Indian Institute of Technology, Kanpur
Undergraduate Project Report

Enegiota: A scalable and feeless decentralized billing system on Tangle

Author: Shikhar Mahajan (150669)
Supervisor: Prof. Sandeep Shukla

13 February 2018

Abstract

One of most actively pursued current Blockchain-related use case has been the Internet of Things. The three prominent reasons being: M2M Payments, Securing Identity of Things and inferably automating the execution of processes. But the reality is Blockchain & IoT aren't ready to be in production yet mainly because of two limitations in current models: scalibility and transaction fees. With the number of connected IoT devices estimated to reach 50 billion in coming years, ability to perform micro-transactions between these devices is a major issue. In comes IOTA, mainly designed to tackle the problem of transactional settlement for IoT, is based on Directed Acyclic Graphs and relates scalability in direct proportion to number of transactions in the network without having any provision of incentivised mining and subsequently transaction fees. In this UGP, we study about the underlying details of Tangle which is the new distributed ledger architecture based on DAGs, transaction complexity involving tip selection algorithm from weighted random walks, the role of full nodes in IOTA, tackling the double spending problem in this mesh net and the newly introduced trinary hash functions and machinery which makes this technology quantum immune. We also employ the Tangle to build a web application named energiota to demonstrate billing in the associated cryptocurrency in a private testnet which involves around 25 different seeds. We also present the concept of attaching transactions to the mesh net and the role of Coordinator for our scheme.

1 Introduction

The ultimate goal of any distributed ledger is to ensure that the transacted data between parties is completely tamper-proof. But, IOTA can bring it in a unique way in the sense that it doesnt get inhibited by fees and scaling limitations like regular Blockchains. The new data structure of the distributed ledger associated with IOTA is called the Tangle which is based on a Directed Acyclic Graph. The Tangle actually has the same underlying principles of P2P network and decentralization as that of a Blockchain hence is usually

referred to as a Blockchain without the blocks and without the chains. The genesis transaction primarily created the whole quantity of cryptocurrency IOTAs that can ever be circulated in the network. All these transactions are called sites in the Tangle. The sites which are unconfirmed at given point of time are called tips. Each transaction or a site has an associated weight with it which is proportional to the amount of computation or work that the validating node has invested into it. The proof of work linked with IOTA is similar to Hashcash but has much less complexity. IOTA does not support smart contracts because it primarily meant for lightweight IoT devices which are very resource and infrastructure constrained but is interoperable with other platforms like Ethereum that support Smart Contracts.

2 Network

The data stored in IoT devices connected to the IOTA network are governed by CAP (Consistency, Availability, Partition Tolerant) theorem and hence this network also doesn't have a global state. This storage is maintained through snapshotting.
While setting up a full node in IOTA, the scheme demands the necessity of neighbour nodes that are connected through mutual tethering while involves sharing of protocol and IP address of the node. When an issuing node broadcasts a transaction or get to know about it from its neighbours, it forwards the information and send it to all of its neighbours which leads to eventual propagation of all the transactions through the network.
The data transfer is achieved through Masked Authenticated Messaging (MAM) which makes it possible to maintain an encrypted, authenticated and secure connection on the Tangle. This also ensures data integrity on the distributed ledger leading to only authorized parties able to access the data stream.

3 Tangle V/s Blockchain

1. **Centralization:** Blockchains tends to centralize in resources leading to aggregation of have mining pools, staking pools etc that make economic sense. Data records suggest that about 10 to 10 mining pools account for over 85% of the total mining processes out of which 8 are in China itself.
   Whereas in Tangle, no such pools make sense because there isn't any economic incentivisation of mining transactions.

2. **Discrimination of participants:** There are two parties in blockchain network-users and miners, both having diametrically opposite intentions. This can be somewhat understood by the laws of demand and supply. IOTA has eliminated this gap.

3. **Scalability:** In Blockchains, total number of users are in inverse proportion to speed of the nerwork due to a fixed size of a block whereas in Tangle the relation is opposite.

4. **Consensus:** In contrast to traditional Blockchains, consensus in Tangle is no longer decoupled but rather is an intrinsic part of the whole system, deducing to a completely decentralized and self-regulating P2P network.
   Also the consensus in IOTA is parallelized as opposed to sequential in case of

Blockchains which causes the network to scale exponentially in terms of number of transactions.

5. **Quantum resistance:** It is speculated that Quantum computers can cripple the Bitcoin network although they aren't in existence currently. IOTA is a future oriented project which uses exclusive quantum resistant cryptography involving Winternitz one-time signatures and a newly designed trinary hash function called Curl to provide a degree of quantum immunity.

