NOTES OF GUIDANCE ON THE COMPLETION OF THE NOTICE OF SUBMISSION FORM

You should complete both sides of the form in typescript or black ink and use block capitals. The completed form must be handed to your Head/Dean of School (or his/her nominee) with two copies of your dissertation and (where appropriate) the examination fee.

Each copy of the dissertation shall contain:

1. a summary not exceeding three hundred words;
2. a statement signed by the candidate showing the extent to which the work submitted is the result of the candidate's own investigation, and an explicit acknowledgement (with references) of any other sources used;
3. a full bibliography;
4. a signed declaration to certify that the work submitted has not been accepted in substance for any degree or award, and is not being submitted concurrently in candidature for any degree or other award;
5. a signed statement regarding availability of the Dissertation;
6. a completed 'Notice of Submission' form.

The declaration and statements referred to in 2 to 5 above should be incorporated at the beginning of the dissertation, as shown in Appendix 1.

Where this is in accordance with the School's policy on the submission of Taught Master's Dissertations:

- candidates may submit their work for examination in temporary binding;
- candidates may be instructed by the Schools to submit one of the two required copies in an approved electronic format.

APPENDIX 1: Specimen Layout for Declaration/Statements page to be included in Taught Master's Degree Dissertations

CANDIDATE'S ID NUMBER	c1130984
CANDIDATE'S SURNAME	**Please circle appropriate value** **[✓ Mr] / Miss / Ms / Mrs / Rev / Dr / Other please specify** ...Gardner....
CANDIDATE'S FULL FORENAMES	Richard Samuel

DECLARATION

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed ......R Gardner...... (candidate) Date ...06/09/2018...

STATEMENT 1

This dissertation is being submitted in partial fulfillment of the requirements for the degree of ......MSc......(insert MA, MSc, MBA, MScD, LLM etc, as appropriate)

Signed ......R Gardner...... (candidate) Date ...06/09/2018...

STATEMENT 2

This dissertation is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A Bibliography is appended.

Signed ......R Gardner...... (candidate) Date 06/09/2018...

STATEMENT 3 – TO BE COMPLETED WHERE THE SECOND COPY OF THE DISSERTATION IS SUBMITTED IN AN APPROVED ELECTRONIC FORMAT

I confirm that the electronic copy is identical to the bound copy of the dissertation

Signed ......R Gardner...... (candidate) Date ...06/09/2018...

STATEMENT 4

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ...R Gardner...... (candidate) Date ...06/09/2018...

STATEMENT 5 - BAR ON ACCESS APPROVED

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for inter-library loans **after expiry of a bar on access approved by the Graduate Development Committee.**

Signed ...R Gardner...... (candidate) Date ...06/09/2018...

NODIADAU CYFARWYDDYD YNGHYLCH LLENWI FFURFLEN HYSBYSIAD O GYFLWYNO

Dylech lenwi dwy ochr y ffurflen mewn teip neu inc du a defnyddio priflythrennau. Ar ôl i chi lenwi'r ffurflen, rhaid ei chyflwyno i Bennaeth/Deon eich Ysgol (neu ei (h)enwebai) gyda dau gopi o'ch traethawd hir ac (os yw'n briodol) y ffi arholi.

Rhaid i bob copi o'r traethawd hir gynnwys:

1. crynodeb nad yw'n fwy na thri chant o eiriau;
2. gosodiad sydd wedi'i lofnodi gan yr ymgeisydd ac sy'n dangos i ba raddau y mae'r gwaith a gyflwynir yn ffrwyth ymchwiliad yr ymgeisydd ei hun, a chydnabyddiaeth benodol (gyda chyfeiriadau) o unrhyw ffynonellau eraill sydd wedi'u defnyddio;
3. llyfryddiaeth lawn;
4. datganiad sydd wedi'i lofnodi i dystio nad yw sylwedd y gwaith a gyflwynir wedi'i dderbyn ar gyfer unrhyw radd neu wobr arall, ac na chyflwynir ef yr un pryd mewn ymgeisyddiaeth ar gyfer unrhyw radd neu wobr arall;
5. gosodiad, sydd wedi'i lofnodi, ynghylch argaeledd y Traethawd Hir;
6. ffurflen 'Hysbysiad o Gyflwyno', a honno wedi'i llenwi.

Dylai'r datganiad a'r gosodiadau y cyfeiriwyd atynt yn 2-5 uchod gael eu hymgorffori ar ddechrau'r traethawd hir, fel y dangosir yn Atodiad 1.

Os yw hynny'n unol â pholisi'r Ysgol ar gyflwyno Traethodau Hir am Radd Athro a Addysgir:

- caiff ymgeiswyr gyflwyno'u gwaith i'w arholi mewn rhwymiad dros dro;
- gall ymgeiswyr gael eu cyfarwyddo gan yr Ysgolion i gyflwyno un o'r ddau gopi gofynnol mewn fformat electronig a gymeradwywyd.

ATODIAD 1: Patrwm Enghreifftiol ar gyfer tudalen y Datganiad/Gosodiadau sydd i'w chynnwys mewn Traethodau Hir am Radd Athro a Addysgir

RHIF ADNABOD YR YMGEISYDD	
CYFENW'R YMGEISYDD	**Rhowch gylch o gwmpas y teitl priodol** **Mr / Miss / Ms / Mrs / Y Parch / Dr / Arall – enwch ef …………………..**
ENWAU BLAEN LLAWN YR YMGEISYDD	

DATGANIAD

Nid yw sylwedd y gwaith hwn wedi'i dderbyn o'r blaen ar gyfer unrhyw radd ac ni chyflwynir mohono mewn ymgeisyddiaeth ar gyfer unrhyw radd ar hyn o bryd.

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ………………………..

GOSODIAD 1

Cyflwynir y traethawd hir hwn gan rannol gyflawni gofynion gradd ………………………..(rhowch MA, MSc, MBA, MScD, LLM ac ati, fel y bo'n briodol)

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ………………………..

GOSODIAD 2

Ffrwyth fy ngwaith/ymchwilio annibynnol fy hun yw'r traethawd hir hwn, ac eithrio lle y nodir fel arall.
Cydnabyddir ffynonellau eraill mewn troednodiadau sy'n rhoi cyfeiriadau manwl. Atodir Llyfryddiaeth.

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ………………………..

GOSODIAD 3 – I'W LENWI OS CYFLWYNIR AIL GOPI O'R TRAETHAWD HIR MEWN FFORMAT ELECTRONIG A GYMERADWYWYD

Yr wyf yn cadarnhau bod y copi electronig yn union yr un fath â'r copi o'r traethawd hir sydd wedi'i rwymo.

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ………………………..

GOSODIAD 4

Rhoddaf ganiatâd drwy hyn i'm traethawd hir, os derbynnir ef, fod ar gael i'w lungopïo ac i'w fenthyca rhwng llyfrgelloedd, ac i'r teitl a'r crynodeb fod ar gael i sefydliadau allanol.

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ……………………

GOSODIAD 5 - CYMERADWYWYD Y GWAHARDDIAD AR FYNEDIAD

Rhoddaf ganiatâd drwy hyn i'm traethawd hir, os derbynnir ef, fod ar gael i'w lungopïo ac i'w fenthyca rhwng llyfrgelloedd **ar ôl i'r gwaharddiad ar fynediad iddo, a gymeradwywyd gan y Pwyllgor Datblygu Graddedigion, ddod i ben.**

Llofnodwyd ………………………………………………………….. (ymgeisydd) Dyddiad ……………………

Understanding the performance of the tangle, a solution to the blockchain problem

Richard Gardner - MSc Computing

September 6, 2018

School of Computer Science and Informatics, Cardiff University

Supervised by Dr Philipp Reinecke

Abstract

Decentralised payment systems based on blockchain - such as Bitcoin face the inevitable problem of scalability, limited by their block size and interval. A new cryptocurrency known as IOTA uses a novel data structure to distribute its ledger - the tangle. A tangle is a directed acyclic graph, where transactions are added one at a time and seemingly unaffected by the same problems as Bitcoin. This project assesses how well and to what extent a data structure such as the tangle can address the problems of Bitcoin and the blockchain, and investigates if there are any unforeseen problems that arise from using the tangle. In this project we simulate the tangle at different throughputs comparable to those of Bitcoin, and with high and low latency subsets of the network to answer these very questions.

Contents

List of Figures

1 Introduction

In 2008, a pseudo anonymous individual known as Satoshi Nakamoto published the Bitcoin whitepaper[15] describing a cashless peer to peer payment system avoiding trusted third parties. Since then Bitcoin has become the most well known and highest valued crypto asset in the world. While popular, Bitcoin has its problems if we are to consider it as a payment system - if we see a payment system as something we might use to purchase goods online or buy our groceries at the supermarket. Bitcoin, like all blockchain based cryptocurrencies faces an inherent throughput bottleneck as a consequence of how often a block is added, and how many transactions can be fit into a block. Users must pay a fee to the ”miners” who compete to solve a puzzle to find a block for their transaction to be accepted - and at times when many people are issuing transactions this fee can be more than the value of their payment.

These problems are well known and there are now many other cryptocurrencies in circulation, some created with these problems in mind, others not. Here we introduce the tangle - or IOTA[17] as the token is called, a tangle is a directed acyclic graph of transactions or, rather than a block encapsulating many transactions as in a blockchain - a ”block” in a tangle is a singular transaction. To become part of the tangle a transaction must approve or validate two others - which mitigates paying transaction fees to miners. As such, a tangle based currency such as IOTA should solve the problems that plague Bitcoin, but to what extent and, given the randomness of a data structure aptly named tangle, are there any problems that are as inherent to a tangle as throughput is to a blockchain?

Broadly, this project aims to examine in detail how Blockchain crpyto tokens as payment systems cannot be sustained, nor compete with classical payment services such as PayPal. But by using the directed acyclic graph of transactions - or ”tangle” as the primary data structure in such a payment system - instead of a blockchain, we can potentially circumvent these problems. Using a simulation model constructed from the discrete event simulation software oment++, we can test different parameters such as throughput in the network and latency of nodes within it, to validate the claim that a tangle solves blockchain’s problems, and gain understanding of any comparably inherent bottlenecks as we find in a blockchain.

Firstly we will examine the abstract data structures: the directed acyclic graph, the tangle and the blockchain. From there we will look at the more concrete implementations: Bitcoin and IOTA, we can then analyse the problems that Bitcoin has and how IOTA could solve them. We will then have a solid understanding of both data structures and be in a good position to run experiments using the simulation model, the experiments run use overall network throughput and a subset of the network with varying latencies as their parameters, we then look to analyse the results to hopefully answer the questions posed above.

