The Economics of Cryptocurrencies – Bitcoin and Beyond*

Jonathan Chiu† Bank of Canada

Thorsten V. Koeppl‡ Queen's University

First version: March, 2017

This version: August, 2019

Abstract

How well can a cryptocurrency serve as a means of payment? Cryptocurrencies need to overcome double spending by using costly mining and by delaying settlement. We formalize this insight through an incentive constraint that rules out double spending and pins down the welfare costs of a cryptocurrency. We find that it is optimal to use seignorage rather than transaction fees to finance costly mining. We estimate that Bitcoin generates a welfare loss that is about 500 times larger than a monetary economy with 2% inflation. This welfare loss can be lowered in an optimal design to the equivalent of a monetary economy with inflation of about 50%.

Keywords: Cryptocurrency, Blockchain, Bitcoin, Double Spending, Payment Systems

JEL Classification: E4, E5, L5

*The views expressed in this paper are not necessarily the views of the Bank of Canada. We thank the audiences at many seminars and conferences for their comments. This research was supported by SSHRC Insight Grant 435-2014-1416. The authors declare that they have no relevant or material financial interests that relate to the research described in this paper.

†Bank of Canada, 234 Wellington Street, Ottawa, ON, K1A 0H9, Canada (e-mail: jchiu@bankofcanada.ca).

‡Queen's University, Department of Economics, Kingston, K7L 3N6, Canada (e-mail: thor@econ.queensu.ca).

1 Introduction

How well can a cryptocurrency serve as a means of payment? Since the creation of Bitcoin in 2009, many critics have denounced cryptocurrencies as fraud or outright bubbles. More nuanced opinions have argued that such currencies are only there to support payments for illegal activities or simply waste resources. Advocates point out, however, that – based on cryptographic principles to ensure security – these new currencies can support payments without the need to designate a third-party that controls the currency or payment instrument possibly for its own profit.[1]

We take up this discussion and develop a general equilibrium model of a cryptocurrency that uses a blockchain as a record-keeping device for payments. Many existing models of cryptocurrencies are built by computer scientists who mainly focus on the feasibility and security of these systems. Crucial issues such as the incentives of participants to cheat and the endogenous nature of some key variables such as the real value of a cryptocurrency in exchange have been largely ignored. Such considerations, however, are pivotal for understanding the optimal design and, hence, the economic value of cryptocurrency as a means of payment.

Our focus is thus primarily on understanding how the design of a cryptocurrency influences the interactions among participants and their incentives to cheat. These incentives arise from a so-called "double spending" problem. Cryptocurrencies are based on digital records that can be duplicated easily and costlessly and, thus, can potentially be used multiple times in transactions.

Our first contribution is to formalize this double spending problem and show how it is addressed by (i) a resource intensive competition for updating the records of transactions – a process commonly referred to as mining – and (ii) by introducing confirmation lags for settling transactions.[2] Double spending with a cryptocurrency requires one to change the record of transactions after they have been recorded and confirmed in the blockchain. Since a blockchain builds records based on previous records, to revoke past transactions one must create an alternative history of transaction records in order to double spend successfully. More intensive competition to update the chain and longer confirmation lags make it more costly and hence more difficult to create an alternative history.[3]

[1] Some central banks and financial institutions have also started to explore the adoption of cryptocurrency and blockchain technology for retail and large-value payments. Examples that use the technology are the People's Bank of China for a nationwide digital currency, the Bank of Canada and the Monetary Authority of Singapore for interbank payments, Ripple and JP Morgan for facilitating cross-border payments.

[2] A more detailed description of the double spending problem can be found in the online appendix.

[3] In reality, double spending is a prime concern for cryptocurrencies. In 2018, a number of smaller cryptocurrencies

These insights lead us to derive a constraint – called the "no double spending constraint" – that guarantees that users of a cryptocurrency do not have an incentive to double spend. The constraint formalizes the fact that larger payments require more intensive competition and/or longer confirmation lags in order to reduce the net benefits from recouping a payment through double spending. But the constraint also implies that any cryptocurrency will face a trade-off between how fast transactions settle and a guarantee (or "finality") for their settlement. Consequently, unlike cash cryptocurrencies cannot achieve immediate and final settlement of transactions.

Our second contribution is to look into the optimal design of cryptocurrencies. A key parameter is the reward to finance mining activities and, hence, create competition for updating the blockchain. There are two ways to generate these rewards, seignorage from issuing new cryptocurrency and transaction fees. We show that it is always better to use seignorage rather than transaction fees.

Both inflation and fees are distortionary taxes. Inflation, however, has two advantages. First, when double spending one can reclaim the payment *and* the fee. Hence, using inflation reduces the incentives to double spend and relaxes the double spending constraint. Second, relative to inflation, transaction fees have a lower "tax base" as one only taxes cryptocurrency used currently in transactions. This implies that one has to resort to a higher tax rate which forces users to carry more cryptocurrency and increases the liquidity costs of holding cryptocurrency. Both effects give inflation the advantage when raising the same rewards for mining.[4]

Our third contribution is to quantify the costs of using a cryptocurrency as a payments instrument. We embed our model of a cryptocurrency into the general equilibrium model of Lagos and Wright (2005) where a medium of exchange is required to support decentralized trade.[5] Calibrating our model to Bitcoin data, we assess the welfare costs that arise endogenously from supporting transactions in Bitcoin without the threat from double spending.

The findings are striking. Using the growth rate of 25 bitcoin for every block and average transaction

were subject to double-spending attacks due to the relatively low mining rewards offered. In response to these attacks, users increased the required number of confirmations to settle transactions (see, for example, goo.gl/H2GeJr).

[4]There can be other reasons for imposing some transaction fees. For example, transaction fees rule out spaming as there is a (small) cost to sending a payment. Fees can also be used to sort transaction according to their urgency. Notwithstanding, our result is still valid as such consideration would imply that the optimal design of a cryptocurrency should rely on transaction fees as little as possible.

[5]Koeppl et al. (2008) and (2012) have extended this framework to study payment systems with periodic settlement. In the online appendix, we use their approach to formally define a blockchain as a record-keeping device for payments.

fees in 2015, we find that Bitcoin generates a large welfare loss that is about 500 times as large as in a monetary economy with 2% inflation.[6] The reason is that, in its current form, Bitcoin spends too many resources to rule out double spending. Reducing the growth rate to 0, but relying on sufficiently large transaction fees – like in the long run design of Bitcoin – will reduce these costs significantly. Still, the optimal design of Bitcoin implies relatively large welfare losses. Compared to the first-best allocation, we find that an optimally designed Bitcoin protocol would roughly lead to a loss of about 0.19% of the consumption in the first-best allocation. This is equivalent to the welfare loss that would be generated in a monetary system with an inflation rate of about 45%.

The economic literature on cryptocurrencies is just emerging.[7] We provide the first analysis that captures the interplay of three crucial elements of a cryptocurrency: its security, its value and its mining ecosystem. As shown in Figure 1, sufficient mining is required for ensuring the security of the blockchain, safeguarding it against double spending attacks. Moreover, only when users trust the security of the system will the cryptocurrency be widely accepted and traded at a high value. Finally, the value of the currency supports a reward scheme that incentivizes miners so that they engage in sufficient mining.

The existing literature also evolves around these three basic components, but does not connect them in a unified framework. A large amount of research has focused on understanding the valuation of cryptocurrencies. Uhlig and Schilling (2018) and Biais et al. (2018) use an Euler equation approach to capture the price dynamics in models where Bitcoin is being used both for transactions and speculation. Choi and Rocheteau (2019) use a model of mining choice to study the dynamics and indeterminancy of equilibrium prices of a cryptocurrency. Cong et al. (2018) study the price dynamics of cryptocurrencies in a model with endogenous user adoption. In our model, the value of a cryptocurrency arises from its use as a means of payment that enables decentralized exchange.[8]

A second line of research investigates how the incentives of miners influence the reliability of the

[6] We use 2015 for our benchmark calibration as more recent data data may be contaminated by extreme price volatility. We found consistent results when using more recent data after Bitcoin reward was halved to 12.5 BTC.

[7] There is some earlier work on related topics such as digital money in the form of smart cards (Berentsen (1998)), digital currencies on platforms such as Facebook Credits (Gans and Halaburda (2013)) or general e-money technologies such as PayPal and Octopus Card (Chiu and Wong (2015)). See also Camera (2017) who discusses the challenges of issuing and adopting electronic alternatives to cash.

[8] See also Fernández-Villaverde and Sanches (2016) that study cryptocurrencies as privately issued fiat currencies and analyze – in the tradition of the literature on the free banking era – whether competition among different currencies can achieve price stability and efficiency of exchange.

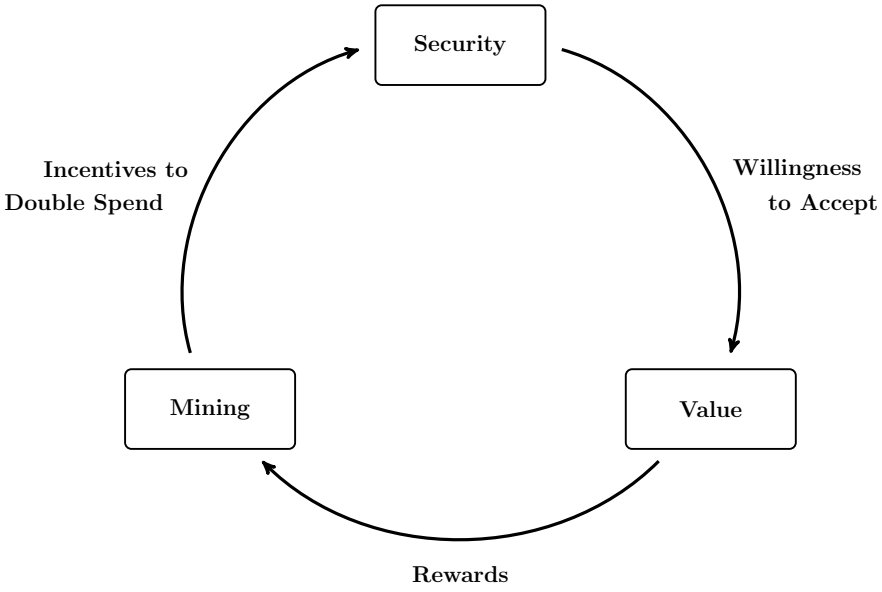


Figure 1: The Interplay of Mining, Security and Valuation in a Cryptocurrency

blockchain. Biais et al. (2017) model the proof-of-work protocol as a stochastic game and analyse the incentives of miners to create forks. Eyal and Sirer (2013) assess the incentives for selfish mining under the Bitcoin protocol. Saleh (2019) examines an alternative protocol, Proof-of-Stake and establishes conditions under which this protocol can generate consensus on the state of a blockchain. We focus instead on the incentives of traders to double spend when competing against honest miners.

Finally, some work tries to understand the relationship between mining and fees. Huberman et al. (2017) and Easley et al. (2019) both study the willingness of users in Bitcoin to post fees that support costly mining. The key channel here is that Bitcoin offers limited throughput that makes fast settlement a scarce resource. We look at a similar channel when endogenizing transaction fees to discuss our main results.

Only very recently, some papers have started to also look into the design of a cryptocurrency. The closest work to ours is Pagnotta (2018) who presents an equilibrium analysis to study the equilibrium prices, mining and usage of Bitcoin. He concentrates, however, on the stability properties of cryptocurrency equilibria against network attacks instead of the double spending problem. Auer

(2019) also discusses the fundamental difference between coin creation and fees to generate revenue, but mainly questions the ability to generate sufficient fees in the long run to offer sufficient rewards for mining in Bitcoin.[9]

2 A Model of the Double Spending Problem

As pointed out in the previous section, due to its digital nature, a cryptocurrency system is subject to the double spending problem. In Appendix B, we describe how Bitcoin is designed to tackle this problem. Importantly, we highlight that the system has the following three key features. First, a consensus protocol based on proof-of-work (PoW) according to which miners compete to update a blockchain. Second, a reward scheme that gives miners incentives to compete for updating the blockchain. And third, settlement lags such that N validations in the blockchain are required before fulfilling any obligations. We will show that these features make revoking a payment (aka double spending) difficult. Our model will formally capture all these key features.

This section first develops a partial equilibrium model to study the incentives to double spend in a single period taking transactions as given. In the next section, we will incorporate this basic set-up into a general equilibrium monetary model to endogenize both transactions in cryptocurrency and mining rewards financed by transaction fees and seignorage. Throughout the paper, we make two facilitating assumptions to streamline our analysis. First, we assume that the blockchain does not have any capacity constraints. In other words, there is no congestion and all transactions within a single period can be included in a single block.[10] Second, we approximate the attempt to double spend by a sequence of exponential races where one needs to win all contests sequentially in order to double spend successfully. This allows for a closed-form solution to describe the incentives to double spend.[11]

[9] A related area of research is on central banks using digital currency. For example, Agarwal and Kimball (2015) advocate that the adoption of digital currencies can facilitate the implementation of a negative interest rate policy, while Rogoff (2016) suggests that phasing out paper currency can undercut undesirable tax evasion and criminal activities. For a general overview, see Bordo and Levin (2017).

[10] This is without loss of generality, since fees for submitting transactions to the blockchain will be part of the design of the blockchain later on. For a model on how congestion drives fees, see Chiu and Koeppl (2019). Also, for most time periods Bitcoin did show only little to moderate congestion. This was certainly the case until early 2017 and includes the periods to which we calibrate the model.

