升级，容易引起硬分叉，对区块链社区带来伤害；

1.3 星云链解决的问题

介绍星云链的贡献：NR 排序算法，核心协议的可升级、PoR 共识（出块速度和安全性）、DIP 开发者激励、智能合约升级设计、

1.4 星云链设计基本原则

阐述星云链设计的基本原则：完全兼容以太坊、为区块链设计价值尺度、更快更合理的共识设计、对区块链生态的有效激励、图灵完备可升级的智能合约、可自我演化的区块链、更好用的区块链基础服务。

2　星云链代币 NAS

robin

3 账号和地址

类似于比特币和以太坊，星云链的帐号也是由椭圆曲线算法作为其基础加密算法。用户的私钥是随机生成的 256 位二进制数，通过椭圆曲线乘法生成 64 字节的公钥。比特币和以太坊的地址都是由公钥通过确定性的哈希算法运算得到，不同的是：比特币地址有校验码设计，防止用户输错了几个字符而把比特币误发给其它用户，以太坊却没有校验码设计。我们认为从用户的角度看，校验码的设计是合理的，因此星云链的地址也包含了校验码，具体计算方法如下：

```
Data = sha3_256(Public Key)[-20:]
CheckSum = sha3_256(Data)[0:4]
Address = "0x" + Data + CheckSum
```

公钥的 SHA3-256 摘要的后 20 位字节作为地址的主要组成部分，对其再做一次 SHA3-256 摘要，取前四位字节做校验码，相当于以太坊的地址尾部加上四字节的校验码。以太坊地址包括 0x 前缀共 42 个字符，星云链地址共 50 个字符，增加了 8 个字符的校验码。除了支持标准的 50 字符的标准地址，为了用户转账的安全，我们还支持扩展地址。借鉴了传统银行转账的设计：转账时除了验证对方的银行卡账号，汇款人还必须输入对方的姓名，只有银行卡账号和姓名都匹配，转账才能正确进行。扩展地址的生成算法如下：

```
Address = "0x" + Data + CheckSum
Ext = sha3_256(Data + nick)[0:2]
ExtAddress = Address + Ext
```

扩展地址在标准地址的尾部追加了 2 字节的扩展验证,共 54 个字符。加入扩展信息,使得在星云链钱包应用中可以增加另一个要素验证,比如: Alice 的钱包标准地址是 0x5d65d971895edc438f465c17db6992698a52318d5c17db69,加入"alice"后的扩展地址是 0x5d65d971895edc438f465c17db6992698a52318d5c17db69aabb; Alice 告诉 Bob 扩展地址 0x5d65d971895edc438f465c17db6992698a52318d5c17db69aabb 和 alice; Bob 在 Wallet 应用中，输入 0x5d65d971895edc438f465c17db6992698a52318d5c17db69aabb 和 alice。Wallet 应用验证钱包地址的一致性，避免 Bob 错误的输成了其它人的帐号。标准地址和扩展地址均可以用于转账，我们的星云链钱包应用将会强制使用扩展地址，转账时需要提供对方的昵称，验证地址一致性，进一步加强安全校验。

4 区块和交易

Wenbo

5 Nebulas Rank 算法

5.1 Introduction

Ranking nodes in complex network has been a fundamental concern in various applications. One canonical example is PageRank[12][47], which is the core algorithm for Google and other search engines[34]. Besides, by ranking algorithm, people also want to find out the most influential spreaders in epidemic and information network[17][31], the most acknowledged scientist by the citation network or co-author network[59][16][50], the most important cities in the transportation network[26], the most important vertices in metabolic network[28], the top VC firms by co-investment[5] etc. And when it comes to designing **Nebulas Rank** algorithm for blockchain world, decades of researches have enlightened us with many measurements like degree centrality[21], eigenvector centrality[7], Katz centrality[29], PageRank[12], HITS[32], closeness centrality[52], betweenness centrality[22][23][24][45][42], etc. Before building our algorithm on the top of them, however, we still need to answer two questions:

1. What properties are embedded in the network?
2. What value should the rank indicate?

For the first question, **Nebulas Rank** uses the transaction graph of blockchain system, which is generated by the transaction history during the past period. We compare blockchain transaction graph with others in three aspects. First, as node representing accounts and edges representing money transferring, basically, the graph is a weighted directed graph, bearing large difference from social network[58] and similarity with webpage network[47] in terms of topology. An asymmetric edge indicates the imbalanced ability of controlling money between two nodes. And edges' weight also characterizes the difference among links' quality. Thus directly applying standard algorithms for unweighted or undirected graph could discard a lot important information. Second, since the graph is derived by the trajectory of money flow, it could be presumed that Exchanges are highly ranked, whereas such accounts are willing to swap money with any client. Thus anyone can acquire unlimited links from those existing important nodes without much cost. Along with the anonymity of typical blockchain systems, sybil attackers could make large amount of transferring with a big Exchange account, in order to improve his influence such as PageRank. (It is still hard to receive money from a large number of non-sybil nodes, though.) This is an essentially difference from applications previously studied. For example, in an online social network with subscription relation, it costs no effort to follow an opinion leader, while attracting a verified opinion leader's attention is not an trivial thing. Third, being different from Bitcoin[39], the newly invented blockchain systems such as Ethereum[60] introduces "smart contract" as a new type of account. After a normal account invoking some method of a contract, a sequence of consequent callings will be raised forming part of a call graph. Unlike Bitcoin transaction graph, which only contains money transferring, Ethereum call graph/network also represents dynamic programming calling. We believe such network embeds more information and should be useful to measure a DApp or smart contract's value.

