Spring 4-27-2018

Use of the Proof-of-Stake Algorithm for Distributed Consensus in Blockchain Protocol for Cryptocurrency

Spencer J. Hosack
spencer.hosack@gmail.com

Use of the Proof-of-Stake Algorithm for Distributed Consensus in Blockchain Protocol for Cryptocurrency

Submitted to the Honors Department at
The University of Connecticut
to fulfill the requirements for
B.S. Finance with Honors

Spencer Hosack
spencer.hosack@uconn.edu
Professor Yaacov Kopeliovich
Finance 4997W
28 April 2018

Abstract

Recent increased attention to Bitcoin and other cryptocurrencies has opened investors and the general public to the realm of digital currency. Greater exposure around the world has led to a frenzy of entry into the market and a test into the long-term feasibility of Bitcoin being able to remain a functioning peer-to-peer (P2P), decentralized currency. Its main structure is supported by the Proof-of-Work (PoW) protocol in which users can elect to participate in determining transaction approval and ensuring an honest blockchain. This system relies on elected users, or miners, to expend great computational power and energy in order to solve puzzles to prove the accuracy of the network's transactions and create new blocks, to then be rewarded with newly created Bitcoins for their effort.

Each cryptocurrency uses their own method to ensure blockchain accuracy, and this paper will focus on how a Proof-of-Stake (PoS) protocol is a superior algorithm to PoW by assigning mining ability equal to one's stake within a coin, rather than her energy consumption, among other factors. We will discuss Bitcoin and PoW as a baseline for our eventual analysis of PoS in terms of advantages and performance metrics. The main factors that can be compared between the two protocols is how each system can prevent itself against a variety of attacks from adversarial users within the network, as well as long-term sustainability. Vulnerabilities in any network protocol can result in adversaries being able to alter the blockchain for their own benefit, at the expense of the majority.

Finally, we will use the Cardano (ADA) cryptocurrency by IOHK as a practical case study for understanding how their Ouroborous Praos PoS protocol works. Our goal is to show how long-term adoption of PoS framework is more realistic from an energy perspective than PoW, however inherent risks still exist within the algorithm.

Table of Contents

1. Introduction

1.1 Emergence and History of Digital Currency

While the emergence of Bitcoin in recent years has piqued the conversation for a decentralized monetary system, the concept of a non-traditional digital currency can be observed as early as 1983. David Chaum, a cryptographer from the University of California, Berkeley, first published a research paper which discussed the viability of anonymous communication, followed by a publication on a secure digital cash proposal. His work also included various cryptographic protocols which eventually led to the founding of DigiCash, an electronic money company based out of Amsterdam which lasted until the late 1990's (Griffith, 2014).

The focus of Chaum's initial 1981 research is the proposal for an anonymous communication network using public keys to encrypt payments in an "envelope". These networks function with a group of senders presenting an encrypted message and the desired recipient to the initial server. The initial server will take the message and add layers of masks which must be solved by the recipient in order to understand the contents of the envelope (Chaum, 1983).

We will discuss the viability of a PoS based protocol for a Blockchain ledger as a response to the limitations of bitcoin and other PoW cryptocurrencies for long-term energy reduction. The main improvement with a PoS system is a matter of lower overall energy consumption and mining requirements. In order to "mine" a coin to approve transactions and increase the public ledger, the PoW miner must invest large amounts of capital into both computational hardware and electricity. The greater computational (or "hash") power a miner has, the better chance she has to receiving a block and obtaining her miner fee. Simply put, the ability to have influence and power within a PoW-blockchain is independent of one's holdings

within the particular cryptocurrency. Later, we will find this as a flaw in PoW due to the various types of “attacks” to be considered if an elected block is assigned to an adversarial miner. We define an adversarial miner as one who looks to disrupt the present blockchain by signing her private key to an incorrect ledger. A move in this manner would allow the adversary to falsify the blockchain for her own benefit, and allow for double spending.

1.2 Definition and Importance of Distributed Consensus

To understand the goal of decentralized currency, one must first recognize the importance of the term *distributed consensus*. This term is the problem that is required to be solved under blockchain protocol such as Bitcoin to ensure that transactions are considered valid and good. We define distributed consensus therefore as a “global agreement between many mutually distrusting parties who lack identities and were not necessarily present at the time of system set up” (Poelstra, 2015). Per the original writings on the Bitcoin protocol, discussed later, distributed consensus can be achieved within the blockchain network by order-time of transactions (Nakomoto, 2009). So long as a transaction can be agreed by multiple parties to have occurred at a given time, any secondary transaction attempting to use the same coin value later on can be determined invalid. Bitcoin differs as the majority of parties are assumed to be trust-worthy and honest, but agrees with the anonymity aspect as users are masked within the system and its cryptography helps to ensure authentication of its currency (Poelstra, 2018).

1.3 Virtual Currency vs. Digital Currency

It is important to note the differences between some common phrasing used to describe any non-physical type of currency, so we will identify how *virtual currency* and *digital currency* alter from one another. The variation in meaning exists similarly to the relationship between

squares and rectangles; all squares are rectangles, however not all rectangles are squares. In our case, all virtual currencies are digital, but not all digital are virtual. Virtual currencies are considered to be used in cases for entertainment and gaming purposes in online worlds that do not deal with real-life. These currencies are unable to be exchanged for real assets, and exist solely in a "fun" nature to be dealt in game settings (Wagner, 2014). Centralization and authoritative power is a major component to virtual currencies in games, as there is normally one group in charge of the money supply (i.e. developers) that can increase or decrease the money supply on a moment's notice by changing price of goods or reward structures in the game. Users could still exchange this "fake" money for real assets or cash in real-life for game usage, however it can lead to legal concerns as there is tax jurisdiction on the virtual currency normally (Wagner, 2014).

Digital currencies provide far more application ability for users than the value of virtual currencies, with the main differences being decentralization of the money supply and redemption value for real assets. Bitcoin meets the requirements as its money supply is not controlled by any central bank, but rather is supported by the users within the system, and its coin supply is a product of market demand (Hankin, 2018). Online retailers have now begun to accept Bitcoin as a method of payment for transactions, with Overstock.com leading the movement. As of May 2017, the site reported that its Bitcoin sales had tripled and they are bringing in $5 million annually from its payment usage. Consumers have an incentive to use cryptocurrency for larger transactions in which money needs to be moved quickly, and they can exploit the lack of taxation present with purchases (Mulqueen, 2018). Use of digital currencies for these deals also allows for the avoidance of unnecessary third-party transaction fees that we must account for when using other payment methods, such as credit cards, to guarantee the validity of the transaction.

1.4 Background on Bitcoin and Double Spend Example

In 2008, a user with the pseudonym of Satoshi Nakomoto released the Bitcoin white paper, outlining the purpose for their creation of the cryptocurrency. He states that the concept of the coin is for "a purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution" (Nakomoto, 2009). Under normal conditions, an exchange of money on the internet between two parties will require the use of a financial institution (FI). For example, if Alice wishes to pay Bob online, she will use her debit or credit card to complete the transaction. An FI will then be responsible for removing money from Alice's account, and crediting the balance to Bob's account.

Nakomoto argues that this process is inefficient in that we depend on these FI's to act as trusted-third parties, which in turn increases transaction costs in the long-run. He defines this weakness as a result of the *trust based model.* Banks and FI's cannot promise totally non-reversible transactions as there will always be a need for mediation in the event of fraud. Thus, we see transactions costs rise as FI's require more information from customers in order to increase levels of trust when completing virtual payments. The trust based model can be ignored when dealing with payments of physical currency (i.e. cash, gold, etc.), as the proof of payment exists to the recipient at the time of transaction (Nakomto, 2009).

The belief is that Bitcoin can alleviate the requirement of a third-party for transactions over a communications channel, such as the internet, on the basis of cryptography rather than trust. Say for instance that Alice and Bob agree to make a transaction online, and the payment will be made via check. Alice only has enough in her bank account for this one order. Bob trusts Alice, and tells her that so long as she tells him the tracking number of the envelope once she

mails it, he will send her goods. Alice follows the orders, and receives confirmation from Bob that her order has shipped. However, Alice also enters into an identical agreement with Charlie, and neither seller knows of the other transaction. Alice now has two checks in the mail, yet only one can be successfully cashed due to her lack of funds. Bob and Charlie will both use their bank as a trusted third-party for this transaction, and the only successful transaction will be for whichever seller goes to the bank first. One seller has now experienced the failure in the trust based model, and will have to now deal with the bank to mediate the situation. Alice has successfully completed *double spending*, and now has her two orders for the cost of one. We will discuss the concept of double spending later on, and how cryptography prevents this problem.

1.5 Cryptography Within Bitcoin Transactions

By using cryptography to prove the order of transactions, sellers like Bob and Charlie would be protected from fraud as the ability to reverse a transaction is lost, and customers could be insured with escrow techniques. This peer-to-peer systems functions without the need for a third-party, and utilizes timestamps to ensure the proper order of transactions and prevent users like Alice from completing double spending (Nakomoto, 2009). Since Bitcoin transactions and the ledger are public, users are assigned a unique string of characters to identify themselves, known as a *bitcoin address*, which ensures anonymity (Böhme, Christin, Edelman, & Moore, 2015). To complete a transaction, users enter the address of the recipient rather than any information which could lead to public identification on the ledger. In order for the transaction to be considered “good”, computers (nodes) on the system will complete complex mathematics problems to ensure the chronology of the ledger is correct, and that the sender is not committing double spending.