2 Aims and Objectives

2.1 Examine the blockchain problem

Firstly I'd like to take a step back and examine what exactly the problem Satoshi Nakamoto [15] was trying to solve with his Bitcoin protocol, and how the steps taken to solve this problem lead us to the present. We can then examine where and how it falls short, and why a structure such as a tangle seems to be a successor for a distributed public ledger as a payment system. This section hopes to identify hard problems that must be overcome for a cryptocurrency to be a viable payment system and other, soft problems that should be overcome as a means to succeed based on the claims made by Nakamoto - or rather, the use case purported in the Bitcoin whitepaper.

2.2 Assess the tangle as a solution and identify potential problem areas

Using a simulation model built using omnet++ we can see how the tangle responds in different scenarios in direct response to the problems examined above. By simulating at different throughputs and varying distributions of latencies, we can extract useful information to better understand the tangle and evaluate if such a data structure is a viable alternative to the blockchain. Using the results we can identify any problems that might not be an issue in a blockchain based cryptocurrency but certainly could be in a tangle. We should expect the latency of nodes to play a more significant role in the success in a tangle based network than that of a blockchain. Is this the case? If so, to what what extent? And furthermore - are there any other factors hitherto not considered that could play a role? The experiments completed using the model will attempt to answer these questions.

3 Background

In this section, we look at the Bitcoin and IOTA protocols and their implementations in detail, as well as others that provide differing solutions - many of which are not without merit. However, IOTA has some advantages over other implementations - which will be explained in section 4, as will the reasoning behind selecting IOTA as *the* solution explored in this project. It should be noted that many concepts discussed, such as: cumulative weight of a transaction and Markov chain tip selection are introduced in two papers [12] and [11] produced by the IOTA foundation research team on simulating the Tangle. This project aims to build on and validate the results seen therein, and offer an extension by looking at the relative latencies of a node and how that affects transaction approval.

3.1 The abstract data structures

Any distributed ledger implementation faces a choice of which data structure to distribute, in case of Bitcoin a blockchain was chosen - which comes with advantages disadvantages as this project will discuss.

3.1.1 A directed acyclic graph

A directed acyclic graph is a graph such that nodes(or vertices) are connected via edges in such a way, as to never allow a path that can lead back to a node that has already been visited. Or rather that the edges have a direction - forward. On an abstract level all distributed ledgers (or anything that has transactional properties) are a directed acyclic graph, in that the direction is time and a transaction cannot be placed in this graph before it occurred.

3.1.2 Blockchain and the Tangle

Given the above, we can think of blockchain as a means of representing the directed acyclic graph of transaction issued. That a finite amount of transaction in the DAG can be represented in a single block, and blocks containing these transactions can be appended as needed. In the same way that blockchain was chosen as the data structure of Bitcoin, the Tangle was chosen to represent transactions in IOTA. A tangle is simply a one for one representation of the abstract DAG a transactional system represents, so rather than blocks with a variable number of transactions, a ”block” is a single transaction.

3.2 Bitcoin and IOTA

Here we examine how our chosen crypto implementations work on an more abstract level, with Bitcoin described before IOTA. Bitcoin was among the first cryptocurrency to be developed and certainly the first to achieve widespread adoption. It's therefore imperative that we frame any discussion of other cryptocurrencies in this context, as many if not all other cryptocurrencies were designed as a direct response to Bitcoin's shortcomings

3.2.1 The Bitcoin protocol

In the whitepaper, Bitcoin is described as a way to enable purely peer to peer payments, without relying on trusted third parties[15]. Or in other words, any centralised entity having the final say over approving or denying transactions - such as a bank. In the Bitcoin network, transactions are broadcast to nodes which then compete to solve a complex puzzle, which on completion allows the solver to append a new block to the chain. This new block is broadcast to all other nodes who verify the authenticity of the completed puzzle, they show their acceptance of the block by starting work on the next block.

This complex puzzle is called proof of work, and it allows participants in the network to be confident that the ledger of transactions are valid (not double spends) as long as at least

50% of the computing power in the network is contributing honestly. Nodes contributing towards the proof of work are *miners*, as such in the bitcoin protocol there are two distinct types of participants: *miners* and *transactors*. Whichever node solves the puzzle first, is rewarded with a not insignificant amount of bitcoins which provides an incentive for miners to participate. Transactors must also pay a transaction fee(included in the transaction itself) which goes to whichever miner finishes the block said transaction is included in. Section 4 examines how miners and transaction fees affects the claim of decentralisation.

The key point here is that Bitcoin and blockchains in general are limited by their difficulty, - how long it takes for a block to be mined, and block size, i.e. how many transactions can be fit into any given block in the chain. This is a key issue which will be examined more closely in this project, for now it is important to be able to understand why other implementations such as IOTA are designed as they are, as we discuss them.

3.2.2 IOTA - The Tangle

The IOTA token [17] is similar to many cryptocurrencies in that consensus is achieved through proof of work[1] - like Bitcoin, this is where the similarities end in the actual implementation. Instead of a chain of blocks containing an arbitrary number of transactions, IOTA transactions are stored in a *Tangle*, which is a Directed Acyclic Graph. Transactions are atomic in the tangle - meaning one "block" is one transaction, and each transaction is connected via edges, where an edge means a transaction has approved another. When a transactor wishes to issue a transaction the issuer must approve two others, where an unapproved transaction is known as a *tip*, the issuer uses a tip selection algorithm to choose which tips to approve and computes the proof of work ensuring their validity.

With this approach, we can see that in the tangle there is only one participant - the transactor. With no miners necessary because every transaction issued acts as a "miner" for two others, which does away with the need for transaction fees. The most important aspect of the tangle and IOTA, is the tip selection method used by a transactor to select attachment sites i.e. which tips to approve.

There will be two types of tip selection methods discussed in this project: Uniform random tip selection(URTS) and weighted a random walk(Walk). URTS chooses a tip completely at random, placing no higher value in choice from one tip to the next. While the walk selection method traverses the tangle from transaction to transaction from a determined distance back, an attachment site is selected when this walk reaches a tip. The walk selection method used in the simulations uses one walker, and finds the highest weight site available, it then chooses to move to the highest weight site or pick from the rest at random - what determines this choice is the alpha value which is between 0 and 1, the walks in the simulations only use an alpha value of 0.1. While URTS has potentially higher performance it does not discourage lazy or malicious entities from approving either the same transaction repeatedly, or forming a malicious sub-tangle that could enable a double spend. The purpose of the Walk selection method is to provide a slight bias towards higher weighted transactions i.e.

[1] Many other tokens such as Ethereum [4], Nano [13] and Dash [7] use proof of stake - to name but a few, as it offers advantages over proof of work. Section 4 discusses this.

those that have been approved or indirectly approved by more transactions, this creates a structure that is much harder to compromise and is essential to a tangle like cryptocurrency.

3.3 Other implementations

IOTA is not the only DAG cryptocurrency, nor is it the only cryptocurrency to try to address the problems of blockchain. This section gives an overview of other implementations doing exactly that. Section 4 provides more detail as to why I selected IOTA as this project's focus.

3.3.1 Nano

Nano uses a Block lattice structure [13], where every account owner has their own blockchain which only they can append to. Each user's chain records send and receive transactions and the network uses proof of stake to resolve any conflicts, by having other nodes vote as to which block is the valid one, with any given node's voting power proportional to the amount of tokens their private chain holds - users can also assign others to represent their voting power, as for most it is impractical to run a node at all times.

The white paper refers to the implementation as a DAG, however this is only in the sense that any ledger representing transactions is, the data structure that holds this DAG is as the paper says - a block lattice.

3.3.2 Bitcoin Lightning network

The Bitcoin Lightning network [16], involves adding peer to peer payment channels on top of the existing Bitcoin blockchain. Not a novel cryptocurrency but rather a way for participants to avoid needless transactions. Channels are created with transactions that are signed by the involved parties - but not broadcast to the network. The channels are only broadcast and therefore added to the network after certain time has passed. This enables many transactions to be condensed into few, with security that if trust between parties in a channel is broken, any coins committed to the channel will be returned simply by broadcasting the transaction prematurely.

4 The problem(s)

Bitcoin succeeds in its claim as a cashless peer to peer payment system, but how successfully? It has limitations both of scalability and decentralisation, these are inherent to most blockchain implementations - this section discusses and attempts to contextualise these problems from a use case perspective.

4.1 Where Bitcoin succeeds

As stated, Bitcoin succeeds in its endeavour to become a cashless peer to peer payment system with no trusted third party, and is currently the most well known, widely adopted cryptocurrency. If you have a quantity of Bitcoin, you are able to send some or all of that to an address of your choosing simply by broadcasting a transaction to the wider network. As long as you pay a sufficient transaction fee, your transaction is extremely likely to be included in a subsequent block - and not left behind unapproved.

In the sense that it does work as intended means it is successful, but the perception of that success is somewhat diminished if we compare it to classical payment systems of the world today - the likes of Visa or PayPal. As a proof of concept for a cryptocurrency Bitcoin succeeds, and undoubtedly any implementation that does so in the future should cite Bitcoin as an enabling factor [2]. But it does not compare to the throughput, cost effectiveness or environmental impact of centralised systems.

4.2 Where Bitcoin falls short

Bitcoin's problems largely stem from the mechanism that ensures its security. In Bitcoin miners are motivated to contribute towards the security of the system by computing proof of work with the chance of receiving bitcoins as a reward if they "find" the next block, they are also rewarded with the fees paid by the transactions included in the block. Here we examine the issue in more detail.

4.2.1 Scalability and transaction fees

The issue of scalability is a simple one, in that fundamentally the transactions per second of any blockchain implementation will be always bottlenecked by how many transactions can be fit in a block, and how often those blocks are appended to the chain. Currently, a Bitcoin block has a max size 1MB, and a transaction is on average 495B, - meaning that 2020 transactions can be included in each. With the adoption of the SegWit protocol[14] - a soft fork[3], the block size remains unchanged but the transactions that can be fit into the block is increased. It's not clear how much of the network adopt segwit, but a non insignificant proportion must, as the peak average block size in December 2017 was 2704 transactions per block [2] - higher than the hard limit of 1MB per block if all transactions were non segwit. If all transactions in a new block were segwit transactions, then the theoretical block size limit would be 4MB, combined with the standard block interval of 10 minutes, we get a throughput of 13.47 Tx/s. While a block containing only non segwit transactions gives us a throughput of 3.37 Tx/s.