For the second question, **Nebulas Rank** aims to measure the value of users, and smart contracts in blockchains. For normal accounts, we define value by two aspects: **Liquidity**, which stresses the ability to control digital assets flow of high quality; **Propagation**, which focus more on the spreading influence. For smart contracts, we also consider **Interoperability** as a measurement. There are three-fold purposes in **Nebulas Rank**: 1) to be a good metric for blockchain accounts and smart contract search engine; 2) to provide a trustful criteria in PoR consensus protocol (see §**??**) , where only high ranked accounts should be eligible to become a validator; 3) to help to build DIP mechanism (see §**??**). [1].

More clearly, **Nebulas Rank** should be based on the formal concept of network flow. As revealed in [8], most centralities can be classified by their type of network flow. From the dimension of diffusion mechanism, traffic flow can spread by different kinds of duplication or transfer. Another spectrum is the trajectory of flow, which can be either paths, trails or walks. Essentially, blockchain transaction graph is the trajectory of money exchange, which falls into the classification of "transfer walk". Imagine an amount of money enters the network. Then the owner node divides the money and transfers to neighbors or keep it. To clarify, the money is divisible, imperishable and non-replicable and each step's direction is random due to the limit of local information. This rules out some measurements for us. For example, Freeman [22]'s betweenness centrality, since it implies geodesic optimal paths.

[1] Note the in this section, our transaction data don't include smart contracts. More about DIP is described in §**??**

Following are our solutions to the challenges described above.

First, in order to turn transactions into a graph, we keep the transfer value as edges' weight and embed the transfer timestamp into nodes' coinage property. Then we exploit coinage and other properties to fix the edge weight:

- We set a time window of T days and aggregate the largest K transactions as an edge's weight. (see §5.3.2;
- We reduce each edge's weight by its target's "coinage". In order to get higher coinage, an owner needs to let the money stay in place for a while, which slows down the speed of sybil attacks such as loop attack. (see §5.3.3)
- Edge weight is also reduced by its target's outgoing amount(see §5.3.6) as well as an encouragement function (see §5.3.4), which helps to mitigate some undesired effect.

Second, with graph generated, we measure each node's importance based on Weighted LeaderRank algorithm[15][35]. LeaderRank is a simple variant of PageRank, which adds a ground node into the network and connect the ground node with each non-ground node. It substitutes PageRank's teleportation parameter by links throughout ground node, which is said to be more effective in computing, robust against manipulations and noise than PageRank algorithm[15]. The intuition for both LeaderRank and PageRank is random walk and Markov Chain. By PageRank, from each node, the probability of jumping to an arbitrary node is the same (or equal to 1 if there is no out links [30]). Whereas by LeaderRank[35][15], different nodes adopt different arbitrary transition probabilities. For example, we could allow a node with more in-links to receive more teleportation probability. This is more plausible in the context of blockchain, since an account with little money transferred in is less trustable. Also, if a node receives much money but hardly spend it, we suppose it has more "surplus value" and assign with more teleportation probability from it. Intuitively, LeaderRank indicates the flux over a node in the dynamic equilibrium of money exchange network flow. From another perspective, mode flux means more control. The LeaderRanks algorithm matches both our goals of measuring **Liquidity** and **Propagation**. We will talk about the details of LeaderRank scheme in §5.4.

~~Our experiment results shows that ......~~

The rest of paper is organized as follows. §5.2 introduces the related works. Then in §5.3, we define the network topology and weight based on blockchain transactions. And in §5.4, LeaderRank with schemes designed for **Nebulas Rank** is introduced. In §5.5, we show our experiment results. And Finally we give all discussions and conclusions in §5.6

5.2 Related Works

Centrality, the core ranking index, is a most studied concept in network science since decades ago[41]. There are a body of literatures introducing various centralities, including degree centrality[21], eigenvector centrality[7], Katz centrality[29], closeness centrality[52], betweenness centrality[22][23][24][45][42], PageRank[12], HITS[32], SALSA[53], etc. It is fundamental to clearly classify these measurements by a unified framework. Borgatti [8] adopts a network flow based view to classify the centrality measurements by two categorical dimensions: material flowing by parallel duplication, serial duplication and transfer; and trajectory following geodesics optimum, path, trail and walk. Borgatti and Everett [9] propose a unified framework with four dimensions from the perspective of graph theory. Lü *et al.* [36] review representative centrality algorithms and classified them into those only based on structural information, those driven by Markov dynamics, those by looking at the effect of removing nodes, those with dynamics-sensitivity and those trying to identify more than one node. With a hierarchical understanding of centrality algorithms, we are able to choose appropriate strategy according to the network scenarios. **Nebulas Rank**'s scenario is the money exchange flow network mentioned in [8].