Transferring ownership of a coin in the Bitcoin system relies on a compilation of digital signatures from an owner to a payee. In order to create a transaction in this system, the following order must be preserved (Kroll, Davey, & Felten, 2014):

- Create a hash of the owner's last transaction and payee's public key
- The owner must digitally sign this hash using their private key
- Signature is then added to the end of the coin
- Payee can verify that the signature of the now previous owner is true
- Payee now becomes the new owner of the coin

This cycle is then repeated infinitely many times as the coin continues to change ownership within the Bitcoin system. However, according to Nakomoto this sequence in itself does not ensure protection against double spending, as the payee is unable to know if the previous owner of their coin engaged in double spending. To ensure the validity of the transaction, there needs to be a method to check each transaction. The ironic solution first discussed by Nakomoto is the concept of a central "mint", which requires coins to pass through a third-party in the system to provide proof that no double spending occurred. This solution is impractical, as whatever company responsible for running the mint would act as a central bank, which is counterintuitive to Bitcoin's strategy of decentralization (Nakomoto, 2009).

Instead, the proposed method for transaction approval involves posting all of these records to a public ledger or *blockchain*, which allows us to remove the need for a trusted third-party. Using the Bitcoin blockchain in conjunction with timestamping requires the implementation of the *proof-of-work (PoW) protocol* for distributed consensus within the system.

2. Proof-of-Work (PoW) Protocol

2.1 Overview and Definition

The proof-of-work protocol in cryptocurrency requires users to expand a level of energy in order to solve a computational puzzle to extend the blockchain, referred to as *mining*. In

practice, the benefit to using computational puzzles to determine consensus within the peer-to-peer network is two-fold. First, completion of a PoW puzzle requires access to large computational power and a high level of energy usage, which creates the concept of difficulty for users in the system. Power and difficulty of these problems increases as the number of bitcoins in circulation approaches its maximum supply of 21 million coins. As of April 2018, MarketWatch estimates that there are approximately 16.9 million in existence, meaning only 20% remain to be created (Hankin, 2018). Although this makes the PoW protocol more expensive over time, this helps to guarantee that the user, or *miner*, of the block has put forth adequate effort to continue consensus within the blockchain, and allow for honest branches from the original genesis block (Kroll, Davey, & Felten, 2014).

2.2 Hashing to Solve for Zero-Bits Requirement

The second advantage in a PoW system is the ease by which the result can be verified. According to Nakomoto, PoW computations aim to find a value whose resulting hash has a targeted number of "zero bits". Each block receives an incremental *nonce* which is used to test whether or not the resulting hash has the desired number of zero bits. (Nakomoto, 2009) A nonce in the bitcoin system is defined as a 4-byte field that acts as the variable solver within a PoW puzzle. Since the other fields within the PoW puzzle all have a "defined" meaning, we cannot change their values to solve for the correct number of zero bits. The nonce must therefore be changed many times to different values in order to solve the PoW puzzle, requiring exponentially more energy with each additional zero bit required (increase in puzzle difficulty). After the computer proves it has solved the puzzle and expended the required effort, the block is unable to be changed unless a computer is to re-do this process for the given block and each additional block chained thereafter. This is considered an easy verification for hash results on each block,

as success is defined simply by whether the computer's effort leads to a match with the hash of the problem.

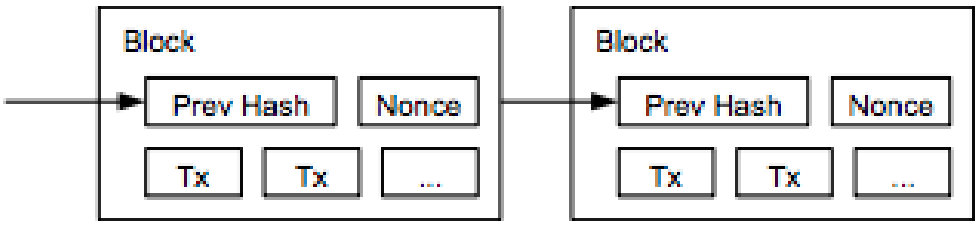

Figure 1: We observe each new block as using the hash of the previous timestamp as a starting point, as well as the variable nonce submitted to test for zero-bits. Tx refers to each transaction filed recorded under each block. This nonce allows us to create timestamping within the blockchain network (courtesy of Nakomoto, 2008).

We can then define that a new block is only considered a beneficial addition to the chain if its resulting hash is less than a target value set within the blockchain itself, and higher hash value results are ignored. If twelve consecutive zero-bits are required but our hash only has eleven, the computer is not determined to have completed the PoW puzzle. Bitcoin assumes in its PoW system that mining will continue on the longest branch of the blockchain (most valid). Miners can be seen as the working arm within the peer-to-peer bitcoin network that allows for transaction approval for participants.

2.3 Computational Energy Needed for Mining

Since there is a cost and energy requirement to mine a block, miners are motivated to expend resources on the most correct (longest) branch. According to MarketWatch, in 2018 the overall cost of bitcoin mining is staggering. The Bitcoin Energy Consumption Index (BECI) estimates that global energy usage in bitcoin mining is equivalent to the power required to sustain the entire country of Denmark with its population of 5.7 million people. Since the

computational power required to mine increases as supply decreases, energy use is expected to rise to equal the uptake of Bangladesh with over 160 million residents (Hankin, 2018).

2.4 Profit Inequality and Arbitrage Opportunity for Miners

Since energy costs and accessibility to renewable power varies greatly among different countries, miner profit is not equal. Miner pay is therefore contingent on a few primary variables: the energy cost required to mine one bitcoin, the trading price of bitcoin, and the incentive for completion of a block. The United States ranks 41st in the world for bitcoin mining costs, averaging $4,758 a coin, and the current bitcoin value is just over $8,000 USD per coin (Hankin, 2018). Thus, miners will only make a profit when the value of bitcoin exceeds the energy consumption costs. New York-based firm Fundstart estimates that on average bitcoin is now trading at its breakeven cost, leaving miners with little to no room for profit. Miners then in this case will no longer be motivated or incentivized by financial benefits to mining, but rather the intrinsic duty of maintaining the validity of the blockchain unless a transaction fee can be assigned for block creation (Hankin, 2018, Nakomoto, 2008).

Energy consumption for mining therefore presents the ability for arbitrage by miners located in different countries. If Alice mines via PoW in Venezuela at an average price of $531/coin and Bob mines in Iceland at $4,746/coin, assuming a base bitcoin value of $8,000 will result in very different profits (Hankin, 2018). Alice will have greater monetary gain by exploiting lower electricity costs in Venezuela, even though her and Bob are both completing similar-level difficulty puzzles for the same end reward. One must obviously consider other transaction costs such as the expense to relocate oneself for mining, however the point remains.

2.5 Incentives for Network Support and Inflation Control

We now discuss how the payoff to miners for their efforts can help to support the network integrity and act as an inflation-proof reward structure after all coins have been issued into circulation. Upon creation of a new block by Alice, she will now own the new coin put into circulation as a result of her computational effort expended. Nakomoto believes that incentivizing PoW miners such as Alice in this manner encourages more users to act as nodes in the system and is a successful method of decentralized coin issuance (Nakomoto, 2009). She equates this to the concept of how gold is able to be brought into circulation-namely of how gold miners must expend time and effort (computational power and electricity) to increase the active supply of gold. Both Bitcoin and gold have a finite supply to ever be issued, and both are dependent on effort as the backbone for cultivation.

The PoW protocol of Bitcoin improves the likelihood of honest activity among nodes as well, even if an adversarial miner was to gain control of the blockchain in a rare situation. While this would require a vast amount of computational power and energy to overhaul all honest nodes in the system, which we will discuss in more detail later, the adversary would be forced to choose how they wish to use their siege. The attacker could attempt to steal back money from transactions she has already committed (i.e. Alice with Bob and Charlie), but she would achieve greater profit by continuing the blockchain and getting rewarded with new coins for each block created. (Nakomoto, 2009). The marginal benefit to furthering consensus on the blockchain therefore is independent of one’s honesty desires-both types of miners are encouraged to continue the blockchain.

Transaction fees can also be incorporated within the PoW mining incentive so that each node in the system can still benefit after all coins have entered circulation. The transaction fee is

calculated as the resulting amount present if the input value is greater than the output value of a transaction. Nakomoto declares that this fee can be added to the reward value of the overall block containing the transaction, and is inflation-proof (Nakomoto, 2009). Since the amount of coins is finite and our initial PoW incentive structure is violated after all 21 million coins are issued, these transaction fees will provide motivation to nodes to remain active (Böhme, Christin, Edelman, & Moore, 2015). Inflation is avoided as the supply will never change once the cap has been hit, so the value of a transaction fee on *d* day (assuming all coins are circulated) will be equal to the transaction fee *n* days after. Maintaining a constant reward structure after all coins are issued then mean adversarial nodes have less profit available to them, should they be able to raise enough energy and power to outdo all honest nodes. 51% attacks (discussed later) and double spending in this case are limited to profit only equal to reversing the money one has sent.