The block interval is set at 10 minutes - or rather, the difficulty of computing the proof of work necessary to find a block, is such that given the total hashing power of the network

[2] Most if not all crypto tokens or currencies do in fact reference the bitcoin whitepaper

[3] A soft fork being a change in protocol that is optional, as transactions not adhering are backwards compatible

- a miner will find a block - on average every ten minutes. This difficulty is increased every 2048 blocks created - it's been shown [3] that the ten minute per block is an average, while as long as the global hash rate increases there will be a blocks found quicker than 10 minutes the closer the protocol to a difficulty adjustment

These transactions per second as above, are insignificant in comparison to a centralised payment network such as PayPal, who during the second quarter of 2018 processed 2,327,000,000 transactions [18] which equates to an average of 294.93 Tx/s - bearing in mind that this is just one example of the many other centralised systems. If the goal of Bitcoin - or even any cryptocurrency is to achieve widespread adoption, then clearly there there is a problem here. This paper [8] showed that the theoretical max throughput of the any proof of work powered blockchain based cryptocurrency - without significantly compromising security - is approximately 60 Tx/s, when you keep a block size of 1MB and decrease the block interval to 60 seconds. A cleverly engineered solution such as segwit, or even a theoretical implementation does not come close to increasing Bitcoin's throughput to rival that of an established payment system.

Network throughput aside we have the issue of transaction fees, a transaction fee must be paid to ensure a transaction will be accepted into a block - with no regard for the monetary transfer value of the transaction. Meaning if you wanted to send a small amount - to buy a coffee for example, depending on network congestion (i.e. how many transactions there are waiting to be approved) you could end up paying a fee larger than the actual transfer value. In fact the average transaction fee for Bitcoin peaked in January 2018 at USD 55.16 [1], meaning any transaction at that time regardless of value could be subject to such a fee. If for example, there were many individual entities issuing many small transactions such as normal people buying coffee then the network would quickly become congested causing fees to increase. This is actually the problem the Bitcoin lightning network [16] hopes to solve, by collating many small transactions into fewer, larger ones to save on block space.

4.2.2 Pseudo decentralisation

For Bitcoin we can see that there are two distinct types of participants in the network: those who issue transactions and those who approve them. The two are not mutually exclusive but given the computational power needed to have a chance at mining effectively, there are many more issuers than approvers. Bitcoin's security relies on at least 50%[4] of the total hashing power of the mining network being honest, assuming that the resources needed to garner such a proportion is not feasible - and given the rewards associated with successfully mining, we can can be reasonably sure of this security.

The problems lies in the self interestedness of both involved parties, there can be no doubt that those contributing to the security of the network are doing so because of the rewards associated, and those issuing transactions are doing so because they wish to send funds from address A, to address B. The issuers self interestedness comes from the motivation for their transaction to be approved, now in a scenario such as December 2017 to January 2018 where

[4] It has been shown that given certain conditions and strategies this proportion can decrease to 30% [8]

the Bitcoin network was congested, fees were at an all time high - if an issuer wants their payment to be approved they must include a fee. Miners are under no obligation to act altruistically, i.e. approve transactions in a chronological order. Were they inclined to do so, miners could only include transactions that pay the highest fees. Given that the rewards for finding a block halve every 210,000 blocks (And will eventually stop altogether), there will come a point where mining profitably involves only accepting transactions with the highest fees - ignoring, or putting a lower priority on those that pay a lower fee. We see that at the moment this is not necessarily the case as [10] showed, with many miners seemingly not adopting this strategy - or mining as efficiently as possible, but this will undoubtedly change when profiting solely off of the block reward only is no longer possible.

We come back to Nakamoto's claim of a payment system that "avoids trusted third parties" [15], while the mining pool is not considered a single centralised entity, given the circumstances above it is not illogical to assume that only those that adopt the most efficient mining strategies will continue to operate (only those that turn a profit). Given that there is an optimal strategy, it makes sense to assume that all those who can adopt this strategy will - when the alternative is not turning a profit. If every entity adopts the same strategy, we can then start to see them as a single centralised entity. In which case Nakamoto's claim if not wholly false, is certainly weaker.

4.2.3 Environmental impact of mining

In a paper recently published [6] it was calculated that currently Bitcoin consumes at least 2.55 GW of power a year - comparable to Ireland which consumes 3.1 GW, and could reach 7.67 GW by the end of 2018. As the popularity of mining for blocks increases, the Bitcoin protocol is designed in such a way that the same block interval is maintained by changing the number of leading zero bits in the nonce to be found. Meaning that the power consumption per transaction increases while the throughput remains the same. While this mechanism all but ensures the security of the system, it is inherently wasteful at a time when there is ever more pressure on the human race to be the opposite.

One method to counter the wasteful power consumption of proof of work mining is to use another mechanism known as proof of stake - used by tokens such as Ethereum [4][5]. In a proof of stake secured cryptocurrency, nodes that wish to contribute to the security of the network can volunteer their assets in what can be described as collateral or a deposit. The network then decides which node gets to mine the next block deterministically, in that the higher the percentage of the total assets in the network owned by a node, the more likely that the node will be chosen to mine the block. This avoids the need for nodes to complete needless hashes to find a nonce, the node still checks the validity of the transactions in that they are not double spends, but only the work needed is actually computed.

While a proof of stake style security mechanism solves the reckless energy consumption of a blockchain cryptocurrency, we then find ourselves yet again in a weaker position in a regards to decentralisation. As those with more assets have more influence on the security

[5] Ethereum started out as a proof of work token, yet completed a hard fork in order to switch to the less wasteful proof of stake

of the network, if a node were to accumulate more than 50% of the total assets available they could compromise the security of such a system. While the costs involved of such an attack make it highly improbable[6], the mechanism again relies on the self interestedness of the stake holders - that those with a higher stake in the system benefit most from its continued security. When a stake holder has a larger influence on the network than a smaller stakeholder, this weakens any claim of decentralisation.

4.2.4 Summary

To summarise, Bitcoin as a whole does not scale anywhere close to an established payment system such as paypal - being limited by its block size and interval, while its method of ensuring security consumes nearly as much power as a small country and finally, at times when the network is congested discourages small payments given the high fees required. It's claim of decentralisation is also made weaker by the miners autonomy in selecting transactions to include. It should also be noted that the problems explored above are not exclusive to Bitcoin, any cryptocurrency that uses a blockchain as its distributed ledger falls victim to the scalability problem whether or not it uses proof of work or stake.

5 How IOTA addresses blockchain's problems

A DAG cryptocurrency such as IOTA could solve or negate all the problems of a blockchain based implementation. Here we take an in depth look at how the problems outlined above are solved when using a DAG instead of a blockchain.

5.1 Scalability and transaction fees

Unlike Bitcoin, IOTA transactions are an atomic part of the data structure. Meaning that the maximum size of any single state change is exactly one transaction, and there can be no other size of state change. Unlike Bitcoin where a state change can range from one to the max number of transactions in a block, every 10 minutes[7]. There are no intervals, so a transaction can be attached and made available for approval as and when it is needed, with no waiting for the next block. This completely negates block size/interval problem of blockchain by being a more granular data structure.

Given that when a transaction is attached it will approve two others, this suggests that the higher the rate of transactions the quicker your transaction itself will be approved. With the limits of blockchain removed from a crpytocurrency, the bottleneck of any such system now becomes how quickly can a network distribute and or acknowledge these transactions -

[6] The market cap of Ethereum at the time of writing is 31,670,253,428 USD [5]

[7] The network choosing to discard a sub-chain for a longer sub-chain could be considered a state change as well. Whereas in the Tangle transactions attached are never discarded, but can be left behind when other transactions choose not to approve them

throughputs such as the 294.93 Tx/s achieved by PayPal could theoretically be far exceeded, and allow for these payments to validated near instantaneously.

Moreover, in IOTA there are no miners - a transactor will act as a ”miner” themselves by approving two other transactions, in order for their transaction to be seen as valid. For that reason there are no transaction fees, a transactor does not need to pay them selves a fee to compute the anti-spam proof of work required. So with the notion of transaction fees completely removed from the network, we no longer discourage transactions of relatively small size - as in Bitcoin.

5.2 Pseudo decentralisation and environmental impact

With the types of participants in IOTA reduced to just one - the transactor, we no longer have the problem of any participant having more influence on the system than another. And, given how only the work needed to approve a single transaction is done per single transaction, we also solve the problem of needless energy usage as in Bitcoin mining. Meaning we do not have to entertain the debate as to the advantages of proof of work versus proof of stake, when the transactions are atomic as in the Tangle the security lies in the throughput of the network - the higher the throughput, in theory the more secure the network is. Which is why this project focusses on IOTA, rather than a token such as Nano [13], which again uses proof of stake as its security measure.

5.3 Summary

On paper, IOTA can scale to levels rivalling or possibly far exceeding that of an established payment service like PayPal - only being limited by the physical throughput of the hardware that transports the information. A problem that is potentially solved by the use of an extra layer on top of the blockchain - such as the lightning network [16], however this only solves the scalability problem, still leaves us with a weaker decentralisation claim and ultimately puts any such layer at the mercy of any protocol change of the underlying blockchain. Furthermore, IOTA does not have miners which could cause a weakening in a claim to decentralisation, and consequentially does not have transaction fees which inhibit the use of micro transactions - which is huge hurdle to overcome if a cryptocurrency could be used in everyday life.

However, the key difference between an implementation such as Bitcoin and IOTA is the atomic versus non-atomic transactional nature of the data structures, in that there is one block per transaction in the tangle and many transactions per block in the blockchain. Given the asynchronous nature of any such system, a blockchain may outperform a tangle when issue rates are low, or for a node issuing a transaction that has a higher latency relative to the rest of the network. As an entity in the tangle can only approve two transactions based on what they saw when they choose to issue - which if the combination of proof of work and latency is large enough, may mean they approve a transaction already approved by one or more others. Given the weighted random walk tip selection method used to deter lazy or malicious transactions, this could increase the likelihood of the transaction remaining unapproved. There are simulations described below that look at varying transaction throughput in the network as a whole, and others that compare the transaction approval rates and approval times for transactors with a higher relative latency to the rest of the network.