[11] In general, a double spending attacker can catch up with the blockchain when falling behind in the sequence of

2.1 Basic Set-up

We begin our analysis by looking at a single transaction period. As shown in Figure 2, there are $\bar{N}+1$ subperiods within the single period. In subperiod 0, a buyer meets a seller to negotiate a trade. All other subperiods $1,\ldots,\bar{N}$ serve as periods for confirming and settling trades that take place in subperiod 0.

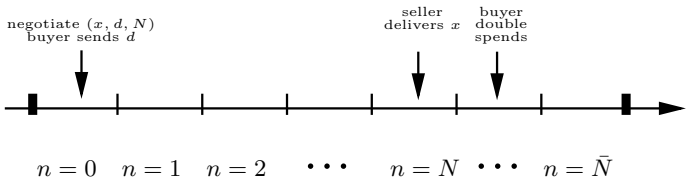


Figure 2: Timeline for a single transaction period

The buyer carries a real balance of cryptocurrency equal to z that can be used to buy an amount of goods x from a seller. Upon being matched, the buyer and the seller bargain to determine the terms of trade (x,d,N) which specify that the buyer pays the seller $d\leq z$ units of real balances and that the seller commits to deliver x units of goods after a number of successive payment confirmations $N\in\{0,\ldots,\bar{N}\}$ in the blockchain.[12] We call N the *confirmation lag* of the transaction. For now, the terms of trade are taken as given, but will be determined endogenously in the next section.

The seller produces the good at unit costs, while the buyer's preference for consuming an amount x with confirmation lag N are given by

$$\delta^N u(x) \tag{1}$$

where $\delta\in(0,1)$ is the discount factor between two subperiods and u is a continuous, increasing and strictly concave function. Hence, discounting across the whole transaction period is given by[13]

$$\beta=\delta^{\bar{N}+1}. \tag{2}$$

Finally, both buyers and sellers value real balances linearly and discount all payoffs that arise after the single transaction period at β.

exponential races. This problem cannot be solved analytically and we leave it for future research.

[12] We consider the case in which a seller can commit to deliver the good. If this were not the case, only spot trades can be conducted that – as we show later – will always be subject to double spending.

[13] There are two ways to interpret the discount factor δ: (i) buyers prefer earlier consumption or (ii) a buyer's preference can change over time so that a seller's goods will no longer generate utility with probability $1-\delta$.

2.2 Mining

There are M miners who compete to update the blockchain with one block for each subperiod $n = 0, ..., \bar{N}$. Since there is no capacity constraint, the first block contains all transactions from the subperiod thereby validating them. All other blocks are empty, but verify the update of the first block.

For each subperiod, miners perform exactly one costly computational task with a random success rate by investing computing power, q, measured in real balances of cryptocurrency. This task is called the Proof-of-Work. We assume that miners also value real balances linearly.

As motivated by the Bitcoin protocol, if the computational power of miner i in a subperiod is $q(i)$, then the probability that a particular miner i will win the mining game is given by

$$\rho(i) = \frac{q(i)}{\sum_{m=1}^{M} q(m)}. \tag{3}$$

In other words, the probability of wining is proportional to the fraction of computational power owned. We take this feature as given here and provide a micro foundation for this result in Section **??**. By winning the competition in any subperiod, a miner can update the blockchain (i.e., append the nth block to the blockchain in the subperiod) and receives a reward R in real balances. We assume that miners receive and consume this reward after the period, discounted by the factor β.

Note that the mining games are independent across subperiods. Hence, taking as given the choice by all miners $m \neq i$, a miner i solves in any subperiod

$$\max_{q(i)} \rho(i)\beta R - q(i) \tag{4}$$

with the first-order condition given by

$$\frac{\sum_{m=1}^{M} q(m) - q^*(i)}{\left(\sum_{i \neq j} q(m) + q^*(i)\right)^2} \beta R = 1. \tag{5}$$

Imposing $q(m) = Q$ for all m, we obtain

$$Q = \frac{M-1}{M^2} \beta R \tag{6}$$

as the Nash equilibrium of the mining game. Consequently, the total computing cost of mining in any subperiod is

$$MQ = \frac{M-1}{M} \beta R. \tag{7}$$

The expected profit of a miner in equilibrium across the single transaction period is thus given by

$$\Pi_m = (\bar{N}+1)\left[\frac{Q}{\sum_{m=1}^{M} Q}\beta R - Q\right] = \frac{\bar{N}+1}{M^2}\beta R. \tag{8}$$

To capture the fact that mining tends to be competitive and open to new entrants, we assume that $M \to \infty$ for the remainder of the paper.[14]

Lemma 1. *As $M \to \infty$, the expected value of miners is zero, and the aggregate computing power of miners dissipates all rewards from mining*

$$MQ = \beta R.$$

2.3 Secret Mining

To complete a transaction, the buyer sends instructions to miners to update the blockchain with his payment to the seller. Seeing such an instruction, however, is not enough for the seller to be sure to receive the payment. Why? The buyer can attempt to mine a block himself in which the payment is not recorded. A seller, however, can protect himself from not receiving the payment by waiting to deliver the goods until the payment has been incorporated into the blockchain.[15] A simple confirmation of the payment in the blockchain, however, may still not be enough. A buyer can secretly mine a different blockchain which he releases only some periods after the seller has delivered the good to replace the original blockchain. We call such mining *secret mining*.

When such secret mining succeeds, the buyer keeps his original balances and the goods while the seller will be left empty handed – in other words, the buyer *double spends* the cryptocurrency.[16] In response, the seller can choose to postpone the delivery of the goods and wait for N confirmations.

[14] In reality, mining tends to be organized around mining pools where individual miners pool their resources to compete against other miners (see Cong, He and Li (2018) for an analysis of strategic competition between a finite number of mining pools). Still, mining appears to be competitive as currently 10 mining pools comprise about 95% of computational power (see https://www.blockchain.com/en/pools). Miners can also shift their computational power between mining different cryptocurrencies ensuring that there is free entry.

[15] For more technical details, please see Appendix B.

[16] Note that a buyer cannot spend the balances of any other agent, because these balances are protected by cryptography. Hence, a buyer can only (i) change the payment instructions of his own transaction and (ii) remove other payment instructions from being mined – and, hence, confirmed – in a block. This implies that in reality a buyer trying to double spend has to remove his own payment and all other payment instructions involving his original balance being spent again.

This confirmation lag can potentially deter double spending by the buyer. The idea is that, to undo a transaction with a confirmation lag of N subperiods, a dishonest buyer needs to win the mining game at least $N+1$ times. As the number of lags increases, the total PoW required to revoke a transfer is increasing, making it more costly for a buyer to double spend.

When double pending, the buyer competes against honest miners. Consequently, secret mining is also deterred by the total investment in computing power which, according to Lemma 1, is increasing in the reward R that is offered for finding new blocks. We now look into the incentives to double spend and call an offer (x, d, N) *double spending proof*, if the buyer has no incentive to engage in secret mining in order to double spend.

2.4 Double Spending Proof Offers

Consider a trade with the terms (x, d, N). The buyer will receive the goods in subperiod N when exactly N confirmations of the payment d have been observed in the blockchain. To double spend, a buyer can secretly mine an alternative history and undo his original payment after he has received the goods.

For simplicity, we assume that to be successful the buyer needs to be first solving the PoW problem in each of $N+1$ consecutive subperiods. For each of the first N subperiods, however, the buyer does not announce the solution immediately, so that some other miners will update the Blockchain and confirm his payment to the seller. The buyer announces his secretly mined blockchain only after he receives the goods and solves the $N+1$th PoW problem. When the double spending attack succeeds, the buyer has received the good x, the original payment is cancelled and the $N+1$ rewards will be given to the buyer (for a stylized representation of the double spending problem see Figure 3).

To analyze the incentives to double spend, note first that the quantity of good x being exchanged does not directly matter for a double spending attack, as it neither changes the payoffs, nor the incentives to engage in an attack. Consider subperiod N where the seller delivers his goods. Conditional on having been successful N times with secret mining, a buyer can now double spend successfully by mining a new block in subperiod N. His expected payoff from investing q_N is given by

$$\rho(q_N)\beta(d + (N+1)R) - q_N. \tag{9}$$

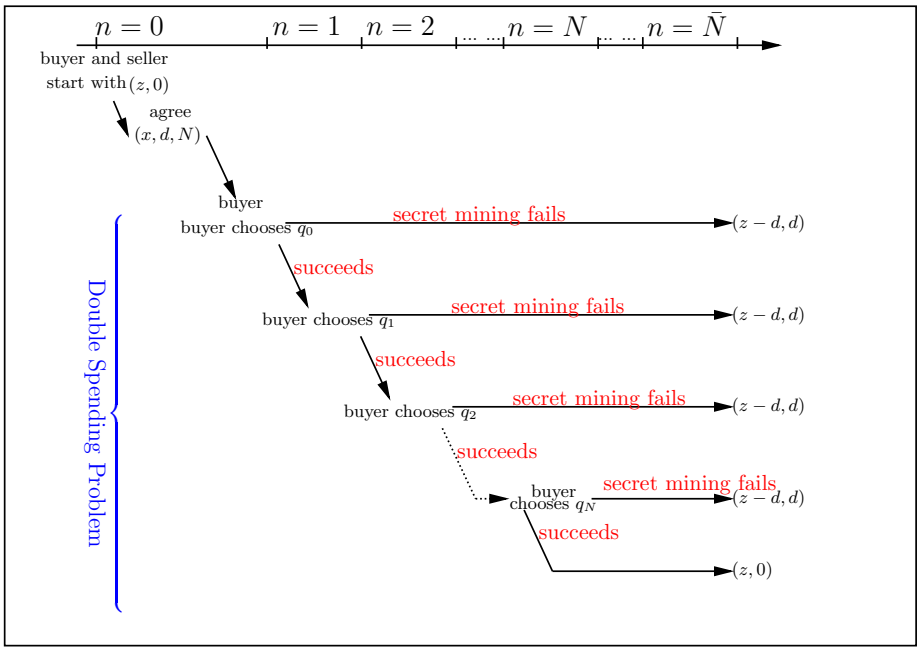


Figure 3: Double Spending Problem – Stylized Description

The first term captures the expected revenue from double spending. Conditional on having solved N blocks, with probability

$$\rho(q_N) = \frac{q_N}{MQ + q_N} \tag{10}$$

the buyer wins the competition again in the $N + 1$th round, so that he can double spend. The revenue from double spending is given by the original payment in subperiod 0 which is d. Also, the buyer obtains the rewards from all the new blocks in his chain of length $N + 1$. The value in real balances of a successful double spend is thus given by $\beta(d + R(N + 1))$, where we have taken into account that the real balances can only be spent in the following period.

Define

$$\Delta = \left(\frac{d}{R} + (N + 1)\right). \tag{11}$$

Since $MQ = \beta R$, the buyer's optimal choice of investing in computing power in subperiod N is given by

$$\hat{q}_N(d, N) = \beta R\left(\sqrt{\Delta} - 1\right) \tag{12}$$

so that the probability of successful double spending in subperiod N is equal to

$$\rho(q_N) = \frac{\sqrt{\Delta} - 1}{\sqrt{\Delta}}. \tag{13}$$

To find conditions for offers to be double spending proof, we work backwards to subperiod 0. Given (d, N) and R, the expected payoff for a double spending buyer in subperiod N having been successful N times already is

$$D_N(d, N) = \beta R(\sqrt{\Delta} - 1)^2. \tag{14}$$

Define recursively the expected payoff from double spending in subperiod n for $n \in \{0, \dots, N-1\}$ by

$$D_n(d, N) = \max_{q_n} \rho(q_n) D_{n+1}(d, N) - q_n, \tag{15}$$

which takes into account that the buyer was n times successful, as otherwise the attempt to double spend has failed already. Note that $D_n(d, N)$ can only be positive if $D_{n+1}(d, N)$ is positive and $\hat{q}_n > 0$ only if $D_n(d, N) > 0$. The first-order condition describing the optimal investment is thus given by

$$\hat{q}_n(d, N) = \sqrt{\beta R \cdot D_{n+1}(d, N)} - \beta R. \tag{16}$$

By backward induction, we then obtain the following result.

Lemma 2. *The expected payoff, probability of success and optimal investment for double spending in subperiod $N - s$ are given by*

$$D_{N-s}(d, N) = \beta R \left(\sqrt{\Delta} - (s+1)\right)^2 \tag{17}$$

$$\rho_{N-s}(d, N) = \frac{\sqrt{\Delta} - (s+1)}{\sqrt{\Delta} - s} \tag{18}$$

$$\hat{q}_{N-s}(d, N) = \beta R \left(\sqrt{\Delta} - (s+1)\right). \tag{19}$$

It follows immediately that $D_n(d, N)$ is strictly increasing in n and, consequently, that the investment into double spending $\hat{q}_n$ is also increasing in n. Hence, if it was optimal to engage in secret mining in subperiod n and one has been successful in subperiod n to solve the PoW problem first, it is also optimal to continue with secret mining in subperiod $n+1$. Consequently, double spending is not optimal for the buyer whenever

$$\beta R \left(\sqrt{\Delta} - (N+1)\right) < 0,$$

so that $\hat{q}_0 = 0$ and $D_0(d, N) = 0$. This yields the following *no double spending constraint* (NDS).[17]

[17] When constraint (20) is binding, a buyer may consider to offer a contract that gives him an incentive to double spend, but compensates the seller for this possibility with a higher, but uncertain payment d. In Section 3, we introduce a sufficient condition so that contracts that violate (20) can never increase the joint surplus of the buyer and seller and, consequently, will never be offered.