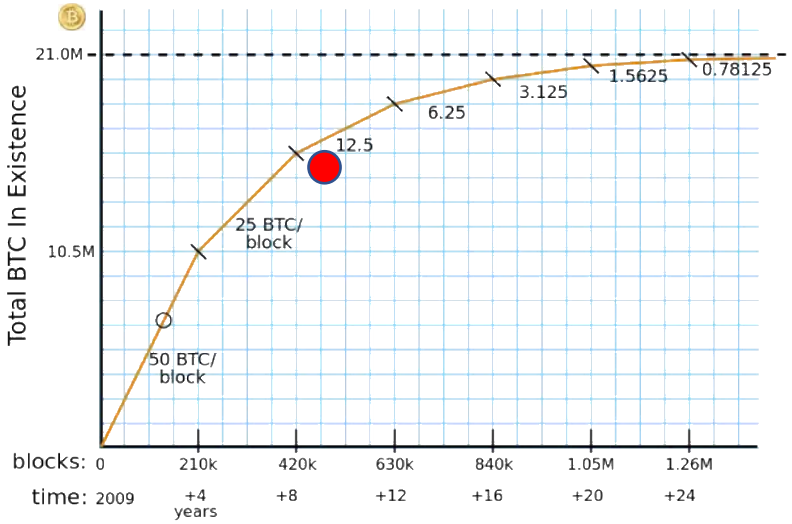

Figure 2: As each increment of 210,000 blocks are created, the reward (issuance) of coin per block decreases for the miner. The limit is defined as the total possible circulation of Bitcoin coins, where a transition to strictly transaction-fee rewards will exist with zero inflation. Note: Given this graph, 2018 circulation falls just below the trend line (courtesy of Quora).

2.6 Likelihood for Adversarial Disruption

While PoW struggles from an energy perspective, it has shown evidence through Bitcoin of the minute likelihood of blockchain disruption by dishonest nodes. Nakomoto defines the competition by a dishonest node to create a differing chain against a series of honest nodes to be equal to a Binomial Random Walk. Should the attacker succeed in generating a new block on their alternate chain, this is deemed a failure under the random walk as the honest node lead is cut down by 1. The same principle exists if we are to consider the successful event to be an honest node placing the next block on the chain, increasing their lead by 1 over the attacking miner (Nakomoto, 2009). The benefit to the PoW system in this case is that should the dishonest attacker succeed in getting the next block on her chain, it does not mean failure for the network as a whole. Assuming the majority of nodes are honest in verifying the correct blockchain, this alternate chain made by the attacker will fail to be verified and agreed upon overall. The attacker is unable to ever get money or coin that never belonged to her, so the only reward for her computational effort would be to retrieve the money she has already spent. This aligns with the incentive structure mentioned previously, as it is more worthwhile for a node to remain honest for their own benefit in order to collect the new coins issued upon block creation and ownership.

Calculating the potential of an attacker to catch up to an honest node to create a branching blockchain for her own benefit and double spend is modeled with the following simplified function and constraints set in place:

$$q_z = \begin{cases} 1, & p \leq q \\ (q/p)^z, & p > q \end{cases}$$

Where:
p is equal to the probability that an honest node gets the following block
q is equal to the probability that an attacker gets the following block
z is equal to the number of blocks behind the attacking is from the honest

If the chance the attacker can reach the next block is greater than or equal to that of an honest node, the ability for the attacker to breakeven with the honest chain is 1. Regardless of whether the attacker can reach this probability over an honest node in the beginning, she will statistically be exponentially unfavored to ever catch up again. The dishonest node would need to have the ability to constantly outwork the honest nodes to have any chance at establishing their alternate chain and completing double spend to scam a recipient. Our second constraint depicts the more common outcome should the probability the honest node wins is greater than that of a dishonest node. Since our exponent is in terms of *z*, as the attacker falls further behind in number of blocks, the difficulty to breakeven with the honest chain increases substantially for each additional block (Nakomoto, 2009).

However, since completing a false transaction is dependent on the dishonest attacker attempting to send and then recall her money via double spend, time plays a factor in the success chance for the scam to occur. We define the process for this falsified action to occur in terms of goals for the sender:

1. Alice would like to buy her goods from Bob, and she agrees to send him Bitcoin as payment, even though she has plans to scam and try to reverse the transaction.

2. Alice plans to send the coin to Bob, and prepare an identical replicate chain in private to send the money back to herself before the transaction is posted to an honest block.

3. If she is able to get far enough ahead in the PoW process as compared to the honest nodes, she can reverse the transaction to return her money in hopes that Bob has already shipped her goods.

4. This is the equivalency of our initial analogy to a bounced check if we were to consider Alice knowing exactly when a bank would attempt to cash her check by knowing where it stood in line for the hypothetical “queue”. She could call and cancel the payment before it can execute as she was able to get ahead of the honest system (the bank ledger), and Bob has already sent the goods.

The attacker can have an advantage should she already know the public key of the receiver for the transaction, as blocks could be prepared in advance before signature occurs. By

having these blocks prepared in a "parallel" blockchain, the attacker could then execute the transaction at the exact moment that she beats out the honest chain. Instead, to increase security for the honest receiver and raise the difficulty level for the attacker (sender) according to Nakomoto's function, the receiver will create a new pair of public and private keys for the transaction. Granting the sender, one's public key right before signing allows the honest nodes to retain the assumed advantage for the next block over the attacker to the network.

Alice, in this case, would execute the transaction to send Bob an amount of Bitcoin, and then have to race to catch up to the honest blockchain to create a different version to the transaction in which her money is returned to her address (Nakomoto, 2009). Bob then will then check to ensure his transaction has been posted to a block and that *z* blocks are chained thereafter by honest nodes. Thus, Alice must then not only replicate the work of the block in which this transaction occurs, but each following block too if she wishes to publish her desired version of the blockchain. If we assume the average target block time in the Bitcoin PoW network to be, say 10 minutes, holds true for this transaction, we can estimate the potential progress of Alice's alternate chain. Many new cryptocurrencies claim shorter *block target times,* but this will still work for our analysis.

> Let's assume it's been 30 minutes, so three blocks have been chained after the block holding Alice and Bob's transaction, so *z* would be equal to 3. In general, Bob cannot know for certain Alice's progress, but can estimate how far behind she is with the following Poisson density equation:
>
> $$\lambda = z\frac{q}{p}$$
>
> For a more simplified version to calculate Alice's current probability of catching up, we calculate the product of the Poisson density for each possible progress made by Alice for *z* blocks by the probability of catching up to that same *z* blocks given Alice's progress of *k*:
>
> $$1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!}\left(1 - (q/p)^{(z-k)}\right)$$

> Beginning with the right-side parentheses, we observe that our exponent reflects the difference between how many blocks behind Alice is and her blocks created thus far in her parallel blockchain. We take *q* divided by *p* raised to the difference between *z* and *k* given the constraint that *k is* less than or equal to *z.* If *k* had managed to be greater than *z,* then we would assume Alice to have a probability to catch up equal to 1 in terms of gaining ground on the honest node.
>
> Subtracting these two values from each other gives us the probability Alice could catch up from the current point. Our fraction measures the Poisson density raised to the power of Alice's block progress *w*, times the mathematical constant *e* raised to the negative Poisson density, all over the factorial of *k*. Summing our results from the assumption Alice has made zero progress to the current breakeven value of *z*, we get our probability that Alice cannot catch up to the chain's current position. Subtracting by one gives us our final answer as to her true probability of breaking even to reverse the transaction on Bob.

Running this equation via script shows favorable results for Bob and other receivers in the Bitcoin PoW network when dealing with senders such as Alice. The following results are those released by Nakomoto in the original publishing of Bitcoin's proposal to the public, and assumes two rates for *q* for the attacker: 10% and 30%. Even at a higher probability of securing the next block, the likelihood in breaking even decreases exponentially as *z* rises. Assuming that Bob obeys by our assumptions to issue a new public key to Alice before the transaction, and that all other energy and block time requirements are kept intact, we find it extremely unlikely that Alice will be able to complete double spend through PoW.

```
q=0.1
z=0     P=1.0000000
z=1     P=0.2045873
z=2     P=0.0509779
z=3     P=0.0131722
z=4     P=0.0034552
z=5     P=0.0009137
z=6     P=0.0002428
z=7     P=0.0000647
z=8     P=0.0000173
z=9     P=0.0000046
z=10    P=0.0000012
```

```
q=0.3
z=0     P=1.0000000
z=5     P=0.1773523
z=10    P=0.0416605
z=15    P=0.0101008
z=20    P=0.0024804
z=25    P=0.0006132
z=30    P=0.0001522
z=35    P=0.0000379
z=40    P=0.0000095
z=45    P=0.0000024
z=50    P=0.0000006
```

Figure 3: Assuming two different values for *q*, Alice still will face poor probability of catching up to the honest blockchain given her current progress in either scenario. PoW under the Bitcoin system provides transaction recipients with confidence that the sender would need extreme

> computational power and energy to call back a payment (courtesy of Nakomoto, 2009).