6 Simulating the tangle

While originally it was planned to produce both a blockchain and a tangle simulation, it was decided that producing just a refined tangle model would be a better use of time. The reasoning behind this is based on the fact that - as stated - a blockchain is fundamentally limited by its interval and time, a hard ceiling in terms of throughput that cannot be broken without the use of another layer such as the Lightning network. What is more interesting is how a tangle - not limited in the same way as blockchain, performs in terms of transaction approval at high and low rates of issue, and how that is affected by entities with differing relative latencies - which is not a problem of Bitcoin with its ten minute interval, but certainly could be for an implementation such as IOTA.

The simulation as detailed below, will record the number of unapproved transactions every time a new transaction is added . They also record the ages of the transactions i.e. from issue to first approval. A tangle based implementation such as IOTA differs in its ”approval” mechanism, in that a transaction in a blockchain once included in a block is considered accepted by the system (unless a new longer chain is found). Whereas, with IOTA a transaction is considered accepted by the system if its cumulative weight grows linearly. Such that after a certain point, most or all newly issued transactions will indirectly approve a transaction if it is ”accepted” - similar to how all transactions will indirectly, or directly approve the genesis transaction. Once the weight grows linearly, it is up to the vendor to decide whether or not the chance of that transaction being reversed is likely - similar to how many blocks to wait for in a blockchain.

6.1 The simulation model

The model created for this project takes advantage of the discrete event simulator omnet++. Omnet++ is an event driven simulation environment where modules representing entities interact with others, in our model we have two types of entities: TxActors(transactors) and the Tangle. The Tangle holds a master list of all currently unapproved transactions, this is copied by a TxActor any time they wish to issue a new one - which is a representation of the TxActor only having a view of the tangle at the point they chose to issue. The TxActor uses their tip selection method to select an attachment site from their view of the tangle, and once they have completed their ”proof of work” the transactions they choose to approve are updated in the master list. Depending on other TxActors who issued in the mean time, this can mean that the master list may not change at all. The master list will only change when a TxActor approves a previously unapproved transaction. This allows an asynchronous network of nodes to be emulated in the model.

TxActors have only one parameter, which is the time taken for them to compute the proof of work needed to approve the transactions they select. This is drawn from a truncated normal distribution assigned at the beginning of the simulation and remains constant throughout - this represents TxActors using the same device to issue their transactions. The inter arrival times of transactions issued by TxActors is drawn from the same exponential distribution.

The model is able to assign different tip selection methods per TxActor - although the simulations that were completed used the same method. Each simulation is run thirty times with a corresponding seed for the random distributions described above - using the Merseinne twister algorithm to generate the variates, the proof of work and inter arrival times are seeded via omnet++, while the randomness of the walks and uniform random tip selection methods were seeded from the random c++ header - again using the Merseinne twister algorithm but seeded from the current epoch time. I chose to make the tip selection methods non-reproducible in this way to ensure complete randomness of the TxActors' tip selection methods.

Omnet++ models the behaviour of the entities within the system, while I wrote a data structure in c++ to work in tandem with the omnet model. Its simply a Tangle object with a genesis transaction, and a list of current tips which the TxActors can copy when they wish to issue a transaction. Each transaction or "Tx" object has pointers to the transactions that approve it, and those it approves. Every transaction also has a a timestamps for when they were issued and first approved. This allows the recursive weight calculation function to traverse the tangle but still disregard transactions that are not supposed to be a part of the view that TxActor has - the same in the walk tip selection method.

6.2 Assumptions

In our simulations, its assumed that all transactions are honest. Meaning that in the case of the tangle simulations, there will be no mechanism for verifying transactions nor will any proof of work be completed. Compared to a blockchain, where most implementations have a protocol for deciding between conflicting transactions i.e picking the longest chain - the comparable tangle mechanism would be two or more sub-tangles merging as there is no need to pick a longer chain, meaning that transactions never become stale (just unapproved) as is the case of stale blocks[8] in a blockchain. See [8], for a more complex blockchain simulation that accounts for this, plus latency between nodes and protocols for transaction broadcasting - all which have an effect on the throughput of both a Tangle and blockchain. This project is focusing on the performance of the tangle, when transactions are comparable to today - at a throughput that Bitcoin has successfully handled in the past, when throughput is at a higher rate than Bitcoin's interval allows and finally, a rate that is higher than we should expect Bitcoin to handle.

[8] A stale block being a block found by a miner, which is then left behind when another miner finds two or more blocks in the same time frame

Were we to implement Blockchain simulation, it would have to be assumed that the transactions accepted into the blocks have paid a fee, again this does have an effect on which transactions are accepted. It's known that the more pending transactions there are, in general the higher fee the issuer will have to pay. However it's been shown that the relationship between transaction fee paid and acceptance is not so straightforward, often with miners adding transactions seemingly regardless of fee paid see [10][9]. In terms of performance however, it would be enough to know that some transactions will be added at the end of each block interval, with no more than the block size allows.

6.3 Choosing the simulation parameters

As addressed previously, one of the problems problem with blockchain based cryptocurrencies is that they do not facilitate micro transactions - if we were to envisage a future where the internet of things grows ever larger, for cryptocurrencies to face widespread adoption they must be able to handle many small transactions without paying a disproportionately large transaction fee. The two experiments to be run using these parameters will be a throughput experiment - to see how different rates of transactions issued affect the tangle, and a latency experiment where we imagine many devices with much lower compute times - to better model the effect of a difference in their latencies.

To follow on from that, it seems logical that in a future where there are many devices looking to make many small payments, any such device would be optimised to make those payments. Therefore there are versions of the simulation with different parameters to account for this, in a situation analogous to the present there is a relatively modest transaction rate, and devices making payments are not built with the payments in mind. So in the case of the throughput experiment, the nodes will have a highly variable proof of work computing time, in order to represent the many non-optimised devices being used to make payments, while the latency simulation will have a very low constant time[10]. Entities in the throughput simulation will be assigned a proof of work time drawn from a variable distribution. While in the latency simulation - given the assumed presence of optimised hardware, the time will be constant at 1 second. These are estimates based on benchmarks of Hashcash, which given ideal conditions can be a fraction of a second - depending on the subject of the hashing algorithm[9]. The soundness of the compute times is not so much important as the difference between a network with a high computing time, to that of one with a comparatively low compute time.

The transaction issue rate per entity stays the same, with only the number of entities issuing transaction s changing from experiment to experiment- to represent more widespread versus lower levels of adoption but still entities issuing transactions independently. Sim-

[9] This is a characteristic endemic to all crypto implementations, that there is no centralised rulebook enforced other than the protocol, how this protocol is adhered to is up to the individual nodes. Miners in particular are allowed to choose which transactions they include in the block they try to solve. The synonymous characteristic in the Tangle is the tip selection method used by entities.

[10] Even now Bitcoin mining is done on highly optimised machines. The protocol ensures that the block interval is relatively constant in either case.

ulations will end after a predetermined number of transactions have been issued by the transactors.

In both experiments, the tip selection method used is a Markov chain process and the governing factor is the alpha value, a higher value means a more deterministic walk, whereas a lower value means a more random walk. A lower value is preferred over a high value, but too a low value means there is no incentive for entities to attach their transactions honestly. The Tangle simulations where the Markov chain walk process is used will have an alpha value of 0.1, found to be deterministic enough to discourage lazy entities and provide security against attacks and lazy entities, yet random enough to avoid honest entity's transactions getting left behind at an unacceptable rate [12]. The Markov process involves computing the weights of every potential site to visit, this is extremely expensive in terms of computation, it involves traversing the DAG using recursion. With an alpha value of zero, there is no need to compute the weights as every site chosen will be chosen at random, it is similar but not identical to uniform random tip selection - as a backtrack is still made into the DAG to serve as a starting point for the walks - were we to simulate a tangle with a transaction rate similar to that of PayPal a walk selection with no weighting would need to be used to ensure reasonable performance of the simulation itself.

The alpha value determines the decision making process of the Markov chain process, but the walk depth will determine the length. The start points of the walks are determined by a uniformly random backtrack to a depth of 60 transactions backwards into the tangle. A minimum number of backtracks of 20 was found to provide enough randomness in the Markov chain process [12] - 60 was chosen to ensure this randomness.

Without empirical data determining a rate of transactions per second per entity, I assumed the same inter arrival time for each entity in both experiments. This makes the data somewhat less meaningful, as the issue rate for each simulation has been reverse engineered to show present day versus potential future scenarios and compare against those seen by centralised systems like PayPal.

As stated, network latency plays a role in both Bitcoin and IOTA. This latency should play a bigger role in IOTA, as the tip selection methods used to determine attach sites depends on the view of the tangle the entity has. For an entity to have a higher latency relative to the rest of the network should mean that that entity will have a view that lags behind the rest. This is not so much an issue with Bitcoin as the onus is on the miners to distribute and include transactions in a block. So if a transactor with high relative latency issues a transaction, it is extremely likely the miner nodes will be aware of such a transaction and start including it in their blocks within one block interval(given an appropriate fee is paid and the transaction is not a double spend). Given that the comparable Tangle mechanism to "finding a block" is attaching an atomic transaction, the latency of the issuer plays a much larger role in how likely that transaction is to be seen and therefore accepted by the greater network i.e. not left behind because it approved a transaction it saw as unapproved, when entities with a lower latency may have already directly or indirectly approved this transaction several times.

6.4 The Simulation

The experiments outlined below are a means of validating the claim that a cryptocurrency such as IOTA can solve the problems of blockchain. The parameters are chosen to represent different scenarios in terms of adoption and throughput, to see how both data structure performs - with scalability in mind. As mentioned, network latency could be a bigger issue in IOTA than blockchain, the second experiment is parametrised to test just this.

The ages of transactions in the tangle will be calculated by computing the weights of a subset of transactions over time. As how many approvals a transaction needs to be considered accepted is a vendors choice[11], we will simply track the weights over time and consider them approved when their weights increase linearly.

6.4.1 Throughput

This experiment is to determine how the Tangle performs at varying throughputs. The simulations were run at 1 Tx/s, 3 Tx/s and 10 Tx/s using both uniform random tip selection and the weighted random walks. The rates chosen reflect rates that the Bitcoin blockchain can comfortably handle, and a rate at which it is known to suffer. Three rates were chosen to enable a line of best fit to be drawn in the analysis stage, where we can then extrapolate the results to the rates seen by systems such as paypal.

We should expect to see transactions using the walk tip selection have higher variance in there approval times than those of the random selection. As stated, using some form of weighted selection is imperative to prevent malicious actions within the tangle, however it is useful to be able to compare the two.