Proposition 3. *A contract is double spending proof if*

$$d \leq R(N+1)N. \tag{20}$$

The intuition for the NDS constraint (20) is straightforward. The reward R increases the incentives to mine and, thus, increases the costs of a double spending attempt. Similarly, for a given payment d, one can rule out double spending by increasing the confirmation lag N as this reduces the probability of a successful double spend. Interestingly, while the impact of the reward R is linear, the effect of N follows a power law. The intuition is that lengthening the confirmation lag N reduces the probability of a successful double spend exponentially, while growing additional rewards only linearly.

More generally, the expected payoff from double spending is given by

$$D_0(d,N) = \beta R\left(\sqrt{\Delta} - (N+1)\right)^2 \tag{21}$$

which is decreasing in R and N and increasing in d. The unconditional probability of a successful double spending is

$$P(d,N) = \frac{\sqrt{\Delta} - (N+1)}{\sqrt{\Delta}}. \tag{22}$$

This allows us to study the settlement properties of a cryptocurrency system. Settlement in our context is the delivery of the good x against payment of d between the buyer and the seller. We define the following concepts.

Definition 4. *The settlement of a contract (x,d,N) is immediate if $N=0$ and delayed if $N>0$. The settlement is final if $P(d,N)=0$ and probabilistic if $P>0$.*

The NDS constraint (20) can thus be interpreted as a condition for finality. Given any payment d and given rewards R, one needs to delay settlement sufficiently to guarantee settlement. As shown in Figure 4, this points to a trade-off between the trade size d, the settlement lag N and finality as captured by $1-P(d,N)$. Given a reward R, finality is only feasible for small transactions and sufficiently delayed settlement. Fast settlement of large transactions can only be probabilistic with the probability decreasing in d and increasing in N. This can be summarized in the following result.

Theorem 5. *For any cryptocurrency based on a PoW protocol, settlement cannot be both immediate and final.*

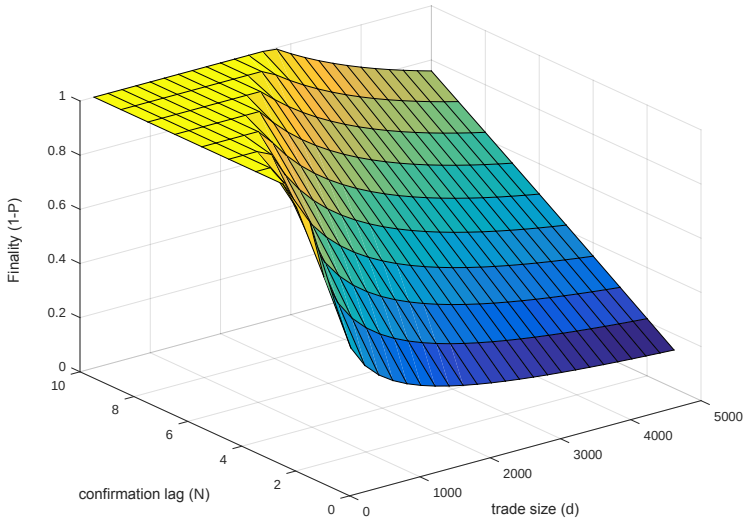


Figure 4: Trade Size, Confirmation Lag and Probabilistic Finality

3 General Equilibrium Framework

We now incorporate our model of double spending into a general equilibrium model where a cryptocurrency is used as a medium of exchange. Recall from the introduction that a cryptocurrency is a closed system where the value of the cryptocurrency determines the rewards that miners receive. These rewards determine the mining effort and, thus, the incentives to double spend. This in turn will feed back into whether people accept the cryptocurrency as a medium of exchange and, thus, the value of the currency. Our general equilibrium can therefore be used to explore the optimal design of a cryptocurrency and to compare the surplus produced by an existing cryptocurrency such as Bitcoin with the surplus of a traditional monetary system.

3.1 A Dynamic Model of Trade

Our model is based on Lagos and Wright (2005). Time is discrete and denoted by $t = 0, 1, 2, ...$ There are a large number B of buyers and a large number $S = \sigma B$ of sellers, where $\sigma \in (0, 1)$. There is also an infinite number of miners so that mining is perfectly competitive.

In each period, there is first a centralized, competitive market where people trade a general good h that they can produce and consume. Then a decentralized market opens where buyers and sellers

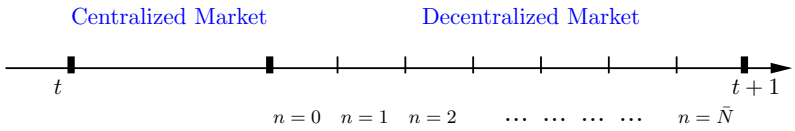


Figure 5: Time line

meet bilaterally. As shown in Figure 5, the time when this market is open is divided further into $\bar{N}+1$ consecutive subperiods with $\bar{N} \geq 1$. In the first subperiod, individual buyers are matched to individual sellers with probability σ. Once matched, sellers can produce a good x for buyers at unit cost in the first subperiod. To introduce heterogeneity among transactions, buyers' preferences are given by

$$\delta^N \varepsilon u(x) \tag{23}$$

where ε is a random variable being drawn from a distribution $F(\varepsilon)$ with support in the interval $[\underline{\varepsilon}, \bar{\varepsilon}] \subset (0, \infty)$ when the centralized market opens. We assume that ε is known to the buyer and seller in a match and that the seller can commit to deliver the good in subperiod N. Preferences express the fact that buyers discount consumption of the good across subperiods according to $\delta \in (0,1)$ and we assume that all people discount utility across periods according to $\beta = \delta^{\bar{N}+1}$. We define $x^*(\varepsilon)$ as the first-best level of consumption such that $\varepsilon u'^*(\varepsilon) = 1$.

3.2 Cryptocurrency

A cryptocurrency is a digital record of ownership of nominal balances m that can be used to pay for transactions. For any transaction, the buyer gives instructions to transfer ownership of a certain amount of his balances to the seller.[18] We assume that transactions and payments in the centralized market can be perfectly monitored. This implies that when a person makes a payment in cryptocurrency in this market he is liable for their authenticity. More precisely, if a payment were to be undone for the payee, the payer would need to reimburse the payee for the loss. This rules out incentives to double spend in the centralized market.[19] To the contrary, trades in the

[18] In the online appendix, we follow Koeppl et al. (2008) to formalize the notion of a blockchain as a transaction-based ledger that records these transfers of cryptocurrency balances throughout time.

[19] One can interpret transactions in the centralized market as special in that they convert crytocurrency into another financial asset such as regular currency. In reality, this regularly occurs through exchanges that need to keep customer data on file and, hence, make transactions in cryptocurrency traceable. Furthermore, depending on the type of transaction, metadata often allow one to link transactions to people. For these reasons, cryptocurrencies are

decentralized market are anonymous. Hence, such trades require a medium of exchange which we assume to be cryptocurrency and, thus, these trades are subject to a double spending problem. The updating of the blockchain with such trades follows a PoW protocol where miners compete for the right to add a new block to the chain each subperiod.

The mining game is identical to the one presented in Section 2, but the reward R is now not exogenous anymore and depends on the design of the cryptocurrency. First, the cryptocurrency can create new balances which are paid to miners that win the competition to update the blockchain. We denote the growth rate of new balances by $\mu \geq 1$. In addition, the blockchain can set a transaction fee for including transactions into a block. We denote $\tau \geq 0$ the fraction of the payment in cryptocurrency that is to be pledged as a fee for the transaction to be included into a block. For simplicity, the $\bar{N}+1$ block winners share the total reward equally. Consequently, the reward per block is given by

$$R = \frac{Z(\mu - 1) + D\tau}{\bar{N} + 1} \tag{24}$$

where Z is the aggregate, real balance of cryptocurrency in circulation and D are the aggregated payments in the decentralized market. Rewards for blocks are only paid next period in the centralized market. Since they are expressed in real terms, we need to take into account inflation across periods. Consequently, miners will discount the reward they receive in real terms by β/μ.

3.3 Centralized Market

In the centralized market, all buyers, sellers and miners can produce and consume the general good with a linear utility function. Miners can also convert general goods one for one into computing power q at any time. Since they cannot transact in the decentralized market, miners will not hold balances across periods due to discounting. Upon receiving balances as rewards, they will simply sell them for the general good in the centralized market. Hence, the problem of miners is identical to the one we analyzed in Section 2 adjusted simply for the discount factor $\beta/\mu \leq \beta < 1$.

often labelled as *pseudonymous*. Newer versions of cryptocurrencies are designed to give full anonymity. However, even these cryptocurrencies can be traced to people once they are traded on an exchange. Finally, some transactions can be interpreted to take place with known counterparties where an independent, auditable record of the transaction exists. Reputation and legal recourse can thus discourage double spending for such transactions.

To ease exposition, we abuse notation and set $\varepsilon = 0$ for the seller. We use $w(z, \varepsilon)$ and $v(z, \varepsilon)$ to denote the value functions in the centralized market and the decentralized market, respectively. The problem of a buyer that draws ε (or a seller with $\varepsilon = 0$) entering the centralized market with real balances z is given by

$$w(z, \varepsilon) = \max_{z', h} -h + v(z', \varepsilon) \tag{25}$$

subject to

$$z + h \geq z' \geq 0 \tag{26}$$

where $h > 0$ ($h < 0$) denotes production (consumption) of the general good and z' are the real balances carried into the decentralized market which have a value of $v(z', \varepsilon)$. The optimal demand for balances is

$$1 \geq \frac{\partial v(z', \varepsilon)}{\partial z'} \tag{27}$$

with equality when $z' > 0$. Linear preferences imply that

$$w(z, \varepsilon) = z + w(0, \varepsilon). \tag{28}$$

Hence, the value function before the realization of ε is

$$\bar{w}(z) = E[w(z, \varepsilon)] = z + W \tag{29}$$

where $W = E[w(0, \varepsilon)]$ is a constant.

3.4 Decentralized Market

With probability σ, a buyer meets a seller in the decentralized market. Holding real balances z and being of type ε, he then makes a take-it-or-leave-it offer (x, d, N) which specifies a payment $d \leq z$ for obtaining x goods to be delivered after confirmation of the payment in N consecutive blocks. The value of a buyer entering the decentralized market with real balances z is then given by

$$\begin{aligned} v(z, \varepsilon) &= \sigma \left(\delta^N \varepsilon u(x) + D_0(d, N) + \beta \bar{w}((z - d)/\mu) \right) + (1 - \sigma) \beta \bar{w}(z/\mu) \\ &= \frac{\beta}{\mu} z + \sigma \left(\delta^N \varepsilon u(x) + D_0(d, N) - \frac{\beta}{\mu} d \right) + \beta W \end{aligned} \tag{30}$$

where $D_0(d, N)$ is the expected value from double spending. It is understood here that $D_0(d, N) = 0$ if the buyer has no incentive to double spend given the offer (x, d, N). Also note that the value

of unspent real balances declines between different periods because of discounting and inflation μ. Since the seller has a linear technology to produce x, the value function of a seller accepting the offer is given by

$$v(z,0) = \frac{\beta}{\mu} d(1-\tau)\left(1 - P(d,N)\right) - x + \frac{\beta}{\mu} z + \beta W. \tag{31}$$

With probability $P(d,N)$ a double spending attack is successful leaving the seller without balances from the trade, where $P(d,N) = 0$ if the buyer has no incentive to double spend. Furthermore, the seller only receives the payment net of the transaction fee which is a fraction τ of the total payment in terms of real balances d.

If balances can be used for trading in the decentralized market, buyers will have a positive demand for real balances ($z' > 0$). Furthermore, since $\mu \geq 1 > \beta$, it is costly to carry extra real balances into the decentralized market. This implies that sellers do not carry any balances, while buyers carry only the amount of balances into the decentralized market they will use to make an offer ($z' = d$). The buyer's optimal take-it-or-leave-it offer is thus a solution to the problem

$$\max_{(x,d,N)} -d + (1-\sigma)\frac{\beta}{\mu} d + \sigma\left(\delta^N \varepsilon u(x) + D_0(d,N)\right) \tag{32}$$

subject to

$$d = \frac{\mu}{\beta} \frac{x}{(1-\tau)\left(1 - P(d,N)\right)} \tag{33}$$

where the constraint expresses the fact that the seller receives zero expected surplus. The first term in the objective function expresses the expected cost of transacting in the decentralized market. The buyer needs to acquire real balances d in the centralized market, but can spend them only with probability σ to reap a surplus from trade in the decentralized market. With probability $1-\sigma$, he cannot trade and his real balance depreciates in value by β/μ due to discounting and inflation.

Buyers and sellers may have an incentive to negotiate an offer that involves a positive probability of double spending. The buyer needs to compensate the seller for the expected loss from double spending with a higher promised payment d, but needs to profit himself from consuming the quantity x earlier due to a lower settlement lag N. The values of $D_0(d,N)$ and $P(d,N)$ in the buyer's problem depend therefore on whether the offer is double spending proof or not. The mining game and the buyer's incentives to engage in double spending are identical to Section 3 except for that payoffs are discounted across periods by $\beta/\mu < \beta$. By Proposition 3, the offer is then double spending

proof and $P(d,N) = D_0(d,N) = 0$ if and only if it satisfies condition (20). Otherwise, we have

$$D_0(d,N) = \frac{\beta}{\mu} R \left(\sqrt{\Delta} - (N+1) \right)^2 > 0 \tag{34}$$

with $P(d,N)$ given by equation (22). Note that double spending gives the buyer an additional payoff $D_0(d,N)$, but also tightens the seller's participation constraint, thereby reducing the amount being transacted in the decentralized market.