PoW has shown us that validity in a transaction comes from proof that energy was consumed via computational work to solve a problem in order to create a block, and provides sound protection against double spending by senders. Nodes show their allegiance and support to the decentralized system by using energy in order to participate in the incentives issued to miners of new blocks, including small transaction fees paid out once the limit of coins are in circulation. Nodes are therefore motivated to remain honest due to the investment set forth to complete these proofs, and how the system is designed to punish those who attempt to outwork honest miners for double spending, such as Alice.

However, the energy requirement for PoW support in a standalone network such as Bitcoin is not feasible for long-term usage once the cost to mine a coin outweighs the marginal benefit (value of the coin), as we have already seen in certain areas of the world. In an attempt to mitigate this concern and lower energy for distributed consensus in decentralized currencies, we must consider the viability of Proof-of-Stake (PoS) protocols found in newer generation cryptocurrencies, such as our case study of Cardano (ADA).

3. Proof-of-Stake (PoS) Protocol

3.1 Overview and Definition

The concept behind PoS was originally proposed on a message board online in late 2011, and more formalized in a paper outlining a new cryptocurrency, “PPCoin” or Peercoin, in 2012 (“The History and Evolution”, 2018). Sunny King and Scott Nadal, the founders of the idea, express that PoS is inspired as a hybrid conjunction with Nakomoto’s PoW design in Bitcoin, but with a focus on reducing overall energy costs when verifying transactions (King

& Nadal, 2012). Instead, Peercoin was built on the premise that *coin age* can be the basis of the PoS design and improve security of the network overall. We define coin age as the amount of currency (number of coins) times holding period value (i.e. 30 coins held for 100 days returns a coin age of 3,000 coin-days). Once the coins have been used, the system declares that the "accumulated coin-age has been consumed" (King & Nadal, 2012). To validate our calculation of coin-age, a timestamp is attached to each transaction within the protocol. Consider the example of Alice, Bob and Charlie once again:

1. Alice would like to buy her goods from Bob, and she agrees to engage in a PoS based transaction such as Peercoin for her payment method. Since we define coin age as equal to the product of *n* days times *c* amount of coins, let us determine her total coin age before this transaction is 5 coins held for 10 days. Alice has a coin age of 500. For the sake of simplicity, let's assume her total Peercoin value equals her obligation to Bob.

2. Alice plans to send her 5 coins to Bob to pay for her goods, but also wishes to complete a double-spend and replicate the same transaction with Charlie. Consider that neither Bob nor Charlie knows of the other transaction.

3. Alice initiates her transaction to pay Bob 5 coins, which creates a timestamp for the blockchain that will determine the validity of her transaction. Her coin age of 500 is now "consumed" and lost, therefore showing her ownership has been transferred. Any additional attempt by Alice to send these same coins to Charlie should be disallowed by the PoS protocol by observing the timestamp of her first transaction.

4. Alice is unable to complete double spend, assuming honest activity among the nodes within the blockchain. Since coin age is dependent upon accumulation of a certain holding period of *d* days, this makes the coin difficult to counterfeit.

5. Bob receives the payment and begins to accumulate his own coin age on the coins, to which he can then transfer to another party should he decide. He can feel confident in this transaction and send Alice her goods, as her loss of stake from this transaction shows her original ownership of the currency. Charlie will not receive a payment from Alice and will therefore refuse to send her the goods.

Since PoS protocol favors using one's overall holdings or "stake" instead of energy consumption as an indication that effort was used to verify a transaction, coin-age can help to provide this proof of ownership. King and Nadal argue that since coin-age is set up to measure based on the time an individual has held the coin since the last transaction, it is difficult to

falsify ownership. If a coin is already spent in another transaction, its coin-age has been reset to zero, therefore not allowing it to be spent again as it is already "consumed". Should an individual be able to provide proof-of-stake with their coin-age being consumed in a transaction, then we can assume this to also be proof of ownership in the currency. This acts as a barrier to counterfeiting much like the security aspect of PoW, but with far less energy required because it removes much of the steps involved in solving puzzles, which we will discuss below (King & Nadal, 2012).

Block generation under PoS improves from PoW as it does not involve the infinite trial and error of an incremental nonce in order to test if the resulting hash meets the target. As of late 2017, it is estimated that in order to generate one new block on the Bitcoin network, a node must complete 2^{60} hashes, requiring high amount of energy (Kiayias, Russell, David, & Oliynykov, 2017). Rather, the block creation by Peercoin involves a combination of two types of inputs, *kernel* and *stake* in an effort to minimize energy usage compared to PoW, and will provide a single stake output. First, the block owner will consume their own coin-age by completing a payment to oneself, and the reward will be the ability to mint the resulting block after proving effort via stake. The need for the kernel input is evolved from PoW to allow for PoS block generating to be a process that remains random over time, and is used to help meet the hash target set within the protocol (King & Nadal, 2012).

Hash target for the kernel in PoS is therefore a target determined by the coin day value consumed by the stakeholder, which differs from PoW as the target hash is set equal for each node on the network. Simply put, this means nodes that have greater coin-age (stake) will face less difficulty in meeting the hash target, and the uniqueness of one's target relative to coin-age preserves energy. If I have a coin-age equal to half of Charlie's, then we can expect

Charlie to generate a kernel to meet the hash target in half the time it takes me. This also provides incentive for users to accumulate greater stake in the currency to improve their chance of securing a block for minting. Say Alice, Bob and Charlie are competing for the opportunity to mine a new block under Peercoin PoS:

1. Alice, Bob and Charlie are among many nodes within the Peercoin system who compete to mint a new block for the blockchain. We define their names respectively as *A, B, C* with some coin age *c* for each, creating A_c, B_c, C_c. Let us assume their coin ages are related in such a manner that:

$$A_c = 2B_c = 4C_c$$

2. Our above equation tells us that Alice's coin age is equal to double that of Bob, and quadruple that of Charlie. We will use each person's stake shortly in order to determine their chance of gaining block generation rights.

3. Each node (Alice, Bob, Charlie) will complete a payment to themselves to consume their coin age, therefore proving their stake in the coin at some time *t*. Depending on their coin ages, each node will have a different hash target to meet. This hash target is based off Nakomoto's work in PoW. However, since Alice has a higher coin age than both Bob and Charlie, her chances to mint the next block will be greater than both of them.

4. Peercoin PoS sets the difficulty of the hash target relative to one's stake. If Alice can prove via a self-consumption of coin age that her age is greater than Bob, her time to reach the hash target should be proportionally faster to the relationship between their stakes. To reach their hash targets, each will test various *kernels*, and the first to do so successfully will be chosen to generate the next block.

5. In our case, Alice has the greatest proportional stake (coin age), and should therefore meet her hash target first. She should finish twice as fast as Bob, and four times as fast as Charlie. Users under this PoS therefore are more successful as nodes as their overall coin age (proportional stake) increases.

Under PoW, we acknowledged that as more coins are brought into Bitcoin circulation, the block reward is decreased in half each 210,000 blocks before reaching the coin supply limit for the currency. When we consider the minimal energy requirement for PoS, Nakomoto's miner reward proves to be an inefficient incentive for long-term decentralized support of the PoW network. While PoW nodes are important for coin minting, they are more critical to verification of transactions in the Bitcoin network. King and Nadal state that as the

marginal reward for coin minting decreases when circulation begins to reach its limit, there is less incentive to mine blocks in a PoW framework (King & Nadal, 2012).

This statement agrees with our earlier argument that miners will only be motivated to expend energy on the system if the cost to do so remains less than the value of the coin. Once circulation limit is reached, this would mean PoW transaction fees need to be increased drastically to support energy requirements for transaction verification, considering the difficulty of PoW problems increase over time. This then transitions the burden of the fee to payees such as Alice and Bob, who may then resort to other payment options with lower fees. Should honest miners drop off in a PoW system at this point and total energy input fall, a security vulnerability arises as verification is tied to miner activity, and provides attackers better chances to claim majority control of the energy in the network.

Incentive structure for minting of new blocks in this PoS protocol is based on a fixed rate reward multiplied by the consumed coin age of the transaction, independent of the number of blocks created. For Peercoin, the outline was to set this rate to one cent per coin-year, as using this low amount as a prevention against high inflation rate going forward (King & Nadal, 2012). Nakomoto also accounted for inflation when determining long-term incentive for Bitcoin after circulation limit has been achieved, however this structure is better suited in the PoS network. Stake nodes will be motivated to continue holding ownership in the coin over the long run if they know their reward is dependent on their overall coin age and not with energy consumption. Continued node activity will be more feasible under PoS because so long as there are individuals willing to participate in the protocol, transactions can be verified for the public ledger, regardless of world electricity or technological infrastructure costs. Over the years, others have begun to note the practicality in adopting and evolving the objective of

distributed consensus via PoS, and we will now examine how Cardano (ADA) by Input Output Hong Kong (IOHK) utilizes their own PoS algorithm in practice.