Since Bitcoin[39] system released in 2009, researchers have done some statistical and empirical analysis on Bitcoin's transaction graph[51][27][43][3], and some use the transaction graph structure to discuss anonymity in Bitcoin[37][46][48][20][19]. After other cryptocurrencies emerged and become popular, transaction graph analysis is conducted with more blockchains[14][1]. **Nebulas Rank** adopts their transaction graph concept, i.e. Entity Graph in [57], with minor revisions. That is, each account, or set of accounts belonging to the same people, is mapped as a node. And each directed edge represents the intensity of transferring between two accounts. Actually before blockchain system like Bitcoin was invented, scientists have tried to study some financial networks among banks and global trading entities[49][11][54][4][18][38][10][33][55]. Comparing with blockchain transaction networks, these early studied finical networks are defined not only by transferring activities, but also by lending-based relationship. Moreover, the scale of these networks is much smaller. To conclude, there is rarely research work proposing custom ranking method for large scale transaction graph, especially blockchain transaction graph.

The most relevant work with **Nebulas Rank** is NEM[40]'s Proof-of-Importance scheme. It adopts NCDawareRank[44], which exploits the clustering effect of network topology, as the ranking algorithm, with clustering algorithm based on SCAN algorithm[61][56][13]. And Fleder, Kester, and Pillai [20] uses PageRank[12][47] as an assisting metric to discover interesting addresses and analyze their activities. However, both NCDawareRank and PageRank are ranking algorithms for webpage network. As we already mentioned in §5.1, blockchain transaction graph is very different from webpage network. And although community structure does exist in transaction graph and should be helpful to handle with spam nodes, it does not suit the consensus purpose mentioned in §5.1. Because in order to compute "unforgeable" node importance, accounts controlled by a single "real world" entity should be guaranteed to be mapped to the same cluster. However it is key difficulty to connect blockchain world with the "real world", and thus there is no proper objective definition for clustering problem. Therefore current clustering algorithms cannot provide meaningful and trustful result. Moreover, [20]'s work does not provide an automated framework to identify important nodes. Instead, with the help of PageRank, it still take manual analysis as core method, which does not match **Nebulas Rank**'s context.

The algorithm we choose is LeaderRank[15][35]. It is a simple variant of PageRank[12][47]. In PageRank, initially every node gets one unit rank value. Then at each iteration, every node distributes its rank value equally to its directed neighbors. To deal with dangling node problem, there is a damping factor, where every node distribute a specific proportion of its rank value to all nodes equally in the network. Chen *et al.* [15] propose a simple yet effective modification on PageRank's damping factor and call it LeaderRank. Then Li *et al.* [35] extend LeaderRank to weighted case and further improve its performance. By weighted LeaderRank[35], an additional ground node is added and a bidirectional link is added between every node and ground node. Every edge targeting to ground node is of same weight and every edge from ground node is weighed positively proportional to target node's in-degree. LeaderRank is more resistant against manipulation and noisy data than PageRank[15][35][36]. In terms of computation, LeaderRank can be seen as PageRank with one more node and set damping factor to be zero. And thus it is easy to implement and very scalable.

Other than LeaderRank, there are also other algorithms modifying PageRank's damping factor mechanism. For example, Baeza-Yates, Boldi, and Castillo [2] proposed a damping function decreasing with distance. Besides, there are some betweenness based algorithms mentioned in [8], such as flow betweenness[24] and random walk betweenness (aka. current flow betweenness)[42]. Although they may be more suitable to represent the flow controlling ability, these centralities are quite computational intensive. So they are not suitable for **Nebulas Rank**. Out of all the current algorithms, we think LeaderRank is a relatively simple yet effective one.

5.3 Transaction Graph

5.3.1 Transactions

The input data for **Nebulas Rank** are all the transaction records, i.e. token transferring, during the past T days, denoted by a set of tuples:

$$T_{xs}^{all} = \{(s,t,\tau,a), \tau = Today - T \dots Today\}$$

, where s, t and a are the source account, target account and amount of an transfer, respectively.

Further, we filter transactions, providing that self transfer and zero amount transfer are excluded:

$$T_{xs} = \{(s,t,\tau,a) | s \neq t \wedge a > \Phi \wedge (s,t,\tau,a) \in T_{xs}^{all}\}, \Phi = 0 \tag{1}$$

5.3.2 Transactions Aggregation

Based on transactions defined above, we construct the directed weighted transaction graph $G = (V, E, W)$, where node set, edge set and weight on edges are denoted by V, E and W respectively. We also denote that $N = |V|$ and $M = |E|$. For simplicity, all nodes are denoted by integer numbers from 1 to N.