6.4.2 Relative latency

In the model created, we can think of the latency of a transactor to be the time taken to compute the proof of work for an individual transaction - given that the transactors are connected only to the Tangle object, rather than interconnected and broadcasting transactions to each other. In order to compare how the latency of a node affects its issued transaction likelihood of approval, we take a small subset of the network and give them a higher proof of work time than the rest. With the normal nodes having a constant compute time of 1 second, while 10% of the network - the "slow" nodes - have a higher compute time. Here four simulations are run only using the weighted random walk tip selection: a control scenario where all transactors are normal, slow nodes compute at 1.5 seconds, 2 seconds and finally 3 seconds. All the simulations run at a throughput of 1 Tx/s. Assigned constant times for all nodes were chosen to eliminate any noise in the results, to better be able to observe the affect of a high relative latency.

[11] Similarly with blockchain, the more blocks added past the block your transaction was included in, it becomes exponentially less likely a longer chain will be found. 6 further block confirmations has been shown to have less than a 1% chance of reversal [15]

7 Results

Here we see how three different metrics: time as a tip, tips seen and approval time - describe the state of the tangle across the two experiments. For the most part what is described is as expected, with the exception of the 10 Tx/s throughput simulation. Otherwise the model produced a sound tangle data structure for appraisal.

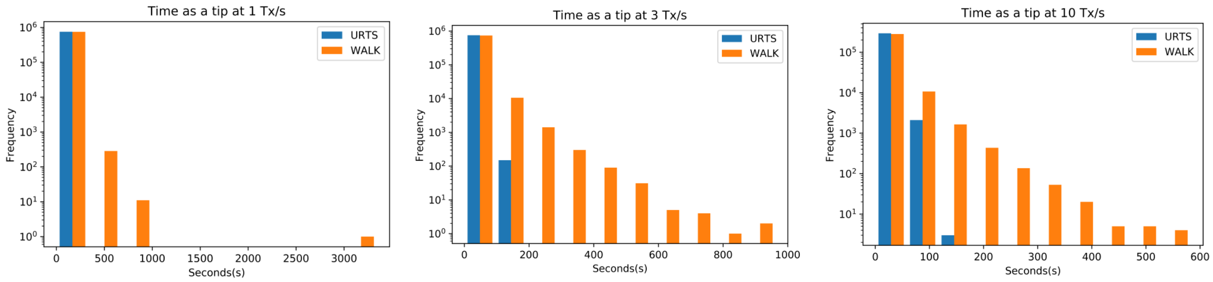
7.1 Throughput

In the throughput experiment, we see how a uniform random tip selection method compares to a weighted random walk method at different throughputs. This was important not only to show the model performing as expected, but to highlight the drawbacks in performance that result from using a weighted random walk - which is a necessity for any tangle instance to punish lazy or malicious nodes.

It's worth noting that the model at 10 Tx/s did not perform well in terms of time to simulate. To have the simulation finish in a reasonable time the decision was made to halt each run after only 10,000 transactions - as opposed to 25,000 for the other throughputs. Meaning the results at 10 Tx/s a second are potentially compromised, more on that at each metric.

7.1.1 Time as a tip

Time as a tip is how long a transaction remains a tip, or rather the time taken for it to be selected as an attachment site by another transaction.



(a) 1 Tx/s

(b) 3 Tx/s

(c) 10 Tx/s

Figure 1: Time as a tip for each transaction

In Figure 1, we can see that in the URTS runs, the time as a tip was both faster and displayed less variance than the walk. This reflects how all tips are treated equally in URTS, even a tip attached long ago has an equal chance of being selected as an attachment site. Whereas the walk selection method places a higher value in transactions that attach to tips with a higher weight, with the attachment site being where a walker first finds a tip. This mechanism allows for a tip to be left behind. Or rather, a higher chance of that tip remaining so for longer, as is not the case in URTS.

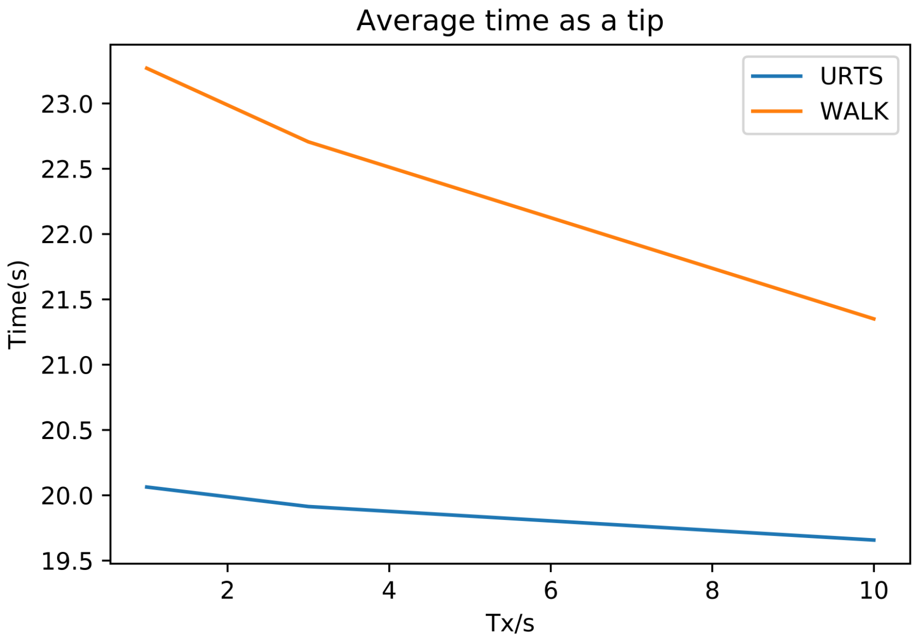
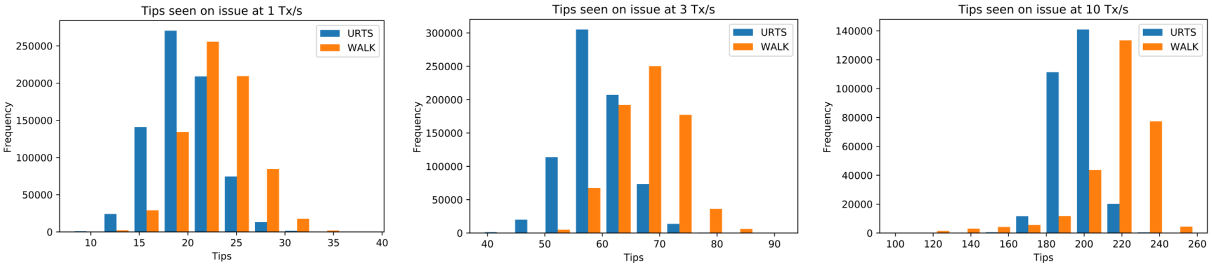


Figure 2: Average time as a tip

Given the disparity as shown in Figure 1, Figure 2 shows how as the throughput increases, so too will the time as a tip decrease, which again is the expected result.

7.1.2 Tips seen on issue

Tips seen on issue is defined as how many unapproved transactions the issuer saw when they decided to issue a transaction, consequentially also the tips they use to either run URTS or compute a start point in the walk tip selection.



(a) 1 Tx/s

(b) 3 Tx/s

(c) 10 Tx/s

Figure 3: Tips seen per transaction issued

In Figure 3, we can see that in both URTS and walk runs, the tips seen by transactions form a normal distribution, with a lower mean in all the URTS experiments. The walk runs having a higher mean is consistent with the expectation that given a bias towards transactions with a higher weight in their walk, it is the case that transactions issued at a similar time (with many of the same tips in each others' view) will approve the same one, therefore leaving less attractive attachment sites as tips.

In Figure 3c we see more distorted distribution of tips seen, with many results skewed towards a lower number - this is likely a result of the number of transactions issued being inconsistent with the other runs.

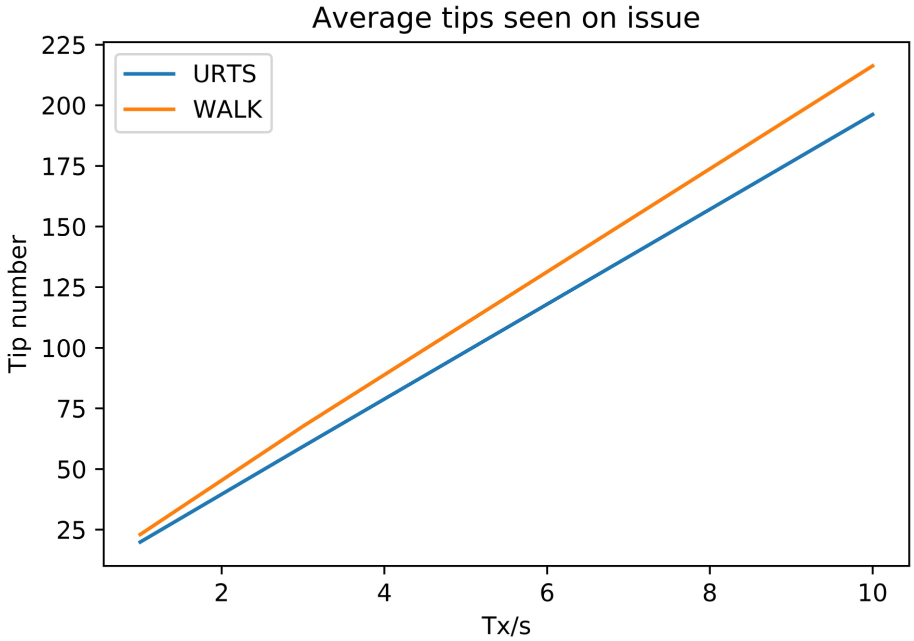


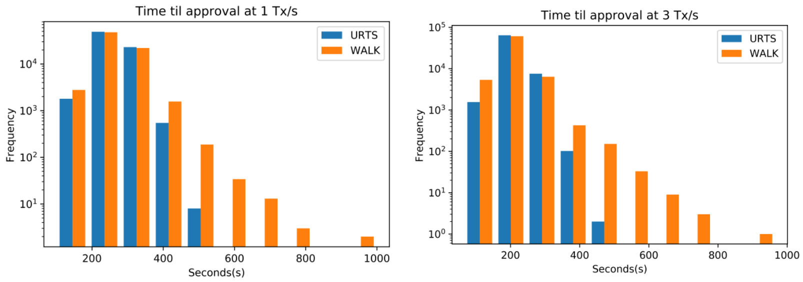
Figure 4: Average tips seen on issue

Figure 4 shows the number of tips in both URTS and walk increase linearly with the throughput. The walk method shows a higher number of tips which diverges upwards from the URTS, this is expected given the bias towards and away from some tips as shown in Figure 3.