6. **Micropayments:** In accordance to Blockchains, as the mining incentives per block are going down as time progresses, the transaction fees are constantly getting up resulting into inability to conduct micropayments. This is not the case with IOTA.

4 Transactions

Every transaction in the Tangle references to two past unconfirmed transactions. This reference is actually an attestation which directly attests those two tips and indirectly attests a subsection of Tangle linked to the tips. The attestation essentially requires listing of all that transactions in that subsection all the way back to the genesis transaction. Issuing a transaction in the Tangle is a 3 step process:

1. Signing: The transaction inputs are signed by the issuer's private keys

2. Tip Selection: Two random tips are selected using Markov Chain Monte Carlo Box (MCMC) which will be validated by the issued transaction. These transactions are referred to as branchTransaction and trunkTransaction.

3. Proof Of Work: For the transaction to accepted by the network, there has to be an investment of computation similar to Hashcash to approve the two transactions.

Let us assume the rate of incoming transactions to be λ. It is self explanatory that if the value of λ is too small, the network is a chain and if it's too large, the network becomes a tangle with each site referencing the genesis. Let the delay involved with the visibility (the time between preparation and propagation) of a transaction be 1 unit. This signifies that transaction after one unit time past the current time won't be available for attestation.

4.1 Tip Selection Algorithms:

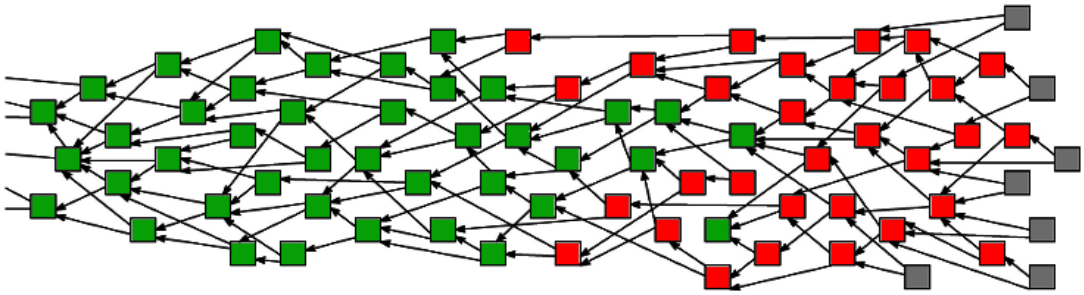
1. **Unweighted Random Walk**
   In this algorithm, we maintain a traverser starting from the genesis, and have it start traversing towards the direction of the tips. At each step it moves to one of the transactions which are directly referencing the current traverser. We select which site to jump into with equal probability, which is where the term unweighted makes sense from.

2. **Weighted Random Walk**
   We specify tips with approves older sites rather than recent ones as lazy tips. Hence our tip selection algorithm should be such that it avoids such lazy tips. To implement this in practice, there must be a provision of pre-defined incentives

against any such behaviour of laziness which would in turn result into unlikeliness of lazy tips getting attested by anyone. In comes the concept of weights. The cumulative weight associated with any transaction is the count of number of direct + indirect approvers of that transaction +1. Hence the algorihm suggests that we are more likely to choose a walk involving a heavy transaction than a light one. Now this leads to a new parameter α which determines the importance of a transaction's cumulative weight. If the value of α is set to zero, we actually follow the unweighted walk and on the other side if the value of α is set very high, we get a super weighted walk. So deciding upon fitting value of α is actually a research topic in the IOTA community. Now we know that this method of deciding probability of each step is known as Markov Chain Monte Carlo (MCMC) approach. To fiddle out with these parameters, one can visit `https://public-qnbiiqwyqj.now.sh/` [1]

5 Consensus [2]

The transaction finality in IOTA is an elegant and a simple characteristic of the system. It is absolutely self-regulating ensuring both security as well as scalability.



[3]

The green squares in the above picture denotes confirmed sites i.e. transactions on which consensus is achieved and hence a degree of finality guarantees. The red squares are the sites whose confirmation on full acceptance is still uncertain and the grey squares are the tips i.e unconfirmed sites.