Each vertex $v \in V$ represents one individual account's address. Each edge represents the transferring intensity between two accounts. Consider $e = (s,t) \in E$, this edge is directed, and naturally, the weight of it should be determined by all related transactions, i.e. $(s,t,\tau,a) \in T_{xs}$. To compute edge (s,t)'s weight, we take the sum of top K amounts out of all related transactions:

$$w_e = \sum_{i=1}^{K} a_i, s.t. a_i \in \{a|(s,t,\tau,a) \in T_{xs}\} \wedge a_1 \geq a_2 \ldots \tag{2}$$

By this mean, the link between two nodes is bi-directed and asymmetric, with top K transactions along each direction aggregated to become the weight. This is different from NEM, which all transfers amounts between two nodes are aggregated into one unilateral edge's weight[40]. We presume NEM's solution is vulnerable to manipulations, since only a simple triangle loop will enhance edges' weight into infinity. Additionally, it is also not truthful to take the average or quartile of all relation transactions, since this forces accounts to transfer very cautiously. We will show the advantage of our aggregation method in §**??** ~~experiment confirming why doing so~~

5.3.3 Temporality Embedding

We noticed that the transactions happens with timestamps. So we try to embed this temporal information as a property of nodes. For each account, we calculate its coinage by the following pseudo-code.

~~formula~~ ~~Defined as C_v normalized by max~~ coinage 的公式和解释

The intuition of coinage is ~~insights~~.

Besides, we conjecture that reducing each transaction's contribution according to its block height, like NEM does[40], encourages users to postpone their transferring until the last day of period, which will cause unnecessary confusion. Instead, **Nebulas Rank** treats each transaction equally, which encourages every account to keep active all the time.

We will talk about the coinage exploitation in §5.3.5. And we will show the advantage of our solution in §**??** ~~experiments confirming the effect of coinage~~.

5.3.4 Encouragement Function

~~formula and intuition~~ ~~defined as B_v normalized by max~~ encouragement function 的公式和解释

We will talk about how we apply the encouragement function in §5.3.5. And we will show its advantage in §**??** ~~experiments confirming the effect of encouragement function~~.

5.3.5 Exploiting Nodes' Property

We defined two node properties C_v and B_v in §5.3.3 and §5.3.4, respectively. Then we reduce each edge's weight by its target node's properties:

$$w_{(.,v)} \leftarrow w_{(.,v)} \times ln(1 + \frac{C_v + B_v}{2}) \tag{3}$$

5.3.6 Mitigating Dormant Effect

Consider a node receiving a large amount of money but does not spend any. This node forces its money to be "dormant " and prevents money from being circulated, which contradicts with **Nebulas Rank**'s purpose(§5.1). Thus we need to mitigate this dormant effect. In detail, we consider 1-hop local information of each node, limiting the amount of its in-transfers by the total amount of its out-transfers[2]:

$$w_{(.,v)} \leftarrow \frac{w_{(.,v)}}{\sum_u (w_{(u,v)})} min\{\sum_u w_{(v,u)}, \sum_u w_{(u,v)}\}. \tag{4}$$

Intuitively, such restricting method is reasonable: 1) Imagine two phases for a piece of blockchain token. First it is made out of thin air, which is as the reward of the system. Second it is either circulated around the whole network, which almost never stops being transferred from accounts to accounts, or

[2] Note that formula 4 is applied after formula 3. Formula 3 is applied after definition 2

活跃币龄(Active Coinage)

活跃币龄：在当前周期内，某个地址新接收和新发出的币为活跃币，这些币的币龄为该地址的活跃币龄

```
def calculate(cur, genesis, nodes):
    for node in nodes:
        # cal coinage according to the active coins
        if node.balance < 0:
            coinage = (genesis.no - cur.no) * node.balance
            node.coinage += coinage
            node.balance = 0
        else:
            node.coinage += node.balance

def cal_coinage(nodes, transactions):
    # first block
    genesis = transactions[0].block
    for trans in transactions:
        # next block
        if cur.no < trans.block.no
            # cal coinage till current block
            calculate(cur.no, genesis.no, nodes)
            cur = trans.block
        # record balance
        edge.from.balance -= edge.coins
        edge.to.balance += edge.coins
    # cal coinage till current block
    calculate(cur, genesis, nodes)
```

enters dormant state, which the last owner does not spend it out. Formula 4 does not affect the first phase, as edges weights can only be reduced as in-links. And in the second phase, accounts are encouraged to spend enough money in order to improve their in-link quality. 2) From the perspective of money flow, only circulated money should be counted. Nodes with dormant money does not control much of the network flow. That is, deleting these nodes does not affect interactions among other nodes. So formula 4 conforms with **Nebulas Rank**'s Liquidity value (§5.1).

We will show the how **Nebulas Rank** benefits from mitigating dormant effect in §**??**.

~~will this affect wgc????~~

5.4 LeaderRank

5.4.1 LeaderRank Weighting Scheme

We build our scoring algorithm based on LeaderRank[35][15]. It does minor modification on the famous PageRank algorithm[12][47]. The modification is to add a ground node into the network, in place of PageRank's damping factor. Our method is as follows.