3.2 Cardano ADA Ouroboros Praos Case Study

Cardano (ADA) is a third-generation cryptocurrency designed by previous Ethereum developer Charles Hoskinson and is currently in circulation on major exchanges such as Bittrex and Binance. As stated, their blockchain protocol is referred to as *Ouroboros Praos*, and is defined to be "a provably secure proof-of-stake system…to the best of [our] knowledge, this is the first blockchain protocol of its kind with a rigorous security analysis" (Kiayias, Russell, David, & Oliynykov, 2017). First, we will discuss the two main goals of the Ouroboros PoS algorithm with regards to transaction verification and a truthful ledger: *persistence* and *liveliness.*

3.2a Persistence and Liveliness as Measures of Stability

Persistence can be observed as the theory that once one node decides that a transaction is "stable", all other honest nodes within the blockchain should follow suit. Here, stability is reflected by a parameter value *k*, referring to a chain of however many blocks following the event agreed the transaction as honest. If a node is to be queried and report the transaction to be in a different position within the ledger or be conflicted by a separate transaction, this is where persistence would fail (Kiayias, Russell, David, & Oliynykov, 2017). Once the transaction has been considered stable by many nodes for a particular number of *k* blocks, stability is assumed. Bitcoin believes its transactions to be true following a similar general idea, being that the longest blockchain in existence should be treated as the best, using the number of blocks as a determinant (Nakomoto, 2009). Ouroboros supports the consideration

of longest chain as well for validity of the network, with some adaption, given that it will reflect growth from honest nodes.

Liveliness in Ouroboros is dependent on time as its measure, calculating how many certain periods have passed that the transaction has been publicly available to the nodes of the blockchain. Cardano considers the threshold amount of time required for stability to be equal to the *transaction confirmation time*, or a certain amount of *u* periods. Combining these two measures for transaction history implies confidence that honest nodes have majority control of the network and that double-spending was not attempted in a transaction (Kiayias, Russell, David, & Oliynykov, 2017).

3.2b Multiparty Coin-Flipping Entropy for Block Generation

Compared to Peercoin, the design of the Ouroboros algorithm improves the usage of stake to prove transactions within the blockchain. No longer is any computational PoW used to determine effort has been expended (kernel). Within this standalone PoS, nodes are now chosen, or rather *elected*, to participate in the communal consensus process by proportion of their stake, and randomness determines who becomes a *slot leader* to have the responsibility to create the new block. These nodes are then assigned to given slots which depend on a central timestamp, where they must be synced to the system in order to produce the next block. One of the major described challenges to creating a secure PoS system, according to the Ouroboros documents, is creating the blockchain to allow fair entropy for changing stakeholders over time. Since one's holdings in the currency is not static, there must be a method to introduce randomness from within the algorithm itself to assign the next block generator, rather than using randomness based on beginning stakes (Kiayias, Russell, David, & Oliynykov, 2017).

Designing this randomness for the election process, however, is susceptible to adversary computational attack if an attacker oversees multiple nodes with varying stakes, as one user can control multiple stakes, called *grinding*. Although the probability is very low, one could argue that theoretically a series of colluding adversaries could be elected to produce a series of blocks in a row under the random entropy. This would lead to an incorrect ledger and a violation of the protocol. Ouroboros addresses this vulnerability using its *coin flipping* selection to encourage entropy, following these rules:

- Coin flipping will be a multiparty venture during each *epoch*
- An epoch is defined as a regular interval of time within the blockchain when current stakes are determined
- During each epoch, this multiparty random event will be communicated to the blockchain as a whole
- The random stakeholders in this group each epoch will carry out the random coin flipping computation to determine the next set of stakeholders in the following epoch, as well as the slot leaders for the epoch
- Grinding is prevented as entropy is used as opposed to computational power to sway results

To demonstrate the election process under Ouroboros, let us once again assume that Alice, Bob and Charlie would like to act as nodes within the blockchain. Since we are dealing with PoS like Peercoin, their participation is assumed to be independent of energy costs or other computational power concerns. The election process goes as follows:

1. Alice, Bob and Charlie are among many nodes within the Ouroboros system who compete to mint a new block for the blockchain. We define their names respectively as *A, B, C* with some proportional stake *c* for each, creating A_c, B_c, C_c for some epoch number u_i with *s* many slots.

2. Because stake under this protocol is dynamic (noted later), one's stake is taken during regular periods to determine their ability to be elected to a slot position in the next epoch based on first *genesis* block each epoch. Let there be many slots in each epoch, all with short time frames less than 30 seconds each.

3. The previous shareholders elected to participate in u_{i-1} carry out the coin flipping experiment and post their results to the blockchain along with their endorsed slots. The coin flippings will determine a value equal to random shareholders that will allow each to be elected as a *slot leader* or *input endorser*.

4. If the value A_c for Alice is 3%, B_c for Bob is 2%, and C_c for Charlie is 0.5%, we note the following. Alice will have the best chance of selection under random entropy as her stake is greater than Bob. Charlie will not be elected for any position as his stake is below the threshold required to participate, which is set at 1%.

5. We acknowledge that a user with some stake above the threshold amount is able to participate in the protocol as long as they abide by requirements mentioned later. Alice and others with greater stake have better chance of being selected to participate. If Alice is chosen, she will gain rights to either endorse inputs to each block in the epoch u_i, or will be allowed to sign the slot to the blockchain (see below for job roles).

3.2c Role Functions of Protocol Participants

The outcome of multiparty coin flipping within Ouroborous assigns multiple roles to nodes for slot responsibilities, where transaction endorsement is required to allow for block generation. We define the major roles in each slot as follows:

- **Slot Leader:** This elected node is granted the right to generate the block during a given slot. Her probability of being selected for this position is equivalent to her proportion of stake at the time of the first (genesis) block within the epoch.
- **Input Endorsers:** These individuals within a slot are elected in the same way as Slot Leaders, however their role is to endorse the transactions being posted to the new block. Blocks are only considered reputable or good if they are endorsed by an input endorser.

Slot leaders under the Ouroboros algorithm therefore are granted the comparable duty to a classic Bitcoin miner, without the race to complete the block against other nodes. Their proof of effort in generating the new block for the chain is depicted by their proportional stake calculated at the genesis block of the current epoch. If they had insufficient stake to compete in the protocol (assumed at 1%) or zero stake at all, they would not be allowed to function as a node. Only allowing stakeholders in the currency to submit their block, per approval from input endorsers, encourages honesty among slot leaders to protect the validity of the blockchain (Kiayias, Russell, David, & Oliynykov, 2017). Because probability of role assignment remains a function of proportional stake, we consider random selection to be representative of the entire network. A user with significant stake only hurts themselves if

they wish to act dishonest by participating. Acting in a way that compromises the blockchain negatively affects the slot leader as they already have investment in the currency itself, mitigating some issues with *nothing at stake* concerns within PoS infrastructure. Let us assume Alice has been chosen to act as a slot leader for some slot in a given epoch, and Bob and Charlie have been elected to be input endorsers for her slot:

1. Bob has used his stake in order to become eligible to participate in the Ouroboros protocol and is responsible for endorsing transaction to a block for publication to the blockchain. Assuming there is no indication of double spending in this transaction, Bob will approve of this as an input to the slot. Under the values of persistence, assuming Bob is honest, Charlie should approve of the same transaction. The longer the transaction has been broadcasted to the blockchain as well, liveliness should infer the transaction is good as well.

2. Once Bob and Charlie, along with the other input endorsers in the slot, have endorsed the inputs for final signing by the slot leader Alice, she will approve of the block and broadcast her decision to the blockchain. Since Alice is elected to participate based upon her stake within the currency, she is encouraged to act rational and honest to protect the value of her investment. Failing to approve of a block for her own motivations only prohibits an honest blockchain, and dishonesty can lead to a drop-in price of the coin.

3. Alice, Bob and Charlie will complete the coin flipping process in order to determine the next shareholders in for the slots in the following epoch. Their stakes will once again be considered simply out of respect for the random entropy protected under Ouroboros. If Alice wanted to act as a dishonest node, the process hinders her ability to manipulate the system to choose herself for consecutive blocks as stakes are not static and calculated often.

Input endorsers in Ouroboros are similar to the philosophy of checks-and-balances within a decision-making community. Since stakeholders chosen to participate in each slot communicate their messages to the blockchain, this improves protection against selfish slot leaders attempting to generate a dishonest block. Should an input endorser find a transaction within the questioned block is attempting to complete double-spend and is unfaithful, the slot leader is unable to generate the block (Kiayias, Russell, David, & Oliynykov, 2017). In PoW, we fail to see the same supervision as its block generation ignores communication between

nodes in this manner. Next, we will discuss the requirements of stakeholders to be eligible for selection to participate, as well as how incentive structure is designed.