In general, the buyer's problem may have multiple solutions. For example, a buyer can be indifferent between double spending and a double spending proof offer. Similarly, a buyer can be indifferent between a contract with a long confirmation lag and large consumption and one with earlier but smaller consumption. For completeness, the seller's value function is given by

$$v(z,0) = \frac{\beta}{\mu} z + \beta W. \tag{35}$$

so that sellers do not carry any real balances into the decentralized market.

3.5 Double Spending Proof Equilibrium

In what follows, we concentrate on equilibria in which the cryptocurrency is being used as a medium of exchange in the decentralized market and all offers are double spending proof. Since we cannot ensure that such offers are preferred, we assume that there are costs (ξ_b, ξ_s) for the buyer and seller to negotiate such an offer.[20] We show in the appendix that imposing the following sufficient condition on these costs can rule out a preference of buyers and sellers for offers that imply double spending.

Lemma 6. *It is optimal for the buyer to make a double spending proof offer if*

$$\xi_b + \xi_s \left(\frac{1 - (1-\sigma)\beta}{\sigma\beta} \right) \geq \varepsilon_{\max} u(x^*(\varepsilon_{\max}))(1-\beta). \tag{36}$$

Note that buyers still have an incentive to double spend. Lemma 6 simply allows us to impose the NoDS constraint (20) directly on the bargaining problem, since buyers will not make any offers that

[20] In practice, it seems difficult to negotiate such offers. The fixed costs are meant to capture the time costs involved and the costs of forecasting accurately the probability of successful double spending. Allowing for double spending in equilibrium also goes against the very idea to rule it out through the design of the cryptocurrency protocol (see for example Nakamoto (2008)).

lead to actual double spending. The sufficient condition reflects that the advantage to consume faster increases with discounting, while the liquidity cost falls with the likelihood of trading. Hence, small σ and a discount factor close to 1 reduce the benefit of a double spending offer. Also, the condition can be fulfilled even if only one party to the trade incurs a fixed cost. We assume from now on that this condition is satisfied, so that we can rewrite the buyer's problem as

$$\max_{(x,d,N)} -d + (1-\sigma)\frac{\beta}{\mu}d + \sigma\delta^N \varepsilon u(x) \tag{37}$$

subject to

$$d = \frac{\mu}{\beta}\frac{x}{1-\tau} \tag{38}$$

$$d \leq RN(N+1) \tag{39}$$

Fix rewards R and define the set of optimal real money demand as $\Gamma(\varepsilon; R)$. For a given selection $z^*(\varepsilon; R)$ from the set $\Gamma(\varepsilon; R)$, aggregate real money demand and aggregate real payments in the decentralized market are given by

$$Z = BE(z) = B\int_{\underline{\varepsilon}}^{\bar{\varepsilon}} z^*(\varepsilon; R)dF(\varepsilon) \tag{40}$$

$$D = \sigma BE(z) = \sigma B\int_{\underline{\varepsilon}}^{\bar{\varepsilon}} z^*(\varepsilon; R)dF(\varepsilon). \tag{41}$$

Definition 7. *A double spending proof cryptocurrency equilibrium with (μ, τ) is given by offers $(x^*(\varepsilon), d^*(\varepsilon), N^*(\varepsilon))$, real money demand $z^*(\varepsilon) > 0$ and a mining choice q^* such that*

1. *real money demand and the offer solves the buyer's problem (37)-(39) for all ε taking as given rewards R*
2. *the mining choice is a Nash equilibrium of the mining game in every subperiod given R*
3. *the centralized market for real balances clears*
4. *rewards R are generated by (μ, τ) and real money demand.*

We next show that under a fairly weak additional condition a double spending proof equilibrium exists. We only discuss the intuition for this result and relegate the formal proof of the result to the appendix. One needs a sufficiently large number of cryptocurrency users B. This ensures

that – for any given (μ, τ) – the reward R is sufficiently high to support enough mining to rule out double spending. If R is too small, double spending can only be averted by requiring a large number of confirmations N. This, however, could make the surplus too small to have trade in the decentralized market. This result thus highlights an important feature of a cryptocurrency: the mining reward R is financed by the *aggregate* transaction volume which increases with the number of users B, while the incentives to double spend depend only on the *individual* transaction size d.

Proposition 8. *If the number of buyers B is sufficiently high, a double spending proof cryptocurrency equilibrium exists.*

3.6 Optimal Cryptocurrency Design

We now look into the optimal design of a cryptocurrency. Since we take the PoW protocol as given, this is equivalent to looking at how to optimally finance the reward of miners to maximize social welfare. As a first step, we restrict attention to steady state allocations and derive a social welfare function.

For steady state allocations, the aggregate surplus for buyers is given by

$$\int_{\underline{\varepsilon}}^{\bar{\varepsilon}} B\left(-d(\varepsilon) + (1-\sigma)\frac{d(\varepsilon)}{\mu} + \sigma\delta^{N(\varepsilon)}\varepsilon u(x(\varepsilon))\right) dF(\varepsilon). \tag{42}$$

All buyers replenish their balances to d in order to transact in the centralized market. There are $B(1-\sigma)$ buyers that have not spent their balances since they had no trade in the previous decentralized market. Their real balances are given by $\frac{d}{\mu}$ taking into account inflation. Finally, a measure of $B\sigma$ of buyers has a transaction in the current decentralized market. Similarly, the aggregate surplus of sellers is given by

$$\int_{\underline{\varepsilon}}^{\bar{\varepsilon}} \sigma B\left(\frac{d(\varepsilon)}{\mu}(1-\tau) - x(\varepsilon)\right) dF(\varepsilon). \tag{43}$$

There are $B\sigma$ sellers who had a trade in the decentralized market in the previous period and earned $d(1-\tau)/\mu$ real balances taking into account inflation. Also, $B\sigma$ sellers will have a trade in the current period decentralized market where they produce an amount x. Finally, the aggregate surplus of miners is given by

$$(\bar{N}+1)\left(\frac{R}{\mu} - \beta\frac{R}{\mu}\right) = B\frac{(\mu - 1 + \sigma\tau)}{\mu} Z(1-\beta). \tag{44}$$

In the centralized market miners turn the block reward R/μ they have earned from mining $\bar{N}+1$ blocks in the previous decentralized market into the general good for consumption. At the same time, they make an investment into new computational power for each block. Lemma 1 implies that the investment is simply the real present value of the total mining rewards $\beta\frac{R}{\mu}$ per block.

Summing over these expressions, we obtain that total surplus in any period t is given by

$$\mathcal{W} = B\sigma \int_{\underline{\varepsilon}}^{\bar{\varepsilon}} \left(\delta^{N(\varepsilon)}\varepsilon u(x(\varepsilon)) - x(\varepsilon)\right) dF(\varepsilon) - \beta\frac{R}{\mu}(\bar{N}+1). \tag{45}$$

This is intuitive. Aggregate welfare is given by the surplus generated in the decentralized market and the aggregate cost spent on mining which is the second term. All other value generated in the centralized market is simply lump-sum transfers between people.

To characterize the optimal design, we look at a Ramsey problem. A social planner chooses the optimal inflation rate μ and fee rate τ to maximize aggregate welfare.[21] When doing so, he has to take into account that his choice will change the equilibrium in the economy. This implies that the constraints for the planner are the market clearing condition and the fact that buyers make optimal take-it-or-leave-it offers that are double spending proof. For stationary allocations, the problem for the social planner is then given by

$$\max_{\mu,\tau} B \int \sigma[\delta^{N(\varepsilon)}\varepsilon u(x(\varepsilon)) - x(\varepsilon)]dF_\varepsilon(\varepsilon) - \frac{\beta}{\mu}R(\bar{N}+1) \tag{46}$$

subject to

$$(d(\varepsilon), x(\varepsilon), N(\varepsilon)) \text{ solves problem (37) - (39) for all } (\mu, \tau, R) \tag{47}$$

$$\frac{R(\bar{N}+1)}{\mu} = B(\mu - 1 + \sigma\tau)\int \frac{1}{\beta}\frac{x(\varepsilon)}{1-\tau}dF_\varepsilon(\varepsilon). \tag{48}$$

Intuitively, the planner trades off relaxing the NDS constraint by raising revenue R through inflation and fees against the costs of mining which are deadweight. When doing so, his policy choice leads to an alllocation (x, N) that satisfies the conditions for an optimal offer

$$\sigma\delta^N\varepsilon u'(x) \geq \frac{1}{1-\tau}\left(\frac{\mu}{\beta} - (1-\sigma)\right) \tag{49}$$

$$\frac{\mu}{\beta}\frac{x}{1-\tau} \leq R(N+1)N. \tag{50}$$

[21] Note that $\bar{N}$ could also be a system parameter in our model. We simply impose here that $\bar{N}$ is sufficiently large to allow for sufficiently many confirmation lags in the optimal DS proof contract given for the largest surplus given by the transaction with $\bar{\varepsilon}$.

where the rewards R are generated endogenously by his choice of policy (μ, τ) and the allocation according to condition (48).

The next result on the optimal reward structure is then driven by two considerations. The planner would like to keep mining costs low. At the same time, for any given level of mining costs, he would like to raise the rewards to finance them in a way that is the least distortionary for consumption. The optimal way to finance rewards is using seignorage instead of fees.

Proposition 9. *The optimal reward structure sets transaction fees to zero and only relies on seignorage; i.e.,* $\tau = 0$ *and* $\mu > 1$.

Using fees paid in cryptocurrency is inefficient for two reasons. First, fees imply a larger liquidity cost for buyers. The reason is that all buyers with positive balances are taxed when using inflation, while only buyers who trade in the current period are taxed when fees are used. Hence, the tax base is smaller for fees whenever $\sigma < 1$. To raise the same real revenue when using fees, the planner needs to induce buyers to carry extra liquidity which is costly.

To illustrate this further, hold the amount x consumed by a buyer constant and look at two policies that yield the same real rewards according to equation (48). The first policy uses fees $\bar{\tau}$ and no inflation ($\mu = 1$), while the second policy uses no fees, but only inflation that is set equal to

$$\mu = 1 + \frac{\bar{\tau}\sigma}{1 - \bar{\tau}}. \tag{51}$$

For the first policy, the liquidity cost for a buyer is given by

$$\Phi_1 = \frac{x}{1 - \bar{\tau}} - (1 - \sigma)\beta \frac{x}{1 - \bar{\tau}} \tag{52}$$

where the first term expresses the real value of cryptocurrency acquired in the centralized market while the second gives the real value of the unspent balances in the centralized market next period. Similarly, the liquidity costs when using inflation only is

$$\Phi_2 = \left(1 + \frac{\bar{\tau}\sigma}{1 - \bar{\tau}}\right) x - (1 - \sigma)\beta x. \tag{53}$$

Thus, the liquidity cost associated with fees is larger with the difference given by

$$\Delta_\Phi = (1 - \sigma)(1 - \beta) \left(\frac{\bar{\tau}}{1 - \bar{\tau}}\right) x. \tag{54}$$

Hence, as long as $\sigma < 1$, replacing fees by inflation can lower the costs for carrying liquidity thereby reducing the distortions in the decentralized market.

Second, when double spending a buyer can reclaim his entire payment including fees. Given the mining cost $\beta R/\mu$ per block, the NDS constraint is given by

$$x + \left(\frac{\tau}{1-\tau}\right) x \leq \beta \frac{R}{\mu}(N+1)N. \tag{55}$$

The left hand side of the constraint shows that the buyer can recoup not only his payment to the seller, but also the fees paid to miners. Other things being equal, higher fees make double spending more attractive. This implies that replacing fees by inflation can relax the NDS constraint and reduce the mining cost that is required to support the same amount x to be transacted.[22]

4 The Welfare Costs of Bitcoin

Based on our theoretical analysis, we are now seeking to understand the limits of using cryptocurrencies for payments. We first use the current parameters of Bitcoin and estimate the welfare loss for using Bitcoin to make payments. Changing the key parameters of the reward scheme – seignorage and transaction fees – we then discuss why an additional loss arises relative to a classical monetary economy. Finally, we repeat this comparison by looking a situation where Bitcoin employs the optimal reward structure.

4.1 Calibration

We use aggregate data on Bitcoin transactions from 2015.[23] Table 1 shows summary statistics of Bitcoin usage and transaction fees for the calendar year 2015. We can use these data to calibrate aggregate parameters of the model. To do so, we make the assumption that all transactions are used for payments and, thus, generate surplus. The length of a period in our model is set to be one day. We first calibrate the total volume of transactions σB. In 2015, the average total number of bitcoins in circulation was about $14,342,500$. The average transaction size based on block level data is approximately 2.086659. Using the total number of bitcoins available, we therefore set

[22] In Bitcoin one can indeed recoup the transaction fee when double spending.

[23] This choice is motivated by the fact that the block reward for Bitcoin halved in 2016. Also, Bitcoin experienced extreme price volatility in 2017 and 2018. Hence, usage during this time period is likely to be influenced more by speculative trading of Bitcoin than payments.

Table 1: Bitcoin Transaction Characteristics for 2015 (Source: blockchain.info)

	Per day	Per block
No of transactions	122129.7534	848.1232877
Estimated transaction volume (BTC)	254843.1781	1769.744292
Transaction fees (BTC)	22.45900183	0.15596529

$B = 6,873,428$. This matches the maximum number of average sized transactions that bitcoins in circulation could support in 2015.[24] Next, we set $\sigma = 0.0178$ to match the average fraction of bitcoins spent per day.