We add one ground node $\mathcal{G}$, numbered as $N+1$, into the network, and double link it with every other node. First every non-ground node sends and receives an amount of "Altruist" money to the the ground node, denoted by A_v. Then every non-ground receives an amount of "Charity" money from the ground node, denoted by C_v. Besides, each node sends an amount of "Surplus" money, denoted by S_v, to the ground node and receives an amount of "Bonus" money, denoted by B_v, from the ground node. Formally, the weighting scheme is given by formula 5 and 6:

$$\forall v \in V, w_{(v,\mathcal{G})} \leftarrow \alpha A_v + \mu S_v \tag{5}$$

$$\forall v \in V, w_{(\mathcal{G},v)} \leftarrow \beta C_v + \lambda B_v \tag{6}$$

We define the altruist and charity money for each node to be equal. And roughly, they should be proportional to the average transaction amount of all nodes:

$$\forall v \in V, A_v = \frac{\sum_{e \in E} w_e}{N}, C_v = \frac{\sum_{e \in E} w_e}{N} \tag{7}$$

Next we define "Surplus" money as the incoming amount of one node minus its outgoing amount:

$$\forall v \in V, S_v \leftarrow \sum_{(u,v) \in E} w_{(u,v)} - \sum_{(v,u) \in E} w_{(v,u)} \tag{8}$$

The intuition is to fix §5.3.6's side effect: after edges' weight are reduced by its targets' dormant effect, there are some nodes whose outgoing amount is less than their incoming amount. Theses nodes don't bring about dormant effect themselves though, they transferred money to some neighbors with dormant money. We suppose these nodes contain "surplus" values, which should not be kept by themselves. Thus all nodes should transfer their "surplus" value to the ground node. This is an extension of Li *et al.* [35]'s weighted LeaderRank algorithm.

And we define "Bonus" money as the total amount transferring to the node:

$$\forall v \in V, B_v \leftarrow \sum_{(u,v) \in E} w_{(u,v)} \tag{9}$$

The Intuition is that nodes with mode incoming transfers should have higher probability to receive money from ground node. This is the same scheme as Li *et al.* [35] designed.

With the ground node and corresponding edges added, the ranking algorithm can be understood as a Markov chain. States are nodes. Transition probability is proportional to the weight of some node's out-edge. It can be described by an iterative process. Initially all node's rank score is the same except ground node, then every node distribute their rank score among their neighbors:

$$p_i^{t+1} = \sum_u p_j^t \times \frac{w_{(j,i)}}{\sum_k w_{(j,k)}}; p_i^0 = \frac{1}{N} \tag{10}$$

, where p_i^t is node i's rank at the end of t-th iteration.

Equivalently, with the form of matrices,

$$P^{t+1} = H \times R^t; P^1 = [\frac{1}{N}, \frac{1}{N}, \ldots, \frac{1}{N}, 0]^T \tag{11}$$

$P^t \in \mathbb{R}^{N+1}$, represents the rank score for all nodes. And H is a $(N+1) \times (N+1)$ matrix representing transition probabilities of a Markov chain. The element in i-th row and j-column is the probability that a random walk hops to node i from node j, computed as

$$h_{ij} = \frac{w_{(j,i)}}{\sum_k w_{(j,k)}} \tag{12}$$

Since every node is connected with $\mathcal{G}$, the sum of each column of H is equal to one. The convergence of LeaderRank algorithm can be calculated by power iteration. Literatures[35][15] show more mathematical details.

After convergence, we get P^*, then distribute ground node's rank among every other node evenly:

$$\forall v \in V, P_v^* \leftarrow P_v^* + \frac{P_{\mathcal{G}}^*}{N} \tag{13}$$

5.4.2 Comparing with PageRank

Comparing with PageRank, we think LeaderRank is more reasonable in the context of **Nebulas Rank**. LeaderRank replace PageRank's teleportation parameter[12][47] with ground node mechanism. Teleportation parameter cannot be explained directly from perspective of network flow, while ground node is more understandable.

On one hand, LeaderRank actually enable us to assign different "teleportation parameter" for each node. On the other hand, in the sense of money flow, by PageRank, each node contributes a proportion of their income to the public, and receives the same amount from it. So PageRank adopts a different ground node weighting scheme from our ranking algorithm. And by evenly distributed arbitrary "surfing" probability, PageRank's scheme grants nodes earning lower income with "friendly" ranking. But since **Nebulas Rank** aims to provide some truthfulness, a nodes with lower "income" is more likely to be a sybil node and thus should be devalued by some more conservative ranking algorithm. There is same problem in NCDawareRank[44], which is also friendly to new nodes with less incoming edges. Together with survey in §5.2, we can conclude that original PageRank and NCDawareRank are not suitable for blockchain transaction graph. This challenges some of the previous studies[20][40].