7.1.3 Approval time

Approval time in a cryptocurrency can be somewhat arbitrary metric, in that once a transaction becomes a permanent part of the data structure, the longer it remains so it becomes exponentially more likely that will not change[12]. Given that in our simulations there are no malicious entities trying to form malicious sub tangles, we can take the approval time to be from the moment of issue, until the moment when the weight of a transaction starts increases linearly.

The weight of 10% of all transactions was recorded every 100 transactions issued in all the experiments run. A subset was chosen as opposed to all transactions, because computing the weight of an individual transaction meant traversing the tangle recursively, which was computationally expensive and increased exponentially as more transactions were issued.



(a) 1 Tx/s

(b) 3 Tx/s

(c) 10 Tx/s

Figure 5: Time taken for a transaction to be a approved

We can see from Figure 5 that the bulk of transactions were approved in a similar time in both the URTS and walk runs - with the walk runs showing a small number of transactions with relatively high approval times. Again this is to be expected, in a similar way to how the time as a tip distributions differed in Figure 1, a walk tip selection method means some tips will take longer to approve than others - as 5 shows.

[12] In a blockchain, a transaction being included in a block would be becoming part of the data structure, the more blocks added past that point the less likely a longer chain will be found to replace it. All incoming transactions approving a given existing transaction would be the synonymous characteristic of a tangle

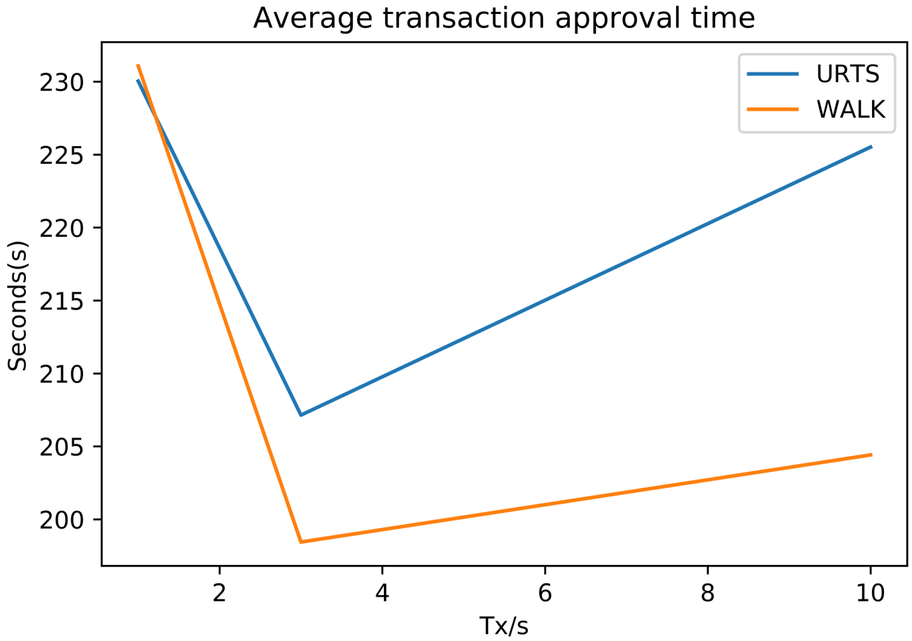


Figure 6: Average time until approval

Figure 6 shows our first unexpected results, firstly that the walk tip selection method starts to outperform URTS at higher throughputs. And secondly that at 10 Tx/s the approval time starts to increase again - dramatically in the case of the URTS. As stated the 10 Tx/s simulations were only run until 10,000 transactions given the time allowed, so either these results are a consequence of not letting the tangle run for long enough to obtain a sound average - or there is a mechanism that prevents the approval time from falling below a certain point.

7.2 Latency

In the latency experiment, we see how a subset of the network with a high relative latency affects their own transactions, transactions of others and the overall structure of the tangle. Latency in this model is analogous to a combination of node's proof of work compute time and "latency" to the rest of the network (given the simplified nature of the model). The low latency group can be thought of as a collection of nodes geographically close together, while the high latency nodes are those connected to that collection but geographically distant.

There are four simulations run: a control experiment where all nodes' latency are 1s, high latency nodes' latency are 1.5s, then 2s and 3s respectively. The results are visualised below, in a similar way to the throughput experiment as a means of comparison. While the throughput experiment had latencies drawn from a truncated normal distribution to represent a varied network, this experiment has a constant time of 1s for all normal nodes.

The distributions of the metrics recorded did not change in any significant way across the parameter values, so only the control experiment is plotted for each. Their is an observed effect best describe as a simple line plot.

7.2.1 Time as a tip

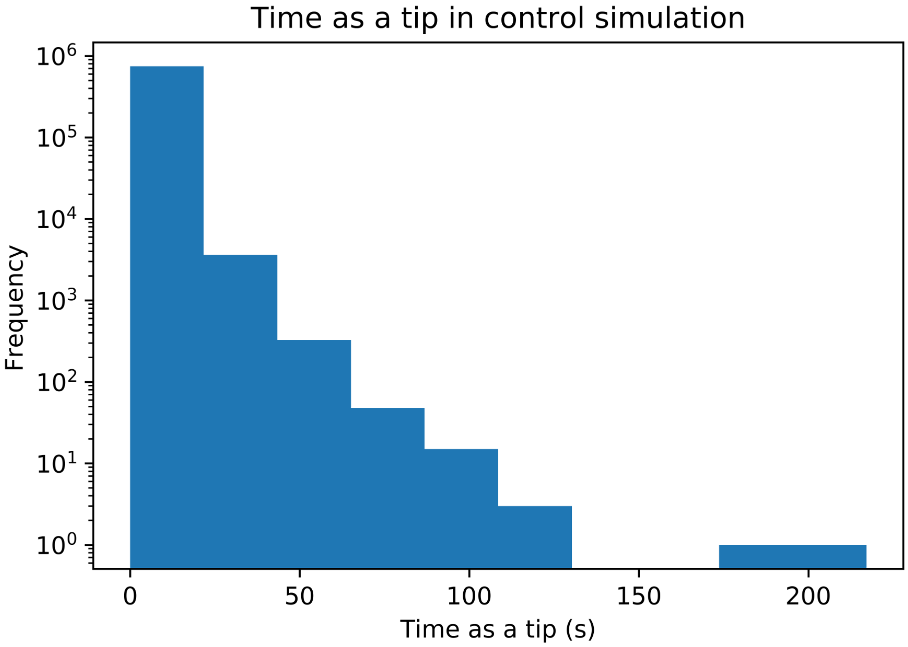


Figure 7: Time as a tip in control: Walk tip selection with alpha 0.1

When we compare Figure 7 to Figure 1a we see a similar distribution but a vastly smaller range of values. Given that the tips seen increases with a higher number of transactions approving the same tips, this is an expected result as all the transactions have a 1s latency, meaning that fewer nodes will have the same view of tips at the same time.

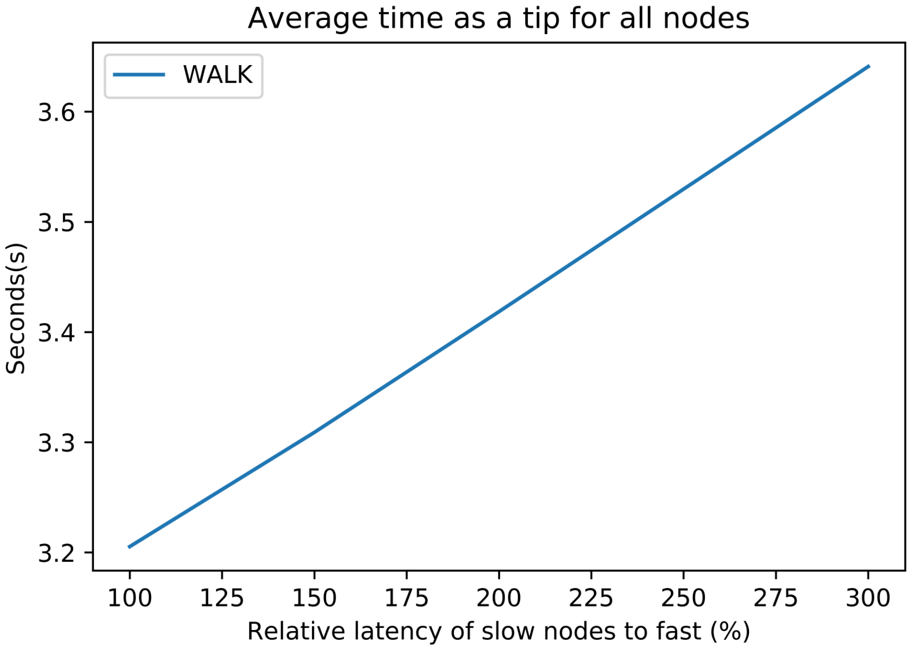


Figure 8: Average time as a tip for all nodes

Figure 8 shows that a small subset of higher latency nodes in the network affects time as a tip for all nodes, not just themselves. A linear increase while small, still shows an observed affect, and as we saw in the throughput results, increased time as a tip seems to be symptomatic of an increase in approval time, and tips seen.