The average currency growth rate is set to reflect an increase in the stock of bitcoin over one day. This yields $\mu = (1+25/14342502.95)^{6\times 24} = 1.00025$ per day, which translates into an average annual currency growth rate of about 9.6%.[25] Transaction fees are given by $\tau = 0.15596529/1769.744292 = 0.000088129$ to match the average transaction fee data in Table 1.

To calibrate preferences, we assume that the buyers' utility function is given by

$$u(x) = \varepsilon(\log(x+b) - \log b)$$

where $b = 10^{-60} \approx 0$.[26] The length of a period is a day and the length of each subperiod is equal to the average block time of 10 minutes. We therefore set the discount factor per day β so that the annualized discount factor is equal to 0.97. To find a distribution for ε, we first solve for the buyer's take-it-or-leave-it offer $(N(\varepsilon), x(\varepsilon), d(\varepsilon))$ given the reward function R. This is possible for the calibrated currency growth rate and transaction fees for the average bitcoin transaction. We then use data reported by Ron and Shamir (2013) on the distribution of transaction sizes in Bitcoin to back out the distribution $F(\varepsilon)$ where we have normalized the mean of the distribution to make it consistent with the mean in the 2015 data.

Finally, we assume that the fixed costs ξ_b and ξ_s are large enough given our calibrated parameters

[24] Interestingly, this value is also close to the number of blockchain wallet users which is 5439181 at the end of 2015 (see blockchain.info).

[25] In Bitcoin, the effective currency growth rate decreases with every block as the reward per block is constant except for periodic halving of the reward. Taking this into account explicitly will not influence our quantitative results in any significant way.

[26] As in Lagos and Wright (2005), the constant b is added to normalize utility so that $u(0) = 0$.

Table 2: Benchmark parameters

	values	targets
β	0.999916553598325	period length = 1 day
δ	0.999999420487088	$\delta = \beta^{1/(1+\bar{N})}$
μ	1.00025	currency growth rate
τ	0.000088	transaction fee
B	$6,873,428$	max. no of average-sized transactions
σ	0.0178	velocity per block (block length = 10 mins)

so that the sufficient condition (36) in Lemma 6 is satisfied. This allows us to impose the NDS constraint directly when doing our numerical exercises. We verify later on that in the benchmark and in our comparative statics exercises on the optimal policy, there is no incentive to negotiate for a contract that involves double spending even if those costs are zero. Table 2 summarizes our benchmark parameters.

4.2 Comparative Statics

Figure 6 shows a buyer's equilibrium offer as a function of his type ε given our baseline calibration. It is straightforward to verify that using log utility implies that, when the NDS constraint is not binding, transaction size x is a linear function of ε. Once the NDS constraint becomes binding as x increases, the buyer can increase the confirmation lag. The concavity of $N(\varepsilon)$ confirms that settlement lags relax the NDS constraint according to a power law.

Equilibrium transactions in a cryptocurrency system are inefficient. Relative to the first-best allocation, the equilibrium consumption size is too low and the confirmation lag is too long. To examine these distortions further, we define first $\Delta(\varepsilon)$ as type ε buyer's loss relative to the first-best allocation measured in consumption units

$$\varepsilon u\left((1-\Delta(\varepsilon))x^*(\varepsilon)\right) - x^*(\varepsilon) = \delta^{N(\varepsilon)}\varepsilon u\left(x(\varepsilon)\right) - d(\varepsilon). \tag{56}$$

We also define a related variable $\Delta_\delta(\varepsilon)$ which measures the loss that arises only from the settlement delay $N(\varepsilon)$

$$\varepsilon u\left((1-\Delta_\delta)x^*(\varepsilon)\right) - x^*(\varepsilon) = \delta^{N(\varepsilon)}\varepsilon u\left(x^*(\varepsilon)\right) - x^*(\varepsilon). \tag{57}$$

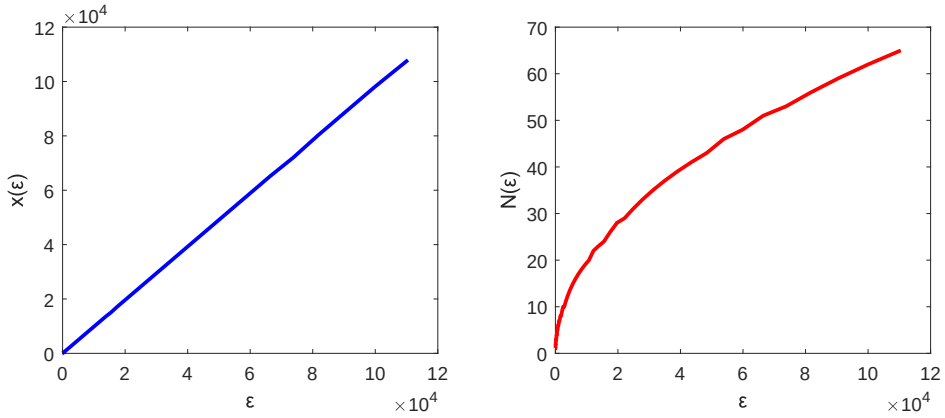


Figure 6: Optimal offer $(x(\varepsilon), N(\varepsilon))$

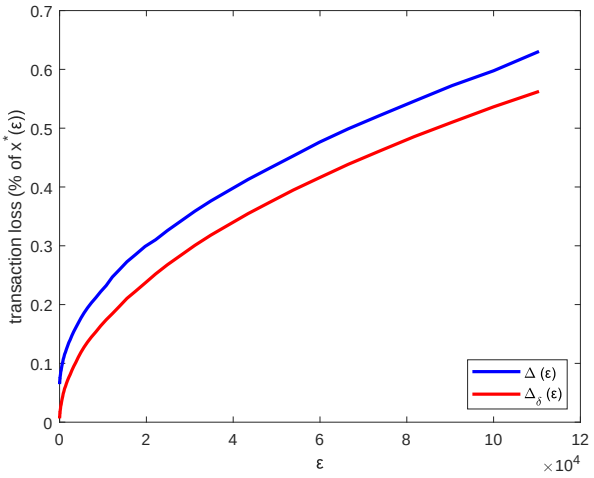


Figure 7: Buyers' loss relative to first-best transactions

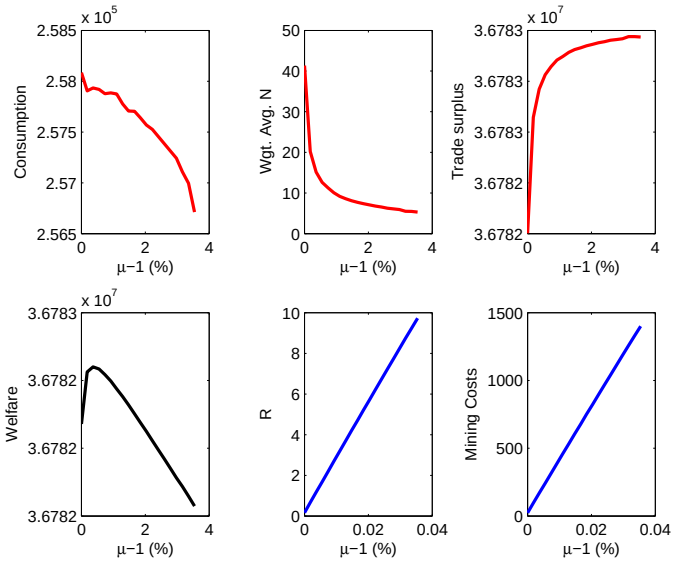


Figure 8: Effects of currency growth

Figure 7 confirms that that welfare loss in a transaction arises from two sources. First, inflation and transaction fees are distortions that reduce the amount transacted relative to the first best allocation. Second, confirmation lags delay settlement and, hence, consumption. Importantly, the loss for individual transactions is increasing in ε. Larger transactions bear a larger loss than smaller transactions with the main source being delayed settlement. While all traders help finance the mining reward, small transactions are subject to shorter confirmation lags since the NDS constraints is less binding.

We next vary the currency growth rate μ in our calibrated economy. When doing so, we fix τ at its calibrated value. This implies that, even without seignorage, sufficient rewards can be generated through transaction fees to satisfy the NDS constraint for all transactions. Figure 8 shows the effects of increasing μ on total welfare $\mathcal{W}$. Interestingly, there is a hump-shaped response of welfare to inflation. To understand this finding, we examine the two components of welfare, total trade surplus and the costs of mining in Figure 8.

The trade surplus first rises as inflation increases above zero which is very different from a classical monetary economy, where inflation reduces trade surplus monotonically. While consumption falls on average due to the inflation tax, transactions are being settled faster as one needs fewer confirmation lags to make transactions double spending proof. This is because higher rewards induce mining activities that help relax the NDS constraint.

The second effect of inflation stems from the costs of mining. Higher rewards financed by currency growth induce miners to spend more resources on mining activities which is socially costly. Overall, the positive effect of faster settlement outweighs the additional mining costs at first for low inflation, but mining costs start to dominate for higher money growth rates. This trade-off generates the hump-shaped response of welfare in Figure 8 resulting in a socially optimal positive inflation rate even in the presence of positive transaction fees.

4.3 Welfare Costs and Optimal Design

As a first step, we derive the welfare costs of Bitcoin associated with different growth rates μ, assuming that transaction fees are adjusted to maximize social welfare. We express the welfare cost as the percentage of consumption Δ_w that people would be willing to give up to move from the cryptocurrency economy to another one supporting the first-best allocation. Denoting $\mathcal{W}(\mu, \tau)$ as the equilibrium welfare of using a cryptocurrency with μ and τ, the welfare cost are then given by Δ_w satisfying

$$\mathcal{W}(\mu, \tau) = B\sigma \int_{\underline{\varepsilon}}^{\bar{\varepsilon}} [\varepsilon u(x^*(\varepsilon)(1 - \Delta_w(\mu, \tau))) - x^*(\varepsilon)] dF(\varepsilon).$$

Figure 9 plots, for different μ, the welfare-maximizing τ as well as the implied welfare costs. The annualized inflation rates on the horizontal axes are reported in log scale. Markers on the curves indicate values corresponding to the actual Bitcoin system as it reduces its block rewards over time. For example, the leftmost marker on a curve denotes the value associated with the Bitocin system in 2015 with a block reward of 25 BTC.

For high block rewards, seignorage is sufficient to make transactions double spending proof. Hence, it is optimal to set transaction fees to zero. Conversely, if seignorage becomes too small, transaction fees need to be strictly positive. This simply reflects the fact that Bitcoin needs to generate rewards from fees as block rewards converge to 0. Figure 9 also suggests that currently most welfare losses in Bitcoin stem from extremely high mining costs, but that these costs will diminish substantially as block rewards fall for Bitcoin over time.

We then compare the current Bitcoin regime with two benchmarks: (i) the optimal Bitcoin regime, and (ii) a cash economy that operates with 2% inflation. Table 3 summarizes our quantitative findings. We first report the welfare costs as a fraction of first-best consumption, Δ_w, as defined above. Furthermore, we consider a hypothetical economy in which people use a traditional cash

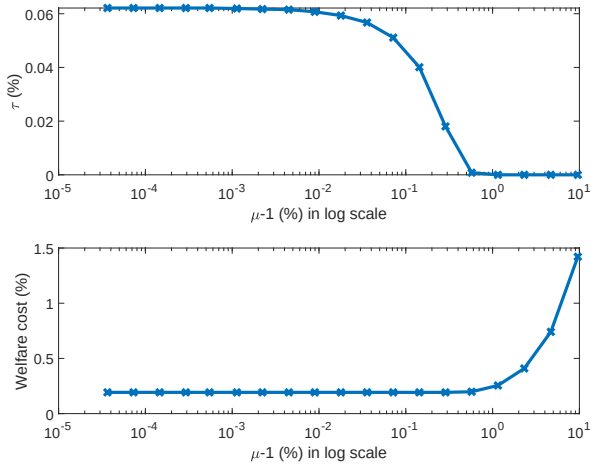


Figure 9: Optimal fee and welfare costs for different μ

system, but face a 2% inflation rate. We calculate the welfare costs of this cash economy and then express the welfare costs of Bitcoin as a multiple of this hypothetical cash economy.

Table 3: Welfare Comparison between Cryptocurrency and Cash Systems

	Bitcoin (benchmark)	Bitcoin (optimal policy)
$\mu - 1$	9.59%	0.40%
τ	0.0088%	0%
welfare costs (% first-best consumption)	1.43%	0.19%
welfare costs ($\times$ 2% cash economy)	478 times	65 times
break-even inflation tax	233.63%	47.58%

In the benchmark regime, the Bitcoin system in 2015 involves a welfare costs equal to 1.43% of first-best consumption. That is, in our model people would be willing to give up 1.43% of their consumption in order to migrate from Bitcoin to an economy that supports the first-best allocation (e.g. cash with the Friedman rule). In contrast, the welfare costs of the cash economy with 2% inflation is only 0.003% of first-best consumption. Hence, according to our model, Bitcoin in 2015 generated a welfare cost almost 500 times bigger than the loss in a 2% cash economy. This welfare cost, however, would shrink significantly in a Bitcoin-like system where rewards are designed optimally. Still, the implied welfare cost in this optimal system are fairly large (65 times of that

in a 2% cash economy). To put this further into perspective, we have calculated a money growth rate of about 48% to equate the welfare cost in a cash economy to the one of an optimal Bitcoin scheme. Even though the optimally designed system has a much lower inflation rate of zero, people would be better off in an economy with inflation below 48% as one avoids the mining costs and the additional distortions that arise from the NDS constraint in the Bitcoin-like system.