3.2e Stakeholder Requirements for Incentive Distribution

By removing the need for large-scale investment in computational resources, PoS via Ouroborous has less barrier to entry for users within the system to participate as nodes, so long as they are present under certain conditions. Cardano believes that the major incentives under PoS should be for *availability* and *transaction verification*. Participation is indeed independent of one's overall investment in computational resources compared to PoW, but for stakeholders to gain incentives for block generation, we must observe these assumptions:

- One slot before her turn, an elected shareholder will sync and query the current longest blockchain and any endorsed inputs (transactions) to be included in the block during her slot.
- She will remain available and online during her elected slot in order to generate the new block.
- In a slot if:
    - The slot falls during the commit stage of the epoch and she is elected to issue the VSS commitment (discussed below)
    - The slot falls during the reveal stage of the epoch and she is elected to issue the required opening shares and opening to her commitment (discussed below)
    - She will frequently check to see if she is elected for participation in the current or next epoch.
    - Be available during her slot assignment of input endorser to process and verify transactions to be input to the block.

Observing these standards for the Ouroboros protocol grant nodes to receive individual and pooled rewards from the block generation during which they were on the committee (Kiayias, Russell, David, & Oliynykov, 2017). Incentives under the Ouroboros PoS are proposed under different options, varying from similar Bitcoin mining rewards to a newer concept of communal distribution of rewards for all stakeholders involved (slot leaders and input endorsers). While PoW issues coins and transaction fees for a specific new block to the node that generated the new block, Cardano has the option to distribute rewards based on

individual blocks or during multiple blocks in an epoch. The various reward structures for Cardano are offered as follows:

- Total transaction fees collected from users within a block are issued to the slot leader who issued the block (similar to Bitcoin).
- Total transaction fees collected during a sequence of blocks in an epoch can be pooled together and distributed to all shareholders who were active and participated in these slots (restaurant splitting tips between servers and busboys at the end of multiple shifts).
- Reward shareholders elected to be committee members for random coin-flipping (tipping for creating entropy).
- Reward input endorsers proportional to however many inputs they managed to endorse (commission based concept).

Users under PoS in varying roles to support the network can earn either active or passive incentives so long as they remain synced to the blockchain when required. It is not necessary to remain active on the system at all times, therefore reducing the already minimal amount of energy required to support PoS compared to PoW nodes. We can now identify the cryptographic process by which Ouroboros proves to verify transactions to be input to blocks.

3.2d Commitment and Opening Cryptography for Transaction Approval

The cryptographic verification process for transactions within Ouroborous can be dissected to the phases *commitment, reveal* and *recovery* which occur during each epoch. Communication to the blockchain network as a whole via the PoS involved in Ouroborous allows for the commitment phase to begin. Electors for each slot (i.e. the slot leaders), will generate a random value via coin-tossing that is kept secret from others initially on the network. This will be known as an elector's "commitment", and it contains encrypted transactions from the input endorsers and a "proof of secret".

Similar to PoW cryptography, the elector will now sign this commitment with her private key, however the commitment will have her public key attached (Rosic, 2018). The benefit in

this case is that the public blockchain has evidence of who submitted the commitment (albeit anonymous assuming public keys are still unrelated to one's information), and that the posting will also show which epoch it belongs to. Publicly announcing this information provides transparency to users and improves overall verification honesty using multiparty communication as slot leaders collect other commitments to post along with their blocks to the network. We will use Alice and Bob to understand the commitment phase:

1. Alice will generate her random value via coin-tossing to keep in private from other nodes within the network initially. She will package this all into her own "treasure chest", with the locked inside being all encrypted transactions to be posted to her slot and a "proof of secret". This will be known as Alice's Commitment, and is broadcasted to the network with her private key as signature to provide proof that she as a slot leader indeed has stake. The blockchain however will see her posted commitment with her public key and an indication of the epoch she is committing to, which helps to create the reinforce the time aspect of the blockchain.

2. Bob is another slot leader, and will complete the same treasure box process as Alice and will eventually post his commitment to the blockchain as well. Alice and Bob are participating in multiparty communication as their block endorsements are being posted to the public blockchain for verification.

The reveal stage of Ouroboros deals with elected node participants carrying out their responsibility to solve a cryptographic message so that the block should indeed be posted. While this mirrors the idea of hashing within PoW, energy consumption is not a concern as the result is quite binary. Electors are required during this phase to "reveal" an opening phrase to the receivers of their original commitment string (Kiayias, Russell, David, & Oliynykov, 2017). The inherent assumption is that the elector is being honest to the receiver that her opening value will be match the commitment phase message in order to "unlock" the encrypted secret message. So long as these two values match, the secret value from the commitment phase for each elector are presented as a randomly generated byte seed (Rosic, 2017). Now that the blocks have been proven to be valid and post-worthy, we must elect new slot leaders for the following epoch, allowing for the high level of internally generated entropy critical to Ouroborous. This seed is

used as the randomness factor within the coin-flipping protocol, and the system returns to the beginning to reassess dynamic stake proportions.

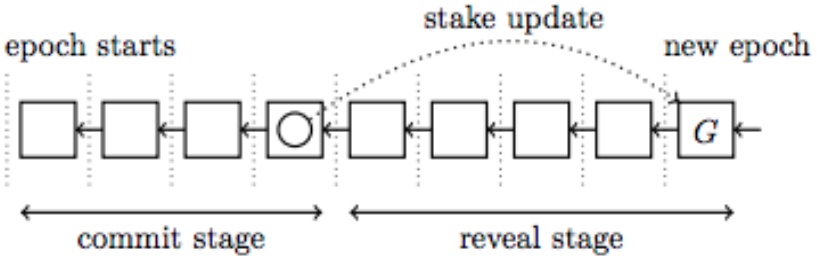

Figure 4: The two main stages of Ouroborous protocol for solving for the cryptographic seed. We note that in order to preserve dynamic stake, a new snapshot of holdings by stakeholders is created to be used along with the secret seed to determine the next slot leaders in the following epoch. The *G* block is defined as the genesis block of the next epoch, where the process will begin once again and the first completed epoch will be posted to the blockchain with all approved slots (courtesy of Kiayias, Russell, David, & Oliynykov, 2017).

Assume that Alice in this case, however, decides to be an honest node while Bob does not wish to reveal his opening phase and allow his block to post. The resulting situation is as follows:

1. For her reveal, Alice will post her secret golden key to open her treasure chest, known as the opening phrase, to the network. Alice is therefore promising the network that she is being honest and that her golden key is the correct key for her specific chest in order to get the secret inside.

2. Should Alice be telling the truth, her block is considered to be trusted and worthy to be posted to the blockchain. Her and the other electors must now complete the coin flipping protocol to determine the next shareholder participants. The proof of secret found inside each treasure chest is used as a random byte string to help initiate entropy for selections in the following epoch.

3. Bob refuses to give up his golden key, but forgets the system is designed to unlock his treasure chest from the inside if nodes cannot have access to the key. This process, explained below, allows other honest nodes to post the encrypted shares included in Bob's commitment phase, and slot leaders can back-solve to create a replica key to open Bob's treasure chest. This allows the honest blockchain to be continued even in the hands of a dishonest slot leader selection.

However, much like the concern within PoW, let us assume a certain level of dishonest nodes are participating in the commitment and opening phase, but refuse to post their opening phrase. Essentially, other slot leaders attempting to access the secret in her signed commitment would be unable to "unlock" its contents assuming there is no back-end method to check. To solve this, Cardano implements the recovery phase which enforces the value of multiparty communication via a process called Verifiable Secret Sharing (VSS) (Kiayias, Russell, David, & Oliynykov, 2017). The goal of this section of the protocol is to allow for honest nodes to recover the secret message should the adversary not wish to post their opening. This is achieved by honest nodes posting the encrypted shares included in the commitment phase and back-solving to find the secret phrase. This prohibits the system from falling victim to a malicious slot leader attempting to corrupt the verification process (Kiayias, Russell, David, & Oliynykov, 2017).

3.2f Defense Against Blockchain Attacks

We have significantly discussed the security surrounding the Ouroborous algorithm to protect randomness within the network and its election procedure for consensus. Double spending is limited due to the presence of multiparty communication within the protocol and by devoting the block generation process to proportionality of stake rather than energy mining capabilities. Persistence under Ouroboros argues that once an honest node has verified the transaction as good to post to the block, no other honest node should disagree. Double spending is therefore challenged as the attacker cannot attempt to convince all the nodes to invalidate the transaction to recall her value sent (Kiayias, Russell, David, & Oliynykov, 2017).

A major consideration in PoS based protocols is to address and eliminate the problem caused by nothing-at-stake users who attempt to act maliciously within the system. We define nothing-at-stake as when dishonest shareholders of the system attempt to create different

falsified copies of the blockchain, given the inexpensiveness of computational effort needed to run a PoS blockchain (Kiayias, Russell, David, & Oliynykov, 2017). In PoW, we see that two shareholders could possibly complete these alternative false blockchains for submission if they can secure computational power to mint the next *n* blocks.

However, Ouroboros claims their system prohibits this forking ability for hour main reasons: amount of time honest nodes are online, Cardano chain selection rule, random entropy of the system for subsequent elections, and a mitigation of the "tragedy of the commons" issue faced in other PoS systems (Kiayias, Russell, David, & Oliynykov, 2017). Since nodes are assumed to be online frequently, they will be synced with the most correct blockchain and these forks will be avoided. Should they be signing on after an absence, users are also instructed to ignore significant forks that may have occurred since their last visit. While the adversary may have been able to create the blockchain very quickly and lengthy, honest nodes will follow the longest, correct chain in order to have the ability to mint a new block and gain rewards (Kiayias, Russell, David, & Oliynykov, 2017). This is supported by the entropy of coin flipping which determines the next shareholder elections as attacking nodes have limited ability to predict when a given honest node will be assigned to minting the next block, lessening the ability to carry out a nothing-at-stake attack.