7.2.2 Tips seen on issue

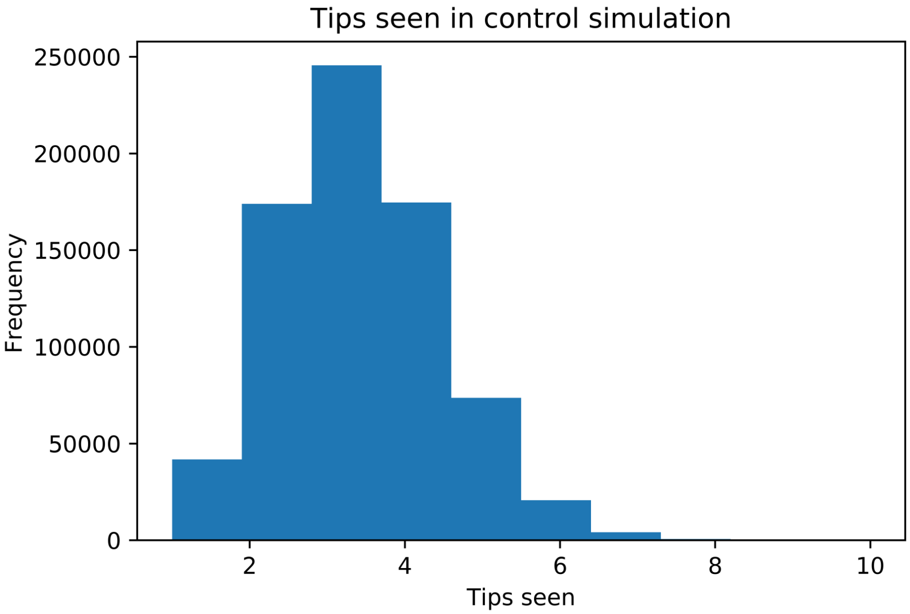


Figure 9: Tips seen in control: Walk tip selection with alpha 0.1

Figure 9 shows the tips seen in the control experiment are as expected, again with a similar distribution to Figure 3a with smaller values given the lower, constant latency of all nodes which creates a normal distribution with much lower standard deviation (As opposed to the throughput experiment where some nodes would have a high latency creating a much higher variance in the distribution).

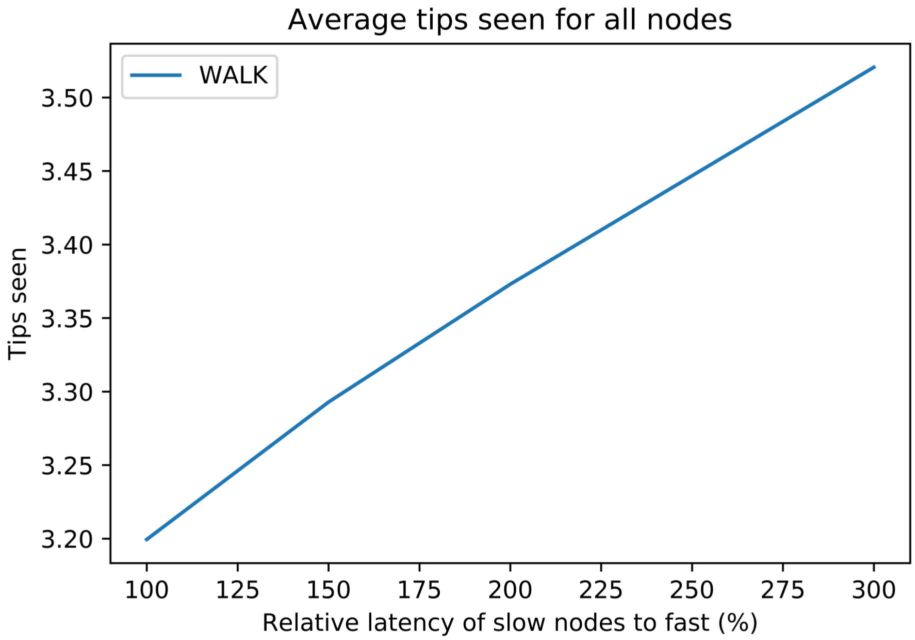


Figure 10: Average tips seen for all nodes

In Figure 10, as expected we see a small linear increase in the average number of tips seen when the relative latency of slow nodes to fast is greater. This corresponds to the fact that there is a small subset of the network who constantly have a view of the tangle that lags behind the rest, meaning they are more likely to approve a tip previously approved - thus increasing the tips seen for the next issuer.

7.2.3 Approval time

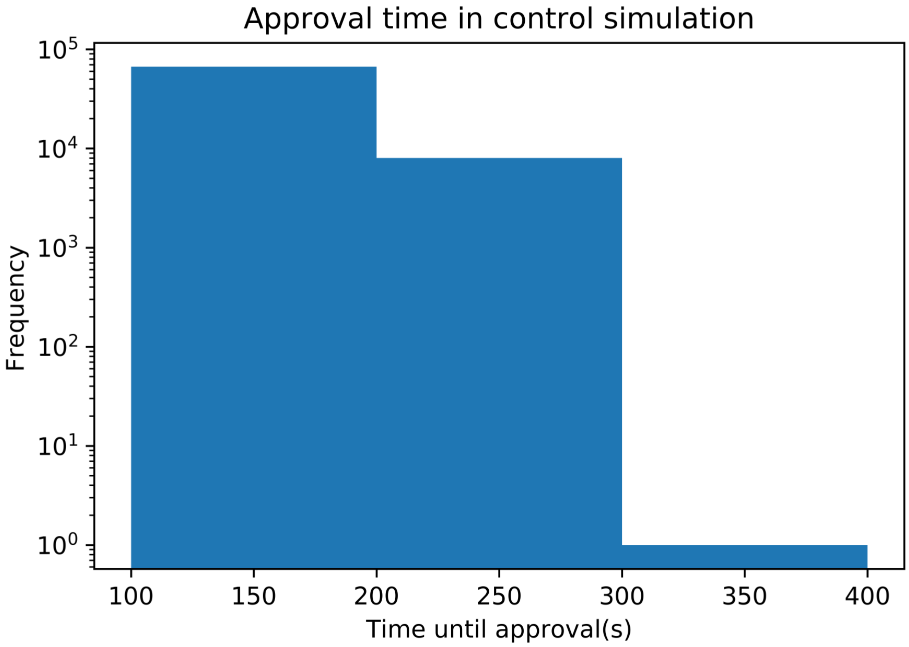


Figure 11: Time until approval: Walk tip selection with alpha 0.1

Figure 11 being consistent with other metrics of the experiment - exhibits a similar distribution when compared with Figure 5a, again with much smaller variance because of the consistency of the node latencies.

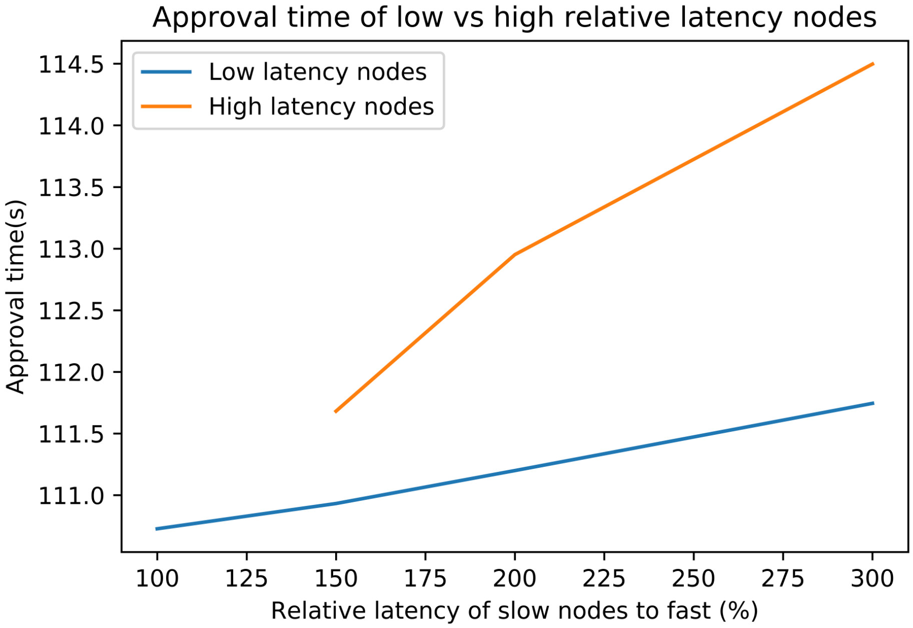


Figure 12: Average approval time per simulation: Walk tip selection with alpha 0.1

Given that we have seen how a constant latency throughout the network affects the time as a tip and tips seen, the results in Figure 12 are as expected. We see a small linear increase in the low latency nodes' transaction approval times, while we see a greater increase in the high latency nodes' approval times.

8 Analysis

8.1 Throughput and relative latency

We should expect to see the average transaction approval time decrease with higher transaction rates, but the results from Figure 6 show otherwise. While a decrease is shown between 1 Tx/s and 3 Tx/s, in both the URTS and walk runs we see a marked increase in approval times at 10 Tx/s. As stated the 10 Tx/s runs were only simulated for 10,000 transactions as opposed to 25,000, and we get the anomalous result at this rate. Occams's razor would suggest that the result is not sound, when the alternative is mechanism or tangle property which has not been mentioned in any of the literature the author has read.

What is interesting however, is when we look at the control results from the latency experiment. With all the nodes having a constant latency of 1s, we get the majority of the approval times being roughly half that of the comparable throughput experiment with variable latencies. Given the results of the latency experiment where we observed an increase in the approval time for high latency nodes, and consequentially a smaller increase for the rest of the network. We can say that in a real tangle, the speed of approval for a transaction depends largely on the rate of transactions being issued, but to some extent is adversely affected by a combination of the node's latency to rest of the network i.e how up to date it's view of the tangle is, and the time it takes to compute the necessary proof of work. The longer this time is, the slower on average a transaction's approval will be. We can also say that even a small subset of the network issuing transactions with a high relative latency to the rest, will slow down the approval times of the rest of the network.

We should expect to see URTS perform much better across the board than the walk selection, however we see in Figure 6 that this appears to not be the case. This is explained by how I calculated the approval times, only those transactions that became approved were counted, meaning that in the case of URTS, it is entirely plausible that a tip might be unlucky and never become approved (or was issued near the end of the simulation), yet this is much less likely than the same thing happening in the walk simulations. Meaning that more extreme values are excluded when we calculate the mean of the walk scenarios, which results in the average being unfairly shifted below the URTS.

8.2 Summary

To summarise, the throughput experiment observes a decrease in transaction time approval from 1 Tx/s to 3 Tx/s, unfortunately the 10 Tx/s experiment was compromised and as such, we are unable to definitively extract a meaningful function to denote how throughput affects transaction approval time. What we can see is that at both 1 Tx/s and 3 Tx/s the approval times were lower than the Bitcoin block interval. If we assume that when the throughput of a blockchain allows a block to be appended without being full, and that all miners include newly received transactions in the block they try to find as soon as they see them. Then a Bitcoin transaction's approval time would be anywhere less than 600 seconds - depending on when the transaction was broadcast. At high throughputs where every block is filled to the max, then the average approval time would increase depending on how long the high throughput was maintained, and how many transactions were left out of a block at every attachment. Compared with a tangle where transactions are attached atomically with no block interval, and approval depends on a transaction being directly or indirectly approved by enough other transactions, at higher through puts we'd expect the approval time to decrease further - as it does between 1 Tx/s and 3 Tx/s in the simulations.