We can decompose the welfare costs of an optimal cryptocurrency system into three parts: (i) the deadweight loss from mining; (ii) the cost from a lower level of consumption; and (iii) the cost from a delay in consumption. To do so, we use our expressions in equation (56) and (57) across all transactions. Note that these expressions do not take into account the deadweight losses. The joint total loss of consumption from (ii) and (iii) is approximately 0.044%, with the loss coming solely from delay being 0.027% of the first best consumption. This implies that about 96.9% of the losses are due to the mining costs followed by about 1.94% from delayed settlement. Hence, currently the welfare losses in Bitcoin are mainly related to the costs of mining. Under the optimal policy, however, excessive mining is reduced. As a result, we find that only 32.12% of the welfare losses come from the mining costs.

The deadweight costs of mining are necessary to overcome the double spending problem. Mining, however, is a public good. Once the NDS constraint is satisfied, adding more transactions does not require additional mining activities. This implies that the total volume of transactions matters a lot for how efficient a cryptocurrency can be (see Figure 10 which shows the effect of scaling the number of participants B in our calibration by a factor χ_B).

The intuition for this result is straightforward. Costly mining helps discourage double spending in each transaction, independent of the number of transactions. At the same time, the intensity of mining increases with the total rewards. Hence, with more transactions, it becomes easier to finance mining rewards to protect the system. For the optimal design, this means that seignorage per transaction can fall when there are more transactions. This in turn implies smaller distortions and, thus, smaller welfare costs.

Our analysis also confirms that smaller cryptocurrencies (in terms of market value and transaction volume) can be at risk for double spending attacks as they do not generate enough mining rewards to disincentivize such attacks.[27] Moreover, when the potential gains from a double spending attack

[27]For example, in 2018, a number of small cryptocurrencies such as Monacoin, Bitcoin gold, Zencash, and Litecoin cash were under 51% attacks.

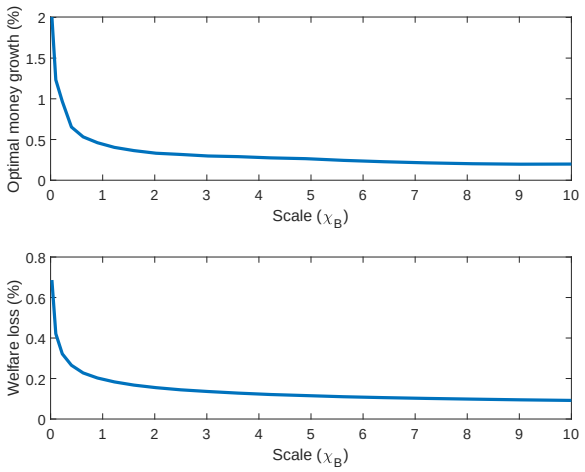


Figure 10: Welfare costs and scale

are small, the mining reward required to protect the system will be lower. This would be the case for a system used only for low value transactions. In conclusion, a cryptocurrency would work best as a retail payment system where there are a large volume of transactions that are relatively small in value. To the contrary, using a cryptocurrency for infrequent, large value payments seems to be very costly.

5 Conclusion

Distributed record-keeping with a blockchain based on consensus through PoW is an intriguing concept. The economics of this technology that underlies most cryptocurrencies are driven by the individual incentives to double spend and the costs associated with reining in these incentives.[28] These costs are both private in the form of delayed settlement and social in the form of mining which is a public good. Consequently, a crytpocurrency becomes more efficient as its scale increases. This explains why a cryptocurrency can avoid double spending only when the user pool is sufficiently large, and why a cryptocurrency works best when the volume of transactions is large relative to

[28] A fundamental issue of a cryptocurrency is to avoid double spending attacks. Nakamoto (2008) states explicitly that "We propose a solution to the double-spending problem using a peer-to-peer network." In practice, double spending – while a first order concern – is not observed with major cryptocurrencies such as Bitcoin. Hence, we think it is not reasonable for a cryptocurrency that involves transactions of significant value to allow for – possibly probabilistic – double spending.

the individual transaction size. This insight seems to be very much ignored in the current debate about the scalability of a cryptocurrency like Bitcoin.

We have assumed that the design of the cryptocurrency specifies transaction fees directly and that there is no congestion (i.e., all transactions can be put into a block immediately). In reality, however, there may be congestion so that users have to compete to get into a block by posting fees. It is straightforward to extend our model to have a restricted blocksize where buyers have an incentive to post fees to have their transaction included into the blockchain (see for example Chiu and Koeppl (2019)). Enriching our model in this direction will not affect the double spending problem and our results on the costs of a cryptocurrency, as the gains from the optimal design of the reward structure are a magnitude smaller than the deadweight costs of mining.

We have also made the simplifying assumption that a double spending attack is a sequence of exponential races. It is more accurate to model a double spending attack as a Poisson race against honest miners where the attacker can catch up if he has fallen behind the honest miners in solving a sequence of PoW problems. There is no closed form solution to modelling optimal decisions in such a Poisson race. We conjecture that our basic results apply to the more general setting once one takes into account the optimal decision of an attacker to incur extra costs of catching up against driving the probability higher.

Finally, a potentially important improvement of PoW is to eliminate inefficient mining activities by changing the consensus protocol altogether. In the Appendix, we explore the possibility of replacing PoW by a Proof-of-Stake (PoS) protocol. Our analysis finds conditions under which PoS can strictly dominate PoW and even support immediate and final settlement. Notwithstanding, as we point out in this Appendix as well, many fundamental issues of a PoS protocol still need to be sorted out. There remains much to be learned about the economic potential and the efficient, economic design of blockchain technology.

References

Auer, R. (2019). "Beyond the doomsday economics of 'proof-of-work' in cryptocurrencies." CEPR Discussion Paper No. DP13506.

Agarwal, R. and M. Kimball. (2015). "Breaking through the zero lower bound." Working Paper WP/15/224, International Monetary Fund.

Aumann, R. J. (1965). "Integrals of set-valued functions." *Journal of Mathematical Analysis and Applications*, 12.1, 1-12.

Berentsen, A. (1997). "Monetary policy implications of digital money." *Kyklos*, 51(1), 89-117.

Biais, B., Bisière, C., Bouvard, M. and C. Cassamatta. (2017). "The blockchain folk theorem." Working paper 817, Toulouse School of Economics.

Biais B, Bisière C., Bouvard M., Casamatta C. and A.J. Menkveld. (2018). "Equilibrium bitcoin pricing." SSRN Working Paper.

Bordo, M. and A. Levin. (2017). "Central bank digital currency and the future of monetary policy", Economics Working Paper 17104, Hoover Institution.

Camera, G. (2017). "A perspective on electronic alternatives to traditional currencies." *Sveriges Riksbank Economic Review*, 2017:1, 126-148.

Chiu, J., and T. Koeppl (2019). "Blockchain-based Settlement for Asset Trading." *Review of Financial Studies*, 32, pp. 1716-1753.

Chiu, J., and T. Wong. (2015). "On the essentiality of e-money." Working Paper 2015-43, Bank of Canada.

Choi, M. and G. Rocheteau. (2019). "Money mining and price dynamics." SSRN 3336367.

Cong, L. W., Z. He, and J. Li. (2019). "Decentralized mining in centralized pools." No. w25592. National Bureau of Economic Research.

Cong, L.W., Li, Y. and N. Wang. (2018). "Tokenomics: dynamic adoption and valuation." Working Paper.

Easley, D., M. O'Hara, and S. Basu. (2019). "From mining to markets: The evolution of bitcoin transaction fees." *Journal of Financial Economics*.

Eyal, I. and E. G. Sirer. (2018). "Majority is not enough: Bitcoin mining is vulnerable." Communications of the ACM 61.7: 95-102.

Fernández-Villaverde, J. and D. Sanches. (2016). "Can currency competition work?", NBER Working Paper 22157, National Bureau of Economic Research.

Gans, J., and H. Halaburda. (2013). "Some economics of private digital currency." Working Paper 2013-38, Bank of Canada.

Huberman, G., Leshno, J. and C. Moallemi. (2017). "Monopoly without a monopolist: An Economic Analysis of the Bitcoin Payment System." Working Paper 17-92, Columbia Business School.

Koeppl, T., Monnet, C. and T. Temzelides. (2008). "A dynamic model of settlement." *Journal of Economic Theory*, 142, pp. 233-246.

Koeppl, T., Monnet, C. and T. Temzelides. (2012). "Optimal clearing arrangements for financial trades." *Journal of Financial Economics*, 103, pp. 189-203.

Lagos, L. and R. Wright. (2005). "A unified framework for monetary theory and policy analysis." *Journal of Political Economy*, 113, pp. 463-484.

Nakamoto, S. (2008). "Bitcoin: A peer-to-peer electronic cash system." Manuscript.

Pagnotta, E. (2018). "Bitcoin as decentralized money: prices, mining, and network security." SSRN Working paper.

Rogoff, K.S., (2016). *The Curse of Cash*, Princeton University Press.

Ron, D. and A. Shamir. (2013). "Quantitative analysis of the full bitcoin transaction graph", International Conference on Financial Cryptography and Data Security, pp. 6-24.

Rosenfeld, M. (2014). "Analysis of hashrate-based double spending." arXiv preprint, arXiv:1402.2009.

Saleh, F. (2018). "Blockchain without waste: Proof-of-stake." Manuscript.

Schilling, L., and H. Uhlig. (2018). "Some simple bitcoin economics." No. w24483. National Bureau of Economic Research.

A Appendix – Proofs

A.1 Proof of Lemma 2

We first look at the incentives to stop and continue with secret mining. Note first, by assumption, if the buyer fails to win the mining competition the double spending attempt fails as well. It is then optimal for the buyer to stop mining. Next, we show that – having started – it is never optimal for a buyer to stop a double spending attack if successful in the first n steps.

Lemma A.1. *If double spending with $(q_0, ..., q_N)$ generates a positive payoff, then the buyer has no incentives to quit mining after winning a block in $n < N$.*

Proof. Consider first the case with $N = 2$. We have

$$D_0 = \rho_0\rho_1\rho_2\frac{\beta}{\mu}[d + 3R] - \rho_0\rho_1 q_2 - \rho_0 q_1 - q_0 = \rho_0\left[\rho_1\left[\rho_2\frac{\beta}{\mu}[d + 3R] - q_2\right] - q_1\right] - q_0. \tag{A.1}$$

Define $D_1 = \rho_1\left[\rho_2\frac{\beta}{\mu}[d + 3R] - q_2\right] - q_1$. Since

$$D_0 = \rho_0 D_1 - q_0 > 0 \tag{A.2}$$

we have

$$\rho_1\left[\rho_2\frac{\beta}{\mu}[d + 3R] - q_2\right] - q_1 > \frac{\beta}{\mu}R. \tag{A.3}$$

Similarly, define $D_2 = \rho_2\frac{\beta}{\mu}[d + 3R] - q_2$. Inequality (A.3) implies that

$$\rho_1 D_2 - q_1 > \frac{\beta}{\mu}R, \tag{A.4}$$

so that $D_2 > 2\frac{\beta}{\mu}R$ or

$$\rho_2\frac{\beta}{\mu}[d + R(1 + N)] - q_2 > 2\frac{\beta}{\mu}R. \tag{A.5}$$

Inequality (A.5) then implies that

$$\rho_2\frac{\beta}{\mu}[d + 3R] - q_2 + u(x) > 2\frac{\beta}{\mu}R + \frac{\beta}{\mu}d \tag{A.6}$$

where the left hand side is the expected payoff of continuing with secret mining in subperiod 2 and the RHS is the payoff of stopping with secret mining and announcing the two blocks. Hence, there is no incentive to stop with secret mining after having found two blocks.

Similarly, condition (A.3) implies that

$$\rho_1 \left[\rho_2 \frac{\beta}{\mu}[d + 3R] - q_2 \right] - q_1 + \delta u(x) > \frac{\beta}{\mu} R + \frac{\beta}{\mu} d \tag{A.7}$$

which shows that there is no incentive to stop with secret mining in subperiod 1 having been successful in the first step.

For a general N, we can show similarly that

$$D_N = \rho_N \frac{\beta}{\mu}[d + R(1 + N)] - q_N > \frac{\beta}{\mu} NR \tag{A.8}$$

$$D_{N-1} = \rho_{N-1} D_N - q_{N-1} > \frac{\beta}{\mu}(N - 1)R \tag{A.9}$$

$$\vdots \tag{A.10}$$

$$D_1 = \rho_1 D_2 - q_1 > \frac{\beta}{\mu} R \tag{A.11}$$

$$D_0 = \rho_0 D_1 - q_0 > 0. \tag{A.12}$$

It follows that

$$\rho_N \frac{\beta}{\mu}[d + R(1 + N)] - q_N + u(x) > \frac{\beta}{\mu}(NR + d), \tag{A.13}$$

$$\rho_{N-1} D_N - q_{N-1} + \delta u(x) > \frac{\beta}{\mu}[(N - 1)R + d] \tag{A.14}$$

$$\vdots \tag{A.15}$$

$$\rho_1 D_2 - q_1 + \delta^{N-1} u(x) > \frac{\beta}{\mu}(R + d). \tag{A.16}$$

Hence, there is no incentives to quit after winning having found n blocks in a row. □

The remainder of the proof is done by induction. The result is true for $s = 0$.