Looking at the picture, the main observable difference between the red and green blocks is the degree of acceptance i.e. the green blocks are indirectly approved by all the grey blocks. This means that for a transaction to get confirmed, a direct path from every tip should lead to it. Now as such, there isn't any straightforward way of determining the confirmation degree of a site. What we follow is executing MCMC tip selection algorithm N times out of which M is the number of times the algorithm lands on a tip having a direct path to that site. Hence the probability with which that site can be accepted comes out to be M/N.

Now as a participant of the Tangle network, one has a complete freedom to decide on what probability factor will they start accepting transactions. For a high value transaction, this threshold factor may be increased to 0.99 or even 1.

6 The Double Spending Problem

Lets understand the IOTA mesh topology as it's a crucial first step and differs majorly from all other Blockchain protocols. Each full node in IOTA sees only a tiny part of the Tangle and that through via information exchange from its healthy neighbours. No particular node has access to all IPs of all the nodes. These two dominant factors of mutual tethering and IoT connectivity makes IOTA a mesh net. This concept of mesh net has strengthened the network resiliency against the network hashing power attack.

Note: In Blockchains, there's a well known 51% attack on the network. A deeper research shows that only 34% of network hashing power may just carry out the attack. [4]

Now, lets dig deeper into the 34% attack possibility in IOTA. As we know Blockchains aren't mesh nets, the 34% attack in Blockchains is just the result of enough hash power to successfully attack the network. Hence the only variable in Blockchain governing the possibility of attack is the fraction of total hashing rate. However in IOTA, there are three different variables governing the same:

1. Percentage of network hash rate (X%)
2. Omnipresence: The ability to see the entire network topology at once i.e. having an overview of the network to determine the speed of getting new information of the Tangle.
3. Omnipotence: The percentage of network that the full node is neighbouring with(Y%)

For example, say $X = 25\%$ and $Y = 15\%$ for an attacker. In this case he will only be able to bring down a small number of edge nodes that he is connected to. Its actually the combination of X and Y that determines the percentage of edge nodes taken down by the attacker and hence, the effectiveness of the attack. X can be as high as 99% for some node but without sufficient Y, its not as effective. This is because when the edge nodes corresponding to these edge nodes are overwhelmed by the compromised transactions, they'll just blacklist the attacker and restart their own node to become functional again.

7 The Coordinator

IOTA is currently not as spread out and is considered to be in 'transition period' towards eventual large scale deployment and standardization. The Coordinator or 'Coo' is essentially the training wheels for the network until the Tangle can be left to evolve unassisted. The Coo controls the transaction status of all the transactions presently. In certain periods of time Coordinator issues a new transaction called 'milestone' (similar looking to a normal transaction but issued by Coo) which indicates all the full nodes to change tangle's states. When a transaction is directly or indirectly referenced to by a milestone, its status will be changed to a confirmed transaction. Hence when you want to know if any particular transaction is confirmed or not, find the newest milestone released by the Coo check if it indirectly attests your transaction.

Why Coo is in existence?
The reason behind the existence of Milestones is that if you pick up any random transaction, there's a possibility that that the node in your neighbour is malicious and is trying to trick you into verifying its transactions. Hence the only role that Coo plays is to protect the network against the hash rate attack in its infancy stage. In fact, its quite easy to change the Coo logic with a Random Walk Monte Carlo logic in its own private testnet.

What if Coo is malicious?
Each node looks at the information it gets from its neighbours and only propagates other neighbours about transactions that are valid. The Coo is also no exception, if it starts issuing bad Milestones, full operational nodes will just reject them. Hence decentralization still pertains.

8 Downsides:

1. All its internal trinary notation have to be encapsulated in binary, leading into significant overhead in storage and computation.

2. Derivably, there is also a small but compulsory need of additional ASIC of IOTA in each IoT device to communicate with the network.