In the latency experiment, we observe an effect not just on the transactions issued by high latency nodes, but also an effect on the low latency nodes as well. The latency experiment was designed to show that effect by itself. As opposed to the throughput, where the latencies were drawn from a distribution, we can think of the metrics seen in the throughput as an amalgamation of the observed effects seen in the latency experiment. The latency experiment also revealed that when latencies are much lower throughout the whole network - the 1 second constant time in the control for example - shows a further, significant decrease in approval times.

9 Conclusions

9.1 Problems solved and problems identified

The soft problems of a blockchain based payment system like Bitcoin, are the transaction fees discouraging micro payments and a weaker decentralisation claim resulting from the miner and issuer disparity. These problems are inherently linked in that without miners there would be no propagation of transaction or new blocks appended to the chain. Coupled with the fact that miners are motivated to participate with the promise of a block reward (which eventually will be zero), and the transaction fees paid in the block found.

With a cryptocurrency such as a IOTA using a tangle, these soft problems are mitigated. Without the use of miners that are motivated by monetary reward, and a proportional computation per transaction, IOTA and the tangle have a much more solid foundation in their design - from a payment service use case perspective.

However, these soft problems being solved mean nothing, if the system does not scale appropriately. The experiment on throughput, did show a decrease in approval time at higher throughputs until the flawed result, so while the author is not in a position to definitively say that a crytpocurrency such as IOTA can scale to rival established centralised payment systems - the trusted results show that even at lower throughputs the approval time can be much faster than the block interval of Bitcoin.

What is clear, is that the main factor determining the success of a tangle is the interconnectivity of the nodes. We showed that when latencies are low and constant for all nodes, approval time were significantly faster than when the approval times were slower and more varied. The latency experiment showed that even a small proportion of the network with high relative latency affects their own transaction approval time and the network as a whole. To conclude, for the tangle to serve as a solution to the problems of blockchain as a payment service, nodes would need to be as evenly distributed as possible - with more nodes issuing transactions the better. Any geographically isolated nodes will have higher approval times - and negatively affect the rest of the networks.

In regards to the selection of experiments and the parameters supplied, the model was constrained by low performance at high throughputs - which meant that the range of parameters supplied to the throughput experiment was limited and as such the results and conclusions drawn are not as sound were it to be otherwise. This experiment was not an appropriate use of the model that was constructed, and more valuable data could have been extracted by focusing on the latency experiment - with different sized subsets of the network with high latencies for example. As we did observe an effect on the overall performance of the network, it would have been a far better contribution to see what extent that performance is affected at other rates, sized subsets and higher relative latencies.

9.2 Recommended further work

This project has not been able to prove that a tangle will solve all blockchain's problems - only that it potentially could, but questions remain unanswered. What it has done is identify the problems that an atomic data structure like a tangle has, or rather the main factor that determines its success - the structure of the network of nodes itself, which is not itself a problem for those issuing transactions in a blockchain based implementation.

With that in mind, to fully answer the question, the author recommends a simulation that models the propagation of transactions from node to node - and tests different geographical distributions at different throughputs. Using a simulation such as this, we could model the effects of latency and proof of work compute time separately - rather than as one as in this project.

Using the model as described above, one could then more accurately model the effect that different proportions of the network having different latencies has on the tangle. It is the authors view that the simplified models such as those seen in this project provide insight into the make up and structure of the tangle - and theoretical throughputs - but again, given the atomic nature of transaction attachment, only with a model that simulates a network first and a tangle second, could we definitively answer the question. If I were to start the project over from the beginning knowing what I have learned throughout, I would create a model as described above using omnet++ - which has excellent support for networks - to better undrstand the latency effect on nodes issuing transactions. I would also write a bespoke simulation in pure c++ - similar in form to the one produced for the experiments in the project, to model transaction rates at levels that the omnet++ model could not compute in reasonable time.

10 Reflections

I chose the subject of this project because I find cryptocurrencies interesting, especially those designed for use a payment system. A mature version of such a technology could have far reaching and global consequences were it to be achieved, but given the state of many of the most well known crypto tokens such as Bitcoin, we clearly have a long way to go before such technology could be considered so.

As such, the general philosophy of this project was to bring some rationality to the debate as to whether or not a cryptocurrency as a payment service is viable, and to shed light on possible issues the next generation of technologies such as IOTA face to achieve such viability. This was done through a combination of reasoned argument and experimental simulations.

10.1 Lessons learned

While this project was able to identify some of the problems facing the tangle, it was not able to quantify specifically how, and to what extent these problems affect the viability of such a data structure as a payment system. The reasons for which are examined below.

10.1.1 Project scope and time management

Initially I had wanted to produce not only a simulation model, but a network of interconnected devices to emulate a real life tangle network. I also wanted to use the a tool known as Blocksci[10] to find an average inter-arrival time for nodes issuing transactions in the Bitcoin network. I spent a lot of time trying to figure out a way to do this, as using the Blocksci parser involved renting an Amazon web server and learning how to use a sparsely documented api to interact with the database. Ultimately I decided that all that work for a single parameter in the simulation model would not be worth it - given the short time available. As for the network emulation I decided that doing so would be interesting, but would not necessarily provide any meaningful contribution in terms of useful data. In the end modelling the effects of varied parameters in the simulation, provided the most insight even without comparable real world metrics to use within it - this was realised quite late in the project as outlined below.

With the scope of the project potentially so large it wasn't until the simulation model started to emerge as complete that I was in a better position to plan experiments, which was late in the project given the time spent researching the methods needed to complete the full scope I had planned. Were I to undertake another similar project, I would try to identify the minimum that the project hoped to achieve, and re-evaluate at determined intervals how likely the minimum was to be achieved in a given time frame and, whether any stretch goals could be achieved without compromising the integrity of the project.

10.1.2 Simulations as experiments

Constructing a simulation model has certainly been a learning experience for me and as explained in the conclusions of this project I would do things differently if I were to start over. I had problems with the simulation running at high throughputs which stemmed from the omnet++ model's management of modules issuing events. Omnet++ is more suited to modelling the events of a network and while this was adequate, it left the model I created in a position where it wasn't particularly efficient at running either of the experiments.

Given another chance, I would focus more on utilising omnet++'s network capabilities in order to create a model that could simulate varying distributions of nodes in an easily configurable way. As the results show, the number of simulations with different parameters did not allow for efficient plotting of the data - or rather it struggled to be able to plot sound lines of best fit in order to extrapolate and make predictions about a tangle in different situations, hence my recommendations for further work.

After identifying node latency as a potential problem, I should have focused on that, or perhaps extended the experiments with greater distributions of latencies and varying proportions of nodes with them. It is clear to me now that having done so would have enabled a better contribution and, better answers to the questions posed in this project.

10.1.3 Strengths and weaknesses

Over the course of the project I had to learn many new skills such as project management and discrete event simulation, while also relying on those I already possessed. In terms of programming I already had a rudimentary grasp of C++, but working with software such as omnet++ required a much greater understanding to effectively use. As such my knowledge of C++ has increased dramatically, and I would count programming in general as one of my strengths throughout the project - as I also used Python's pandas library to generate the data visualisations seen in the results section.

One of my weaknesses that should be apparent is a lack of statistical analysis skills, these were improved over the course of the project as for example, I had to learn the difference between a normal distribution and an exponentially distributed random variable. While this was key to creating a sound, reproducible simulation (seeding the random number generators is a good example) my lack of knowledge in this area becomes particularly apparent when we look to analyse the results. Planning the experiments would have been easier if I had a greater grasp of what is required to quantify the effects predicted and seen in the experiments.

References

[1] bitinfocharts.com. Bitcoin average transaction fee historic chart. Available: https://bitinfocharts.com/comparison/bitcoin-transactionfees.html Accessed: August 2018.

[2] Blockchain.com. Average number of transactions per block. Available: https://www.blockchain.com/charts/n-transactions-per-block Accessed:Auguest 2018.

[3] Rory Bowden, Holger Paul Keeler, Anthony E Krzesinski, and Peter G Taylor. Block arrivals in the bitcoin blockchain. *arXiv preprint arXiv:1801.07447*, 2018.

[4] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform. *Available: http://ethereum.org/ethereum.html*, 2017.

[5] coinmarketcap.com. Ethereum stats. Available: https://coinmarketcap.com/currencies/ethereum/ Accessed: August 2018.

[6] Alex de Vries. Bitcoin's growing energy problem. *Joule*, 2(5):801–805, 2018.

[7] Evan Duffield and Daniel Diaz. Dash: A privacy-centric crypto-currency, 2014.

[8] Arthur Gervais, Ghassan Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. On the security and performance of proof of work blockchains. In *Proceedings of the 23nd ACM SIGSAC Conference on Computer and Communication Security (CCS)*. ACM, 2016.

[9] hashcash.org. hashcash benchmarks. Available: http://www.hashcash.org/benchmark/ Accessed: August 2018.

[10] Harry Kalodner, Steven Goldfeder, Alishah Chator, Malte Möser, and Arvind Narayanan. Blocksci: Design and applications of a blockchain analysis platform. *arXiv preprint arXiv:1709.02489*, 2017.

[11] B Kusmierz. The first glance at the simulation of the tangle: discrete model, 2017.

[12] Bartosz KuSmierz, Philip Staupe, and Alon Gal. Extracting tangle properties in continuous time via large-scale simulations. *Available: https://www.iota.org/research/academic-papers*, 2018.

[13] Colin LeMahieu. Nano: A feeless distributed cryptocurrency network. *Available: https://nano.org/en/whitepaper*, 2017.

[14] Eric Lombrozo, Johnson Lau, and Pieter Wuille. Bip: 141 segregated witness (consensus layer). *Available: https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki*, 2015.

[15] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Available: https://bitcoin.org/bitcoin.pdf*, 2008.

[16] Joseph Poon and Thaddeus Dryja. The bitcoin lightning network: Scalable off-chain instant payments. *URL: https://lightning.network/lightning-network-paper.pdf*, 2016.

[17] Serguei Popov. The tangle. *Available: https://iota.org/IOTA_Whitepaper.pdf*, 2018.

[18] statista.com. Paypal's net number of payments from 1st quarter 2014 to 2nd quarter 2018. Available: https://www.statista.com/statistics/218495/paypals-net-number-of-payments-per-quarter/ Accessed: August 2018.