Suppose then the result holds true for $s = n - 1$. For $s = n$, we obtain

$$q_{N-n}(d, N) = \sqrt{MQ \cdot D_{N-n+1}(d, N)} - MQ \tag{A.17}$$

$$= \beta R(\sqrt{\Delta} - n) - \beta R \tag{A.18}$$

$$= \beta R[\sqrt{\Delta} - (n + 1)] \tag{A.19}$$

for the optimal investment into mining,

$$\rho_{N-n}(d, N) = \frac{q_{N-n}(d, N)}{MQ + q_{N-n}(d, N)} \tag{A.20}$$

$$= \frac{\sqrt{\Delta} - (n + 1)}{\sqrt{\Delta} - n} \tag{A.21}$$

for the probability of finding a block first and

$$D_{N-n}(d,N) = \rho_{N-n}(d,N)D_{N-n+1}(d,N) - q_{N-n}(d,N) \tag{A.22}$$

$$= \frac{\sqrt{\Delta}-(n+1)}{\sqrt{\Delta}-n}\beta R(\sqrt{\Delta}-n)^2 - \beta R[\sqrt{\Delta}-(n+1)] \tag{A.23}$$

$$= \beta R[(\sqrt{\Delta}-(n+1)]^2 \tag{A.24}$$

for the expected value from secret mining. This completes the proof.

A.2 Proof of Lemma 6

Let $(\hat{x},\hat{d},\hat{N})$ be a contract with double spending for a buyer with ε. To compensate the seller, the buyer needs to offer a payment $\hat{d} = \frac{\mu}{\beta}\frac{1}{1-P}\frac{\hat{x}+\xi_s}{(1-\tau)}$. The payment reimburses the seller for his fixed cost ξ_s and the expected loss from double spending.

Consider now a new contract (x',d',N') where

$$x' = \hat{x} \tag{A.25}$$

$$d' = \frac{\hat{x}}{1-\tau}\frac{\mu}{\beta} \tag{A.26}$$

$$N' = \min\{N : \frac{\hat{x}}{1-\tau}\frac{\mu}{\beta} \leq R(N+1)N\}. \tag{A.27}$$

By construction, this contract satisfies the NDS constraint. Furthermore, such a contract is feasible, since the buyer never wants to consume more than the first-best quantity $x^*(\varepsilon) \leq x^*(\varepsilon_{\max})$ and where we assume that $\frac{x^*(\varepsilon_{\max})}{1-\tau}\frac{\mu}{\beta} \leq R(\bar{N}+1)\bar{N}$.

The expected payoff of the contract $(\hat{x},\hat{d},\hat{N})$ for the buyer is given by

$$V(\hat{x},\hat{d},\hat{N}) = -\hat{d} + (1-\sigma)\frac{\beta}{\mu}\hat{d} + \sigma\left(\delta^{\hat{N}}\varepsilon u(\hat{x}) + D_0(\hat{d},\hat{N}) - \xi_b\right) \tag{A.28}$$

$$\leq -\hat{d} + (1-\sigma)\frac{\beta}{\mu}\hat{d} + \sigma\left(\delta^{\hat{N}}\varepsilon u(\hat{x}) + P\beta\frac{\hat{d}}{\mu} - \xi_b\right) \tag{A.29}$$

$$= \hat{d}\left(\frac{\beta}{\mu}-1\right) + \sigma\left(\delta^{\hat{N}}\varepsilon u(\hat{x}) - (1-P)\beta\frac{\hat{d}}{\mu}\right) - \sigma\xi_b \tag{A.30}$$

$$= \frac{\mu}{\beta}\left(\frac{1}{1-P}\right)\left(\frac{\hat{x}+\xi_s}{1-\tau}\right)\left(\frac{\beta}{\mu}-1\right) + \sigma\left(\delta^{\hat{N}}\varepsilon u(\hat{x}) - \frac{\hat{x}+\xi_s}{1-\tau}\right) - \sigma\xi_b. \tag{A.31}$$

The weak inequality follows from the fact that miners make zero profits and, thus, a secret miner can make at most zero profits from obtaining the mining rewards and paying the mining costs.

Hence, his expected profit can be at most the additional expected reward $P\hat{d}(\beta/\mu)$ from reclaiming the payment next period.

The expected payoff from the contract (x', d', N') that satisfies the NDS constraint is given by

$$V(x', d', N') = d' \left(\frac{\beta}{\mu} - 1\right) + \sigma \left(\delta^{N'} \varepsilon u(x') - \beta \frac{d'}{\mu}\right) \tag{A.32}$$

$$= \frac{\mu}{\beta} \left(\frac{\hat{x}}{1-\tau}\right) \left(\frac{\beta}{\mu} - 1\right) + \sigma \left(\delta^{N'} \varepsilon u(\hat{x}) - \frac{\hat{x}}{1-\tau}\right) \tag{A.33}$$

Therefore, this new contract is better for the buyer if

$$\frac{\mu}{\beta} \left(\frac{\hat{x}}{1-\tau}\right) \left(\frac{\beta}{\mu} - 1\right) + \sigma \left(\delta^{N'} \varepsilon u(\hat{x}) - \frac{\hat{x}}{1-\tau}\right) \tag{A.34}$$

$$> \frac{\mu}{\beta} \left(\frac{1}{1-P}\right) \left(\frac{\hat{x} + \xi_s}{1-\tau}\right) \left(\frac{\beta}{\mu} - 1\right) + \sigma \left(\delta^{\hat{N}} \varepsilon u(\hat{x}) - \frac{\hat{x} + \xi_s}{1-\tau}\right) - \sigma \xi_b \tag{A.35}$$

or

$$\sigma \delta^{N'} \varepsilon u(\hat{x}) > \frac{\mu}{\beta} \left(\frac{\xi_s}{(1-\tau)}\right) \left(\frac{\beta}{\mu} - 1\right) + \sigma \left(\delta^{\hat{N}} \varepsilon u(\hat{x}) - \frac{\xi_s}{1-\tau}\right) - \sigma \xi_b \tag{A.36}$$

where the last inequality follows from the fact that $\mu \geq 1 > \beta$. This reduces to

$$\xi_b + \xi_s \left(1 + \frac{1-\beta}{\sigma\beta}\right) > \varepsilon u(\hat{x}) \left(\delta^{\hat{N}} - \delta^{N'}\right). \tag{A.37}$$

The sufficient condition then implies that the new contract is better for any buyer with type ε, since

$$\varepsilon_{\max} u(x^*(\varepsilon_{\max}))(1-\beta) \geq \varepsilon u(x^*(\varepsilon))(1-\beta) > \varepsilon u(\hat{x}(\varepsilon)) \left(\delta^{\hat{N}} - \delta^{N'}\right). \tag{A.38}$$

A.3 Proof of Proposition 8

Given (μ, τ), define the correspondence

$$\varphi(R) = B \left(\frac{\mu - 1 + \sigma\tau}{\bar{N} + 1}\right) \int z(\varepsilon; R) dF(\varepsilon) \tag{A.39}$$

where $\int z(\varepsilon; R) dF(\varepsilon)$ is aggregate money demand. The correspondence φ expresses total rewards for mining as a function of aggregate money demand which itself is a function of R through the NDS constraint (20). Since $z(\varepsilon; R)$ is a correspondence, it is useful to define $z_n(\varepsilon; R)$ as the optimal money demand conditional on confirmation lag $n \in \{1, \ldots, \bar{N}\}$. Note that $z_n(\varepsilon; R)$ is a function that is continuous by the Theorem of the Maximum and weakly increasing in R.

We will show that the correspondence φ has a fixed point for some R by using Kakutani's fixed point theorem. This implies that (i) we need to restrict the mapping φ to a non-empty, convex and compact set, and (ii) we need to show that φ is non-empty, upper hemicontinuous, convex-valued and closed-valued. To restrict the mapping to a compact interval, we start out by choosing the number of buyers B large enough for any R so that $\varphi(R) > R$. Then, we show that for any B, we can bound the value of $\varphi(R)$.

Lemma A.2. *For $R_{\min} > 0$, there exists a sufficiently large B such that $\varphi(R) \geq R_{\min}$ for all $R \geq R_{\min}$.*

Proof. Note first that for $R > 0$, $z_n(\varepsilon; R) > 0$ for all ε. For any $R_{\min} > 0$, we can then find B sufficiently large so that

$$B\left(\frac{\mu - 1 + \sigma\tau}{\bar{N}+1}\right)\int \min_n z_n(\varepsilon; R_{\min})dF(\varepsilon) > R_{\min} \tag{A.40}$$

Since $z_n(\varepsilon; R)$ is weakly increasing in R, we have that

$$\varphi(R) = B\left(\frac{\mu - 1 + \sigma\tau}{\bar{N}+1}\right)\int z(\varepsilon; R)dF(\varepsilon) \geq B\left(\frac{\mu - 1 + \sigma\tau}{\bar{N}+1}\right)\int \min_n z_n(\varepsilon; R_{\min})dF(\varepsilon) \tag{A.41}$$

for all $R \geq R_{\min}$. □

Lemma A.3. *Given B, there exists $R_{\max}$ such that $\varphi(R) \leq R_{\max}$ for all $R \leq R_{\max}$.*

Proof. The surplus from real balances z is bounded by

$$\sigma(\varepsilon u(x^*) - x^*) - z + (1-\sigma)\frac{\beta}{\mu}z \tag{A.42}$$

where x^* satisfies $\varepsilon u'(x^*) = 1$. Hence, there is an upper bound $\bar{z}(\varepsilon)$ on money demand that is independent of R. Thus, given B, for all $R \leq R_{\max}$, we have that

$$\varphi(R) = B\left(\frac{\mu - 1 + \sigma\tau}{\bar{N}+1}\right)\int z(\varepsilon; R)dF(\varepsilon) \leq B\left(\frac{\mu - 1 + \sigma\tau}{\bar{N}+1}\right)\int \bar{z}(\varepsilon)dF(\varepsilon) = R_{\max}. \tag{A.43}$$

□

Hence, for any $R_{\min} > 0$, there exists a sufficiently large B and $R_{\max}$ such that $R_{\min} < \varphi(R) \leq R_{\max}$. This allows us to restrict the correspondence φ to a compact interval $[R_{\min}, R_{\max}]$ where

$\varphi(R) : [R_{\min}, R_{\max}] \to [R_{\min}, R_{\max}]$, whenever B is large enough. We turn next to the properties of the correspondence φ.

Lemma A.4. *The money demand correspondence $z(\varepsilon; R)$ is non-empty, compact-valued and upper hemicontinuous.*

Proof. Take the optimal money demand functions conditional on n and form the constraint correspondence

$$\mathcal{Z} = \bigcup_{n=1}^{\bar{N}} z_n(\varepsilon; R). \tag{A.44}$$

Note that this correspondence is both upper and lower hemicontinuous and, hence, continuous. Furthermore, it is compact-valued as it is comprised of a finite number of values.

Let $V_n(R)$ be the utility associated with $z_n(\varepsilon; R)$. Then, the optimal money demand is given by a solution to the problem

$$\max_n V_n(R) \tag{A.45}$$

subject to

$$z_n(R) \in \mathcal{Z}. \tag{A.46}$$

By the Theorem of the Maximum, we have that the correspondence of the maximizer $z(\varepsilon; R)$ is non-empty and upper hemicontinuous. Since it comprises at most a finite number of elements, it is also compact-valued. □

It follows immediately that the correspondence $\varphi(R)$ is also non-empty, compact-valued and convex-valued (see Aumann (1965), Theorem 1, 2 and 4). Furthermore, $\varphi(R)$ is upper hemicontinuous (see Aumann (1965), Corollary 5.2). By Kakutani's Fixed Point theorem, there exists $R^* \in [R_{\min}, R_{\max}]$ such that

$$R^* \in \varphi(R^*) \tag{A.47}$$

which completes the proof.

A.4 Proof of Proposition 9

The idea of the proof is as follows. If fees are not zero, one can design a new policy that (i) uses less mining costs and (ii) makes all buyers better off. The policy change will combine three changes.

First, it reduces transaction fees to zero. Second, it will increase the seignorage rate to compensate for the loss in the expected revenue from fees taking into account the different tax base for the two taxes. Third, it will generate at least the same rewards for miners.

The proof relies on the fact that the planner can generate more revenue than the amount of rewards paid out to miners. Hence, we assume for now that the revenue constraint (48) is a weak inequality. This implies that the planner makes a distinction between the *revenue raised* and the *rewards paid* to miners. To this end, we define a policy to be (μ, τ, R). In the last step of the proof, we will show that this assumption is innocuous.

Step 1

Suppose that $\tau_0 > 0$, $\mu_0 \geq 1$ and R_0 is a feasible policy. Consider the following policy change

$$\tau_1 = \tau_0 - \xi \tag{A.48}$$

$$\mu_1 = \mu_0 + \sigma\xi \tag{A.49}$$

$$R_1 = R_0 \frac{1-\tau_0}{1-\tau_1}\frac{\mu_1}{\mu_0} \tag{A.50}$$

Note that $\tau_0 \geq \xi > 0$ is arbitrary and, thus, we can set it to τ_0. We denote the optimal choice for each policy by $(x_i^*(\varepsilon), N_i^*(\varepsilon))$ and show first that this policy change is feasible, if the transaction size increases in all bilateral meetings.

Lemma A.5. *The policy change is feasible if $x_1^*(\varepsilon) \geq x_0^*(\varepsilon)$ for all ε.*

Proof. For any $n \in \{1, \ldots, \bar{N}\}$, the no double spending constraint at the old policy is given by

$$\frac{\mu_0}{\beta}\frac{x(\varepsilon)}{1-\tau_0} \leq R_0(N(\varepsilon)+1)N(\varepsilon). \tag{A.51}$$

Using the definition of R_1, it is immediate that – by construction – the NDS constraints are identical for all $n \in \{1, \ldots, \bar{N}\}$ for the new policy. Hence, the set of feasible choices (x, N) remains unchanged.