5.5 Experiments

5.5.1 Ethereum Stats

~~degree - avg neighbor degree and dynamics; hhi~~

5.5.2 Exploitability of 1-hop Local Information

5.5.3 Noise Resistance

5.5.4 Sybil Attack Resistance

1. top k vs total
2. coinage vs no coinage
3. no reduction by date vs reduced
4. encouragement vs no encouragement
5. no dormant vs dormant
6. leaderRank with all vs others

1. loop

2. star
3. loop star including exchanges
4. random send money and exchanges
5. random graph including exchanges

~~all with comparison~~

5.6 Discussions And Conclusions

6 PoR 共识算法

jingchan

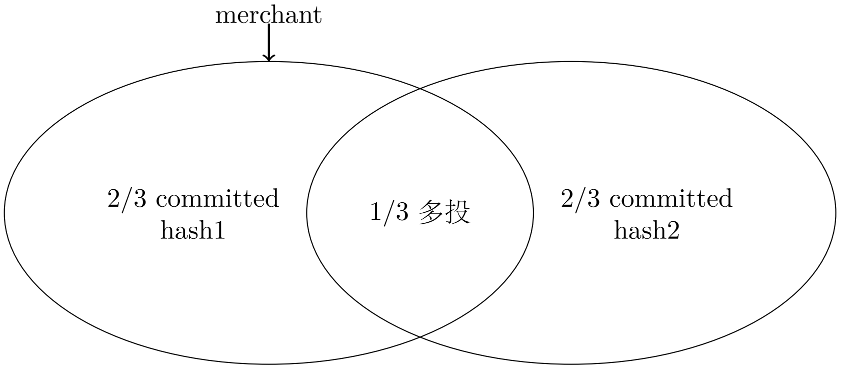

Figure 1: 双重支付攻击示意图

7 DIP 开发者激励协议

Shangshu

8 Nebulas Force

wenbo

9　智能合约

Wenbo

10 星云链基础服务及开发工具

References

[1] Luke Anderson *et al.* “New kids on the block: an analysis of modern blockchains.” In: (2016). arXiv: `1606.06530`. URL: `http://arxiv.org/abs/1606.06530`.

[2] R Baeza-Yates, P Boldi, and C Castillo. “Generalizing pagerank: Damping functions for link-based ranking algorithms.” In: *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (2006), pp. 308–315. DOI: `10.1145/1148170.1148225`.

[3] Annika Baumann, Benjamin Fabian, and Matthias Lischke. “Exploring the Bitcoin network.” In: *WEBIST 2014 - Proceedings of the 10th International Conference on Web Information Systems and Technologies* 1 (2014), pp. 369–374. ISSN: 9789897580239 (ISBN). DOI: `10.5220/0004937303690374`. URL: `https://www.engineeringvillage.com/blog/document.url?mid=cpx%7B%5C_%7D9ce5505146fd48dcbdM557010178163125%7B%5C&%7Ddatabase=cpx`.

[4] Morten L Bech and Enghin Atalay. “The topology of the Federal Funds markets.” In: *Economic Policy Review* 14.2 (2008).

[5] Harish S Bhat and Bryan Sims. “InvestorRank and an Inverse Problem for PageRank.” In: *PhD dissertation, University of California* (2012).

[6] C. Bienia and K. Li. “Parsec 2.0: A new benchmark suite for chipmultiprocessors.” In: *Proceedings of the 5th Annual Workshop on Modeling Benchmarking and Simulation.* 2009.

[7] Phillip Bonacich. “Factoring and weighting approaches to status scores and clique identification.” In: *Journal of Mathematical Sociology* 2.1 (1972), pp. 113–120.

[8] Stephen P. Borgatti. “Centrality and network flow.” In: *Social Networks* 27.1 (2005), pp. 55–71. ISSN: 03788733. DOI: `10.1016/j.socnet.2004.11.008`.

[9] Stephen P. Borgatti and Martin G. Everett. “A Graph-theoretic perspective on centrality.” In: *Social Networks* 28.4 (2006), pp. 466–484. ISSN: 03788733. DOI: `10.1016/j.socnet.2005.11.005`.

[10] Michael Boss, Martin Summer, and Stefan Thurner. “Contagion Flow Through Banking Networks.” In: *Lecture Notes in Computer Science 3038* (2004), pp. 1070–1077. ISSN: 03029743. DOI: `10.1016/j.jfi.2008.06.003`. arXiv: `0403167v1 [arXiv:cond-mat]`.

[11] Michael Boss *et al.* “The Network Topology of the Interbank Market.” In: *Quantitative Finance* 4.6 (2004), pp. 677–684. ISSN: 1469-7688. DOI: `10.1080/14697680400020325`. arXiv: `0309582 [cond-mat]`. URL: `http://arxiv.org/abs/cond-mat/0309582`.

[12] Sergey Brin and Lawrence Page. “Reprint of: The anatomy of a large-scale hypertextual web search engine.” In: *Computer Networks* 56.18 (2012), pp. 3825–3833. ISSN: 13891286. DOI: `10.1016/j.comnet.2012.10.007`. arXiv: `1111.6189v1`.

[13] Lijun Chang *et al.* “pSCAN: Fast and Exact Structural Graph Clustering.” In: *IEEE Transactions on Knowledge and Data Engineering* 29.2 (2017), pp. 387–401.

[14] Tao Hung Chang and Davor Svetinovic. “Data Analysis of Digital Currency Networks: Namecoin Case Study.” In: *Proceedings of the IEEE International Conference on Engineering of Complex Computer Systems, ICECCS* (2017), pp. 122–125. DOI: `10.1109/ICECCS.2016.023`.