Finally, the "tragedy of the commons" refers to the concept that under other PoS based currencies, users may submit themselves to the adversary under fear that they themselves will prosper more from joining than from remaining honest. They feel that should they not join the attack, they would be presented with a lose-lose scenario in terms of finances invested in the system regardless of their allegiance in honesty to the system (Kiayias, Russell, David, & Oliynykov, 2017). Yet, under Ouroboros, even if the slot leader or input endorser could act in a

dishonest manner, they have no reason to as the reward for completing the correct blockchain will most likely be greater than the bribe given to work on the competing chain under nothing-at-stake. Since users invest equity into the currency in order to improve their proportion of stake for election, there is minimal reason to act maliciously, as it could hurt one's holdings should an attack impact the market.

Because it is much more difficult and expensive for a dishonest node to control the majority of stake in Cardano than it is to obtain high computational power to solve PoW puzzles, we consider the ability for a 51% attack under PoS (Kiayias, Russell, David, & Oliynykov, 2017). For a PoW based cryptocurrency, a 51% attack would be achieved under the condition that dishonest miners are able to collude and control 51% of the total mining power of the network. This would give an advantage to attacking nodes to generate progressive blocks and achieve double-spend. A PoS based 51% attack would require adversarial nodes to control the majority stake in Cardano, which is considered to be highly unlikely due to cost. However, in the rare situation this is achieved, a dishonest group of miners could create a *fork* of the blockchain, which is one that benefits themselves rather than the network. Assuming honest nodes ignore these forks, persistence and liveliness can be regained (Kiayias, Russell, David, & Oliynykov, 2017).

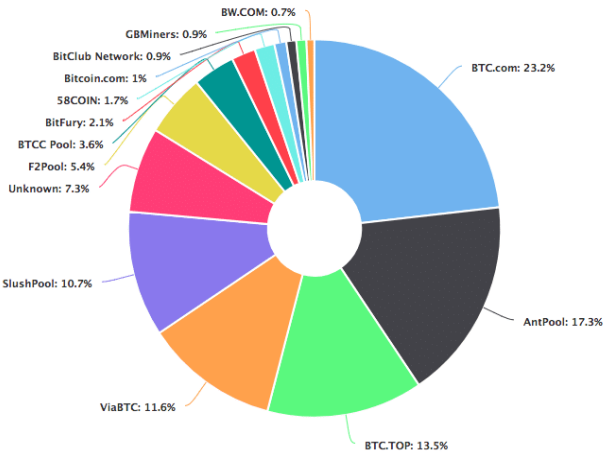

Figure 5: Breakdown of the largest mining pools found under Bitcoin PoW protocol based on computational power. Theoretically, should the top 3 pools decide to become dishonest adversaries, they can achieve a 51% attack on the system. Validity of transactions and security against double spending are therefore violated if this occurs (courtesy of Rosic, 2017)

Ouroboros also allows for increased protection from the idea of bribery attacks issued by dishonest miners wishing to coerce honest nodes to acting against protocol standards and creating false chains. The integrity of the system effectively limits the need for involvement in these actions as compared to Bitcoin and PoW due to staking requirements for Cardano block generation participation. PoW miners may engage in these bribes for two reasons: no stake is required to mine in PoW, and the difference between the bribery amount and block reward (Kiayias, Russell, David, & Oliynykov, 2017). Especially as more Bitcoin enters circulation, a miner with no financial risk in the currency itself will opt to take a bribery reward to compromise the system so long as it exceeds the present value of a PoW block reward and/or transaction fees. While unethical, it makes sense for the miner from a financial standpoint should that be their main motivation.

PoS mitigates bribery attacks by forcing protocol participants to use their stake as collateral in order to participate in incentives and verify the blockchain. Dishonest slot leaders suffer from not acting according to protocol standards as they are unable to make as much profit should their block fail to post. Another angle to this attack would be considering the manner in which cryptocurrency markets are set with volatility. Should it become public that there is significant dishonest activity on the PoS blockchain, the price of the currency will fall and dishonest nodes will lose value/equity in their holdings as a result of their actions (Kiayias, Russell, David, & Oliynykov, 2017). Thus, bribery attacks present a lose-lose under Cardano unless the bribe offered is enough to outweigh the sum of one’s entire holdings, potential loss spread, and plausible earnings from honest block generation incentives.

3.2g Limitations and Considerations

The initial work and creativity of King and Nadal with Peercoin paved the way for blockchain protocol efforts to attempt to solve the energy consumption and related issues with PoW. By involving one's stake as proof to measure the validity of a block rather than computational power, energy is conserved and participants are encouraged to remain honest for the sake of their invested equity in the coin. Yet, due to the immaturity of PoS algorithm-based currencies within the market today, there still are notable limitations and vulnerabilities to consider.

A main concern among PoS critics is to understand the consequences should the assumption of cost to control 51% of the stake is failed from our prior discussion. We assume that the price of Cardano and its overall coin circulation will create a cost to own the majority of stake as greater than the energy cost to control 51% of computational power in PoW. PoS skeptics believe it is more expensive to attack a PoW system such as Bitcoin as dishonest players are unable to enter due to the cost of mining that significant amount of energy being greater than the amount possible to steal (Rosic, 2017). If this is true and the value of a Cardano coin falls to be less than this amount, theoretically an attacker may target a PoS system instead.

Much like how the PoW blockchain is dependent on the miners in order to generate new blocks for transactions to be posted, Cardano requires that users should frequently query the current blockchain and remain online to participate. This also means that of those online for PoS under Cardano, the majority is assumed to be honest players compared to adversarial stakeholders. Major violation of either of these values could threaten persistence within Ouroboros if nodes are not active enough or willing to put their stake in to participate. Some users also are omitted or barred from becoming elected should they not meet the minimum

amount of stake in Cardano (1% according to Cardano) (Kiayias, Russell, David, & Oliynykov, 2017). Honest users may be left out and substituted for dishonest nodes with more investment (stake), or an inability to understand their role as an input endorser or slot leader.

3.2h Transactions per Second (TPS) for Bitcoin vs. Cardano

When analyzing the usability of decentralized currency and removing the need for third-party intervention for transactions, overall transaction approval time is a major concern. Recently, increased activity on the Bitcoin market has bogged down transaction time for users as compared to speeds recorded by other major cryptocurrencies, and information provided below from Blockchain Luxembourg shows the moving average of Transactions per Second (TPS) over the last two years (Blockchain.info). The steady TPS for Bitcoin of less than 10 (which is generous) over the last two years exhibits the concern for long-term usage as other protocols can create products with faster TPS. Bitcoin is stunted in its evolution as it is not managed by any developer or group, but rather is standalone and static. In the last month, we observe that TPs has dropped to below 5 on average. It is worth noting that Bitcoin, when compared against Cardano TPS data, is not totally balanced. Bitcoin functions on a much larger scale than Cardano, and Ouroboros was tested on a limited node amount via the Amazon Web Server cloud.

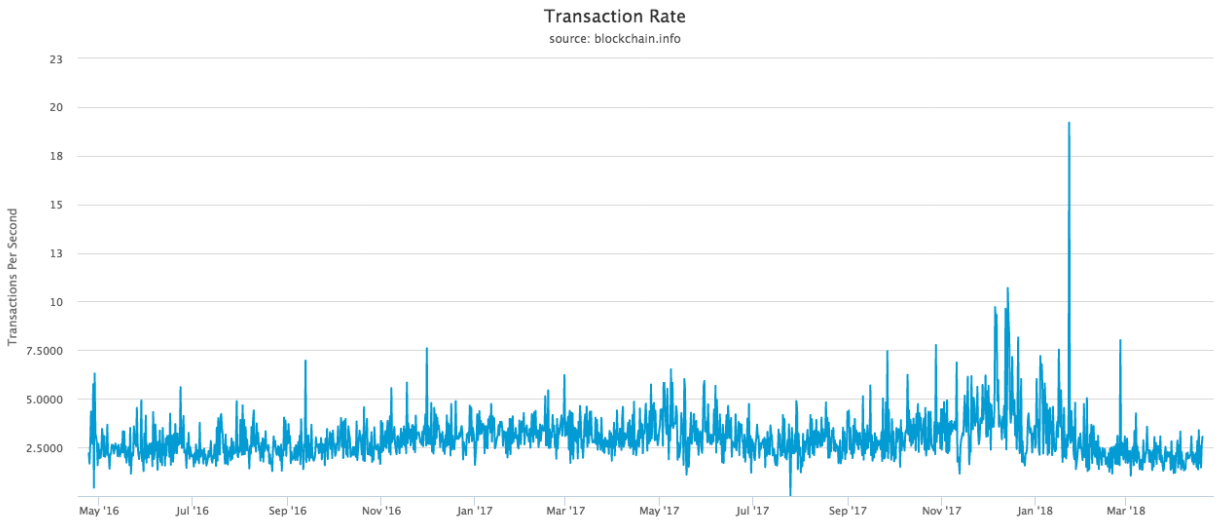

Figure 6: TPS data for Bitcoin ranging from May 2016 to present. Average performance is observed to be limited to under 7-10 TPS. Extrapolations could be during periods of increased node activity/oversaturation of transactions awaiting confirmation to a block (courtesy of Blockchain.info).