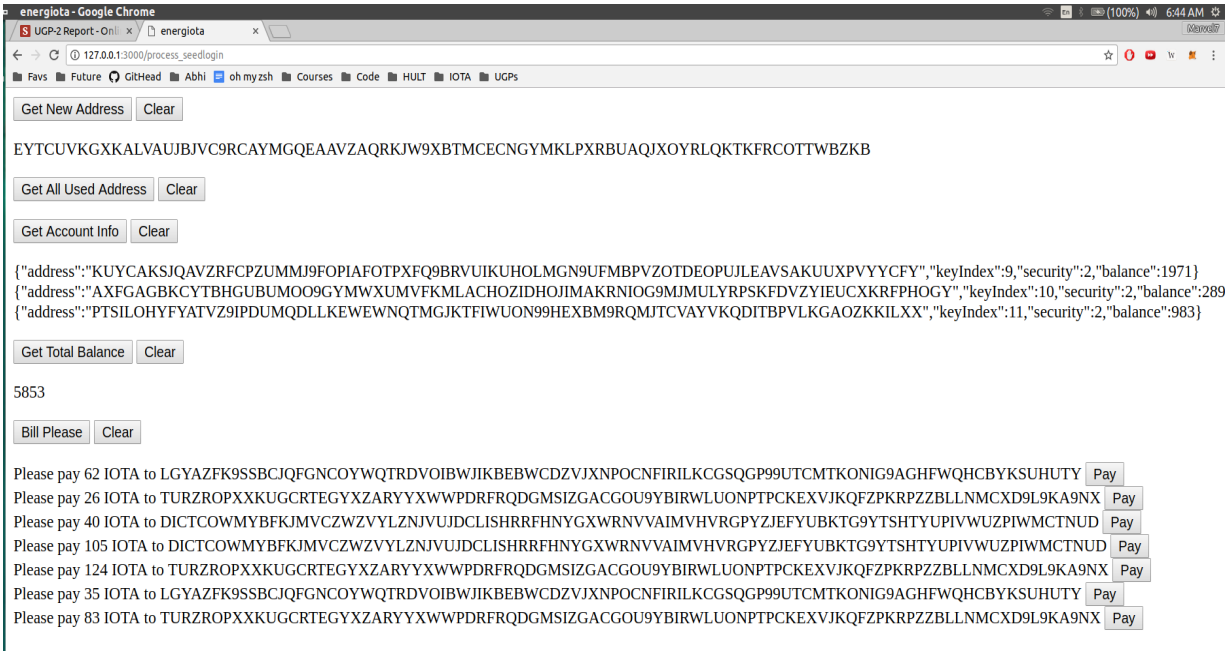
3. They have reinvented their own hash function called the Curl which has been scrutinized to many hash-cracking attacks. It also violates the Rule 1 of Cryptography: "Don't roll your own crypto".

9 Energiota: A demonstrative billing system

Energiota is a web application leveraging the Tangle and the IOTA network in a private testnet [5]. The transactions in this testnet are governed by Milestones generated by the Coo as and when required. The initial snapshot consists of distribution of all the 2.8 Peta Iota (2,779,530,283,277,761) into about 25 different seeds initially having 3-5 holding addresses each. Out of these, 3 addresses are chosen to that of electricity providers to whom the generated bill has to be paid. The bill is currently generated randomly between 25 to 125 iota but linking it to a real electricity meter is one of the main prospects of the project.

Operations

Following is the image of the current UI provided by the billing Dapp:



The description of each function is as follows:

1. **getNodeInf:**
   (a) Returns the getNodeInfo method from the iota.js api
   (b) Outputs the name and version of the IOTA software, available cores for jre in the machine, amount of free memory for JVM, information about the Milestones etc
2. **getNewAddr:**
   (a) Returns the getNewAddress method from the iota.js api
   (b) Outputs the next unused and unattached address linked with the login seed
3. **getUsed:**
   (a) Returns the 'addresses' field of getAccountData method from the iota.js api
   (b) Outputs the array of all used addresses which have transactions associated with them from the login seed
4. **getAccountInfo:**
   (a) Returns the 'inputs' field of getAccountData method from the iota.js api
   (b) Outputs the array of all inputs available for the login seed
   (c) Follows the getInputs format of 'address', 'balance', 'security' and 'keyIndex'
5. **getBal:**
   (a) Returns the getBalances method from the iota.js api
   (b) Outputs the latest confirmed balance associated with the login seed

6. **getBill:**
   (a) Generates a single bill to be paid to one of the billers amounting randomly between 25 and 125
   (b) Also embeds the script to pay the bill in the action of 'Pay' button
7. **payBill:**
   (a) Calls the sendTransfer method from the iota.js api
   (b) Extracts the biller and amount from the bill, sets the MWM to be 14 and issues a proper transaction

References

[1] Alon Gal. IOTA Tangle Visualisation. https://public-qnbiiqwyqj.now.sh/.

[2] Alon Gal. A primer on IOTA. https://blog.iota.org/a-primer-on-iota-with-presentation-e0a6eb2cc621.

[3] Scott J. IOTA Masterclass. https://forum.iota.org/t/iota-consensus-masterclass/1193.

[4] Eclipse Attacks. https://eprint.iacr.org/2015/263.pdf.

[5] Private IOTA Testnet. https://github.com/schierlm/private-iota-testnet.