[15] Duan Bing Chen *et al.* “Identifying influential nodes in large-scale directed networks: The role of clustering.” In: *PLoS ONE* 8.10 (2013), pp. 1–10. ISSN: 19326203. DOI: `10.1371/journal.pone.0077455`.

[16] Peng Chen *et al.* “Finding scientific gems with Google’s PageRank algorithm.” In: *Journal of Informetrics* 1.1 (2007), pp. 8–15.

[17] Benjamin Doerr, Mahmoud Fouz, and Tobias Friedrich. “Why rumors spread so quickly in social networks.” In: *Communications of the ACM* 55.6 (2012), pp. 70–75.

[18] Giorgio Fagiolo. “The International-Trade Network: Gravity Equations and Topological Properties.” In: (2009). arXiv: 0908.2086. URL: http://arxiv.org/abs/0908.2086.

[19] Danno Ferrin. “A Preliminary Field Guide for Bitcoin Transaction Patterns.” In: *Texas Bitcoin Conference* (2015). URL: http://texasbitcoinconference.com.

[20] Michael Fleder, Michael S. Kester, and Sudeep Pillai. “Bitcoin Transaction Graph Analysis.” In: *… -Transaction-Graph-Analysis. Pdf” …* (2015), pp. 1–8. arXiv: 1502.01657. URL: http://arxiv.org/abs/1502.0165%7B%5C%%7D5Cnhttp://people.csail.mit.edu/spillai/data/papers/bitcoin-project-paper.pdf%7B%5C%%7D5Cnhttp://arxiv.org/abs/1502.00165%7B%5C%%7D5Cnhttp://arxiv.org/abs/1502.01657.

[21] L Freeman. “A set of measures of centrality: I. Conceptual clarification.” In: *Soc. Networks* 1 (1979), pp. 215–239.

[22] Linton C Freeman. “A set of measures of centrality based on betweenness.” In: *Sociometry* (1977), pp. 35–41.

[23] Linton C Freeman. “Centrality in social networks conceptual clarification.” In: *Social networks* 1.3 (1978), pp. 215–239.

[24] Linton C Freeman, Stephen P Borgatti, and Douglas R White. “Centrality in valued graphs: A measure of betweenness based on network flow.” In: *Social networks* 13.2 (1991), pp. 141–154.

[25] *GNU Portable Threads.* http://www.gnu.org/software/pth/.

[26] Roger Guimera *et al.* “The worldwide air transportation network: Anomalous centrality, community structure, and cities’ global roles.” In: *Proceedings of the National Academy of Sciences* 102.22 (2005), pp. 7794–7799.

[27] Bernhard Haslhofer, Roman Karl, and Erwin Filtz. “O Bitcoin Where Art Thou? Insight into Large-Scale Transaction Graphs.” In: ().

[28] Gábor Iván and Vince Grolmusz. “When the Web meets the cell: using personalized PageRank for analyzing protein interaction networks.” In: *Bioinformatics* 27.3 (2010), pp. 405–407.

[29] Leo Katz. “A new status index derived from sociometric analysis.” In: *Psychometrika* 18.1 (1953), pp. 39–43.

[30] SungJin Kim and SangHo Lee. “An Improved Computation of the PageRank Algorithm.” In: *Advances in Information Retrieval* 2291 (2002), pp. 73–85. ISSN: 16113349. DOI: 10.1007/3-540-45886-7_5. URL: http://dx.doi.org/10.1007/3-540-45886-7%7B%5C_%7D5.

[31] Maksim Kitsak *et al.* “Identification of influential spreaders in complex networks.” In: (2010). ISSN: 1745-2473. DOI: 10.1038/nphys1746. arXiv: 1001.5285. URL: http://arxiv.org/abs/1001.5285%7B%5C%%7D0Ahttp://dx.doi.org/10.1038/nphys1746.

[32] Jon M Kleinberg. “Authoritative sources in a hyperlinked environment.” In: *Journal of the ACM (JACM)* 46.5 (1999), pp. 604–632.

[33] Lothar Krempel. “Exploring the Dynamics of International Trade by Combining the.” In: December (2002), pp. 1–22.

[34] Amy N Langville and Carl D Meyer. *Google’s PageRank and beyond: The science of search engine rankings.* Princeton University Press, 2011.

[35] Qian Li *et al.* “Identifying influential spreaders by weighted LeaderRank.” In: *Physica A: Statistical Mechanics and its Applications* 404 (2014), pp. 47–55. ISSN: 03784371. DOI: 10.1016/j.physa.2014.02.041. arXiv: arXiv:1306.5042v1.

[36] Linyuan Lü *et al.* "Vital nodes identification in complex networks." In: *Physics Reports* 650 (2016), pp. 1–63. ISSN: 03701573. DOI: `10.1016/j.physrep.2016.06.007`. arXiv: `1607.01134`.

[37] Sarah Meiklejohn *et al.* "A fistful of Bitcoins: Characterizing payments among men with no names." In: *Proceedings of the Internet Measurement Conference - IMC '13* 6 (2013), pp. 127–140. ISSN: 15577317. DOI: `10.1145/2504730.2504747`. URL: `http://dl.acm.org/citation.cfm?id=2504730.2504747`.