Immaturity of the Ouroboros protocol within Cardano speed tests remains to be a considered liability when making overall assumptions about the efficiency of the network. As mentioned, Ouroboros speed was experimented using varying node activity on the Amazon Web Services EC2 Cloud server. Node presence ranged from 10 to 40 throughout these tests, as well as altering slot lengths between 5 and 20 seconds (Kiayias, Russell, David, & Oliynykov, 2017). The results show a much higher TPS recorded for Cardano on its small-scale deployment. Median TPS in the below graph was calculated to be 257.6 for a 40-node test given slot length of 5 seconds. We must wait to compare TPS with fairness between Cardano and Bitcoin until Ouroborous implementation has existed for longer periods of time.

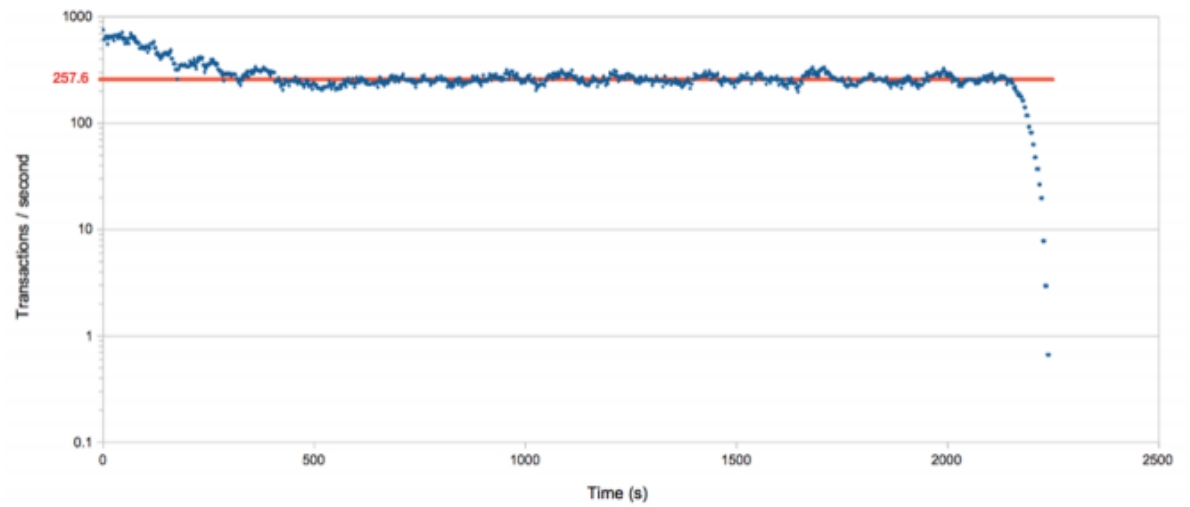

Figure 7: TPS data for Ouroboros in limited field test. Median performance is observed to be limited to under 257.6 TPS assuming 40 nodes utilizing 5 second slot times during each epoch. Node amount limited to EC2 server constraints (courtesy of Kiayias, Russell, David, & Oliynykov, 2017).

Conclusions

After analyzing the evolution of Nakomoto's Bitcoin PoW protocol and cryptocurrency efforts in relation to the progress made in PoS protocol, we have identified savings in stake-based distributed consensus. These benefits are measured by the independence of computational power from transactional validity and removing the need for blockchain nodes to actively race to gain the potential to mine a new block. Over a long-term, dependence on electricity for participation in PoW protocol is inefficient and unstable given the cost to produce power in various locations around the world, which also acts as a barrier to entry in some areas where a large amount of power is limited. Instead, utilizing one's stake (s) within a cryptocurrency, such as Cardano, theoretically makes more sense for future adoption. Since players use their own equity in the coin as proof of transaction verification through a proven random process, there is a greater value in protecting the currency and major nothing-at-stake attacks are therefore limited in feasbility. Should a node act dishonest and wish to negatively affect the PoS blockchain, the coin could lose price value and she would suffer losses in her holdings. Users are also

encouraged through this concept to continually invest in the currency as a method to increase their stake and probability to be elected for participation in block generation during a given slot. Ouroboros and other PoS algorithms that focus on probability of stake or coin-age therefore promote the value in a coin even as circulation reaches its limit and rewards are limited, unlike Bitcoin and PoW.

PoS and its effect on the field as a whole remain to be seen, however, due to immaturity on exchanges and a requirement to replace the household name of “Bitcoin”. While it improves on many aspects of protecting users from double spending attacks via multiparty communication in Cardano and minimizes the need for honest miners to engage in bribery attacks, we still are unsure of how it will withstand on large scale deployment. Cardano TPS tests showed significant improvement to Bitcoin, albeit on a much smaller and controlled scale. Much of the hype surrounding security with PoS also comes from assumptions that 51% attacks are more expensive under PoS given it is based on coin value rather than computational energy costs. Should this assumption fail, the system is more prone to adversary activity.

Overall, it appears that PoS in its seemingly early stages of implementation help to provide distributed consensus in alternative cryptocurrencies while minimizing node costs to support the network. In the coming years of lessened rewards under Bitcoin PoW protocol and possibly rising transaction fees to users as coins approach the circulation limit, cryptocurrency users can aim to find value in PoS algorithms for decentralized currencies.

References

Böhme, R., Christin, N., Edelman, B., & Moore, T. (2015). Bitcoin: Economics, Technology, and Governance. *Journal of Economic Perspectives*,*29*(2), 213-238. Retrieved April 8, 2018, from https://pubs.aeaweb.org/doi/pdfplus/10.1257/jep.29.2.213.

Chaum, D. (1983). Blind Signatures for Untraceable Payments. *Advances in Cryptology*,199-203. doi:10.1007/978-1-4757-0602-4_18

Griffith, K. (2014, April 16). A Quick History of Cryptocurrencies BBTC - Before Bitcoin. Retrieved April 13, 2018, from https://bitcoinmagazine.com/articles/quick-history-cryptocurrencies-bbtc-bitcoin-1397682630/

Hankin, A. (2018, March 29). Here's how much it costs to mine a single bitcoin in your country. Retrieved April 4, 2018, from https://www.marketwatch.com/story/heres-how-much-it-costs-to-mine-a-single-bitcoin-in-your-country-2018-03-06

Kiayias, A., Russell, A., David, B., & Oliynykov, R. (2017). Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. *Advances in Cryptology – CRYPTO 2017 Lecture Notes in Computer Science*,357-388. Retrieved March 3, 2018, from https://eprint.iacr.org/2016/889.pdf.

King, S., & Nadal, S. (2012, August 19). PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stak. Retrieved April 1, 2018, from https://peercoin.net/assets/paper/peercoin-paper.pdf

Kroll, J. A., Davey, I. C., & Felten, E. W. (2014, January). The Economics of Bitcoin Mining or, Bitcoin in the Presence of Adversaries. Retrieved March 26, 2018, from The Economics of Bitcoin Mining or, Bitcoin in the Presence of Adversaries

Mulqueen, T. (2018, February 26). 'Now Accepting Bitcoin': A Retailer's Guide To Digital Currencies. Retrieved March 13, 2018, from https://www.forbes.com/sites/tinamulqueen/2018/02/23/now-accepting-bitcoin-a-retailers-guide-to-digital-currencies/2/#231893c4701f

Nakomoto, S. (2008, October). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved February 28, 2018, from https://bitcoin.org/bitcoin.pdf

Nonce. (n.d.). Retrieved April 4, 2018, from https://en.bitcoin.it/wiki/Nonce

Poelstra, A. (2015, March 22). On Stake and Consensus. Retrieved March 8, 2018, from https://download.wpsoftware.net/bitcoin/pos.pdf

Rosic, A. (2017, December 28). Proof of Work vs Proof of Stake: Basic Mining Guide. Retrieved April 10, 2018, from https://blockgeeks.com/guides/proof-of-work-vs-proof-of-stake/

The History and Evolution of Proof of Stake. (2017, October 15). Retrieved March 15, 2018, from https://cointelegraph.com/news/the-history-and-evolution-of-proof-of-stake

Total Bitcoin in Existence vs. Time [Chart]. (n.d.). In *Quora*. Retrieved March 23, 2018, from https://www.quora.com/Extrapolating-from-current-rates-and-numbers-when-will-the-last-bitcoin-be-mined-What-will-be-the-price-at-those-times

Transaction Rate. (n.d.). Retrieved April 10, 2018, from https://blockchain.info/charts/transactions-per-second?timespan=all

Wagner, A. (2014, August 22). Digital vs. Virtual Currencies. Retrieved March 14, 2018, from https://bitcoinmagazine.com/articles/digital-vs-virtual-currencies-1408735507/