Next, we show that the revenue constraint is also fulfilled as long as the optimal individual trans-

action sizes do not decreases. We have

$$B(\mu_1 - 1 + \sigma\tau_1) \int \frac{x_1^*(\varepsilon)}{1-\tau_1} dF_\varepsilon(\varepsilon) \tag{A.52}$$

$$\geq B(\mu_0 + \sigma\xi - 1 + \sigma\tau_0 - \sigma\xi) \int \frac{x_0^*(\varepsilon)}{1-\tau_1} dF_\varepsilon(\varepsilon) \tag{A.53}$$

$$\geq \frac{\beta}{\mu_1} R_0 \frac{1-\tau_0}{1-\tau_1} \frac{\mu_1}{\mu_0} (\bar{N}+1) \tag{A.54}$$

$$= \frac{\beta}{\mu_1} R_1 (\bar{N}+1) \tag{A.55}$$

which completes the proof. □

Step 2

Next, we show that the policy change increases the planner's objective as long as consumption and confirmation lags weakly increase for all ε.

Lemma A.6. *If $x_1^*(\varepsilon) \geq x_0^*(\varepsilon)$ and $N_1^*(\varepsilon) \geq N_0^*(\varepsilon)$, the policy change increases the planner's objective function.*

Proof. First, mining costs decrease since we have

$$\frac{\beta}{\mu_1} R_1 (\bar{N}+1) = \frac{\beta}{\mu_0} R_0 \frac{1-\tau_0}{1-\tau_0+\xi} (\bar{N}+1) < \frac{\beta}{\mu_0} R_0 (\bar{N}+1). \tag{A.56}$$

Second, we can rewrite the trade surplus in the decentralized market for each ε as

$$\begin{aligned} \sigma\left(\delta^{N_1^*} \varepsilon u(x_1^*) - x_1^*\right) = \sigma\delta^{N_1^*} \varepsilon u(x_1^*) - \frac{x_1^*}{1-\tau_1}\frac{\mu_1}{\beta} + (1-\sigma)\frac{x_1^*}{1-\tau_1} \\ - \sigma x_1^* + \frac{x_1^*}{1-\tau_1}\frac{\mu_1}{\beta} - (1-\sigma)\frac{x_1^*}{1-\tau_1} \end{aligned} \tag{A.57}$$

where the first three terms correspond to the buyer's objective function. Since the buyer's choice is optimal, we obtain using the conditions in the lemma that

$$\sigma\left(\delta^{N_1^*} \varepsilon u(x_1^*) - x_1^*\right)$$

$$\geq \sigma\delta^{N_0^*} \varepsilon u(x_0^*) - \frac{x_0^*}{1-\tau_1}\frac{\mu_1}{\beta} + (1-\sigma)\frac{x_0^*}{1-\tau_1} - \sigma x_1^* + \frac{x_1^*}{1-\tau_1}\frac{\mu_1}{\beta} - (1-\sigma)\frac{x_1^*}{1-\tau_1} \tag{A.58}$$

$$= \sigma\left(\delta^{N_0^*} \varepsilon u(x_0^*) - x_0^*\right) + \frac{1}{1-\tau_1}\left(\frac{\mu_1}{\beta} - 1 + \sigma\tau_1\right)(x_1^* - x_0^*) \tag{A.59}$$

$$\geq \sigma\left(\delta^{N_0^*} \varepsilon u(x_0) - \sigma x_0^*\right) \tag{A.60}$$

which completes the proof. □

Step 3

We are left to show that that it is weakly optimal for the buyer to increase both, the optimal size of the transaction and the confirmation lag. Define $x^*(n)$ as the buyer's optimal choice of x given a fixed confirmation lag n. We first show that, for any given n, the optimal choice $x^*(n)$ is weakly increasing in the policy change and then show that the buyer is never better off lowering the optimal confirmation lag N^* when the policy changes.

Lemma A.7. *Given n, the optimal choice $x^*(n)$ is weakly increasing in the policy change for all ε.*

Proof. Define $x^*(n)$ as the optimal quantity chosen by the buyer for any given n, that is

$$x^*(n) = \arg\max_x -\frac{x}{1-\tau}\frac{\mu}{\beta} + (1-\sigma)\frac{x}{1-\tau} + \sigma\delta^n\varepsilon u(x) \tag{A.61}$$

subject to

$$\frac{x}{1-\tau}\frac{\mu}{\beta} \leq R(n+1)n. \tag{A.62}$$

The first-order condition is given by

$$\sigma\delta^n\varepsilon u'(x) \geq \frac{1}{1-\tau}\left(\frac{\mu}{\beta} - (1-\sigma)\right) \tag{A.63}$$

with equality when the NDS constraint (A.62) for n is not binding.

We have that

$$\begin{aligned}
&\left(\frac{\mu_0}{\beta} - (1-\sigma)\right)(1-\tau_1) - \left(\frac{\mu_1}{\beta} - (1-\sigma)\right)(1-\tau_0) \\
&\quad = \left(\frac{\mu_0}{\beta} - (1-\sigma)\right)(1-\tau_0+\xi) - \left(\frac{\mu_0+\sigma\xi}{\beta} - (1-\sigma)\right)(1-\tau_0) && \text{(A.64)} \\
&\quad = \frac{\mu_0}{\beta}\xi - (1-\sigma)\xi - (1-\tau_0)\frac{\sigma}{\beta}\xi && \text{(A.65)} \\
&\quad = \frac{\xi}{\beta}\left(\mu_0 - \beta(1-\sigma) - \sigma(1-\tau_0)\right) && \text{(A.66)} \\
&\quad > \xi(\mu_0 - 1) > 0. && \text{(A.67)}
\end{aligned}$$

Hence, the right hand side of the first order condition is decreasing in the policy change so that, for any given n, the optimal choice is $x_1^*(n) \geq x_0^*(n)$. □

Lemma A.8. *For all ε, we have $N_1^*(\varepsilon) \geq N *_0 (\varepsilon)$.*

Proof. We first show that for the optimal offer all NDS constraints must be binding for $n < N^*$ and that $x^*(n) < x^*(N^*)$ for all $n < N^*$.

Suppose to the contrary that the No-DS constraint is not binding for some $n < N^*$. Denoting the optimal choice conditional on n by $x^*(n)$, we then have that

$$\sigma\delta^{N^*}\varepsilon u'(x^*(N^*)) \geq \sigma\delta^n\varepsilon u'(x^*(n)) = \left(\frac{\mu}{\beta} - (1-\sigma)\right)\frac{1}{1-\tau} \tag{A.68}$$

with strict inequality if the No-DS constraint is binding at N^*.

Since $n < N^*$, it must be the case that

$$x^*(n) > x^*(N^*). \tag{A.69}$$

But this implies that $x^*(N^*)$ is feasible with a confirmation lag n. Hence, N^* cannot be optimal, since choosing $(x^*(N^*), n)$ is feasible, has the same cost, but earlier consumption. A contradiction.

By a similar argument, it is then straightforward to show that $x^*(n) < x^*(N^*)$, since otherwise $x^*(N^*)$ would be feasible with n and yield a higher utility than confirmation lag N^*.

With the policy change, we have that for all $n < N_0^*$ the NDS constraints are still binding. Hence, the optimal choice is given by $x_1^*(n) = x_0^*(n)$ for $n < N_0^*$. We have

$$\sigma\delta^{N_0}\varepsilon u(x_0^*) - \sigma\delta^n\varepsilon u(x_0^*(n)) \geq (x_0^* - x_0^*(n))\left(\frac{1}{1-\tau_0}\right)\left(\frac{\mu_0}{\beta} - (1-\sigma)\right) \tag{A.70}$$

$$> (x_0^* - x_0^*(n))\left(\frac{1}{1-\tau_1}\right)\left(\frac{\mu_1}{\beta} - (1-\sigma)\right) \tag{A.71}$$

since $x_0^* > x_0^*(n)$ and the last term is decreasing with the policy change. This implies that the original choice at N_0 still yields higher utility than the optimal choice for $n < N_0$ after the policy change. Hence, the optimal choice must satisfy $N_1^* \geq N_0^*$ which completes the proof. □

Step 4

We are left to show that a binding feasibility constraint is optimal for the planner. Set $\tau = 0$ and suppose that the revenue constraint is not binding. Consider now a policy change according to

$$\mu_1 = \mu_0 - \xi \tag{A.72}$$

$$R_1 = R_0\frac{\mu_1}{\mu_0}. \tag{A.73}$$

It follows immediately that Lemma A.5, since the revenue constraint was slack at $(\mu_0, 0, R_0)$. Similarly, Lemma A.6 goes through except for that the mining costs remain constant with the change now. Finally, nothing in the remainder of the proof is affected by keeping $\tau = 0$, since larger transaction sizes become cheaper with lower inflation. This implies that the proposed policy change will increase the planner's objective function until the feasibility constraint will eventually become binding for μ sufficiently close to 1.

B Appendix [FOR ONLINE PUBLICATION ONLY] – A Stylized Introduction to the Double Spending Problem

Our modern economy relies heavily on digital means of payments. Trade in the form of e-commerce for example necessitates the usage of digital tokens. In a digital currency system, the means of payment is simply a string of bits. This poses a problem, as these strings of bits as any other digital record can easily be copied and re-used for payment. Essentially, the digital token can be counterfeited by using it twice which is the so-called double-spending problem.

Traditionally, this problem has been overcome by relying on a trusted third-party who manages for a fee a centralized ledger and transfers balances by crediting and debiting buyers and sellers' accounts. This third-party is often the issuer of the digital currency itself, one prominent example being PayPal, and the value of the currency derives from the fact that users trust the third-party to prohibit double-spending (top of Figure 11).

Cryptocurrencies such as Bitcoin go a step further and remove the need for a trusted third-party. Instead, they rely on a decentralized network of (possibly anonymous) validators to maintain and update copies of the ledger (bottom of Figure 11). This necessitates that consensus between the validators is maintained about the correct record of transactions so that the users can be sure to receive and keep ownership of balances. But such a consensus ultimately requires that (i) users do not double-spend the currency and (ii) that users can trust the validators to accurately update the ledger.

How do cryptocurrencies such as Bitcoin tackle these challenges? Trust in the currency is based on a *blockchain* which ensures the distributed verification, updating and storage of the record of transaction histories.[29] This is done by forming a blockchain. A block is a set of transactions that have been conducted between the users of the cryptocurrency. A chain is created from these blocks containing the history of past transactions that allows one to create a ledger where one can publicly verify the amount of balances or currency a user owns. Hence, a blockchain is like a book containing the ledger of all past transactions with a block being a new page recording all the current transactions.

[29] In this sense it is different from traditional money which is merely a partial memory of past transactions as it only records the current distribution of balances and does not record how past transactions generate the current distribution.

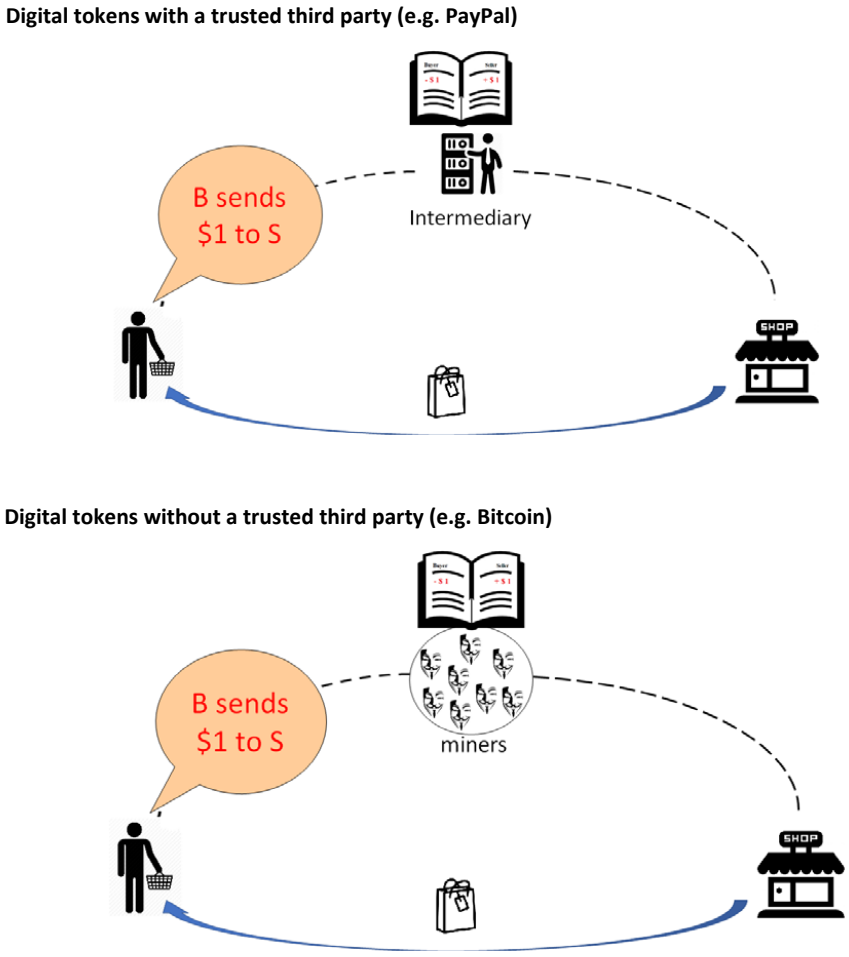


Figure 11: Digital Currency vs. Cryptocurrency