[38] L Morten, J Robert, and E Walter. "The topology of interbank payment flows." In: (2006).

[39] Satoshi Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System." In: *Www.Bitcoin.Org* (2008), p. 9. ISSN: 09254560. DOI: `10.1007/s10838-008-9062-0`. arXiv: `43543534534v343453`. URL: `https://bitcoin.org/bitcoin.pdf`.

[40] *NEM Technical Reference.* `http://nem.io/NEM_techRef.pdf`. Accessed: 2017-08-01.

[41] Mark Newman. *Networks: an introduction.* Oxford university press, 2010.

[42] Mark EJ Newman. "A measure of betweenness centrality based on random walks." In: *Social networks* 27.1 (2005), pp. 39–54.

[43] Dá Niel Kondor *et al.* "Do the Rich Get Richer? An Empirical Analysis of the Bitcoin Transaction Network." In: *PLoS ONE* 9.2 (2014). DOI: `10.1371/journal.pone.0086197`.

[44] Athanasios N. Nikolakopoulos and John D. Garofalakis. "NCDawareRank." In: *Proceedings of the sixth ACM international conference on Web search and data mining - WSDM '13* February 2013 (2013), p. 143. DOI: `10.1145/2433396.2433415`. URL: `http://dl.acm.org/citation.cfm?doid=2433396.2433415`.

[45] Jae Dong Noh and Heiko Rieger. "Random walks on complex networks." In: *Physical review letters* 92.11 (2004), p. 118701.

[46] Micha Ober, Stefan Katzenbeisser, and Kay Hamacher. "Structure and Anonymity of the Bitcoin Transaction Graph." In: *Future Internet* 5.2 (2013), pp. 237–250. ISSN: 1999-5903. DOI: `10.3390/fi5020237`. URL: `http://www.mdpi.com/1999-5903/5/2/237/`.

[47] Lawrence Page *et al. The PageRank citation ranking: Bringing order to the web.* Tech. rep. Stanford InfoLab, 1999.

[48] Thai Pham and Steven Lee. "Anomaly detection in bitcoin network using unsupervised learning methods." In: *arXiv preprint arXiv:1611.03941* (2016).

[49] Marc Pröpper, Iman van Lelyveld, and Ronald Heijmans. "Towards a network description of interbank payment flows." In: (2008).

[50] Filippo Radicchi *et al.* "Diffusion of scientific credits and the ranking of scientists." In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 80.5 (2009), pp. 1–11. ISSN: 15393755. DOI: `10.1103/PhysRevE.80.056103`. arXiv: `0907.1050`.

[51] Dorit Ron and Adi Shamir. "Quantitative Analysis of the Full Bitcoin Transaction Graph." In: ().

[52] Gert Sabidussi. "The centrality index of a graph." In: *Psychometrika* 31.4 (1966), pp. 581–603.

[53] Computer Science and The Technion. "The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC E ect." In: (2001).

[54] M. Ángeles Serrano, Marián Boguñá, and Alessandro Vespignani. "Patterns of dominant flows in the world trade web." In: *Journal of Economic Interaction and Coordination* 2.2 (2007), pp. 111–124. ISSN: 1860711X. DOI: `10.1007/s11403-007-0026-y`. arXiv: `0704.1225`.

[55] Ma Angeles Serrano and Marián Boguñá. "Topology of the world trade web." In: *Physical review. E, Statistical, nonlinear, and soft matter physics* 68.1 Pt 2 (2003), p. 015101. ISSN: 1063-651X. DOI: `10.1103/PhysRevE.68.015101`. arXiv: `0301015 [cond-mat]`.

[56] Hiroaki Shiokawa, Yasuhiro Fujiwara, and Makoto Onizuka. "SCAN++: efficient algorithm for finding clusters, hubs and outliers on large-scale graphs." In: *Proceedings of the VLDB Endowment* 8.11 (2015), pp. 1178–1189.

[57] Florian Tschorsch and Björn Scheuermann. "Bitcoin and Beyond : A Technical Survey on Decentralized Digital Currencies." In: *IEEE COMMUNICATIONS SURVEYS & TUTORIALS* PP.99 (2015), pp. 1–1. ISSN: 1553-877X. DOI: `doi:10.1109/COMST.2016.2535718`.

[58] Johan Ugander *et al.* "The Anatomy of the Facebook Social Graph." In: *Arxiv preprint arXiv* abs/1111.4.November 2011 (2011), pp. 1–17. ISSN: 07308078. DOI: `10.1.1.31.1768`. arXiv: `1111.4503`. URL: `http://arxiv.org/abs/1111.4503v1`.

[59] Dylan Walker *et al.* "Ranking scientific publications using a model of network traffic." In: *Journal of Statistical Mechanics: Theory and Experiment* 2007.06 (2007), P06010.

[60] Gavin Wood. "Ethereum: a secure decentralised generalised transaction ledger." In: *Ethereum Project Yellow Paper* (2014), pp. 1–32.

[61] Xiaowei Xu *et al.* "Scan: a structural clustering algorithm for networks." In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM. 2007, pp. 824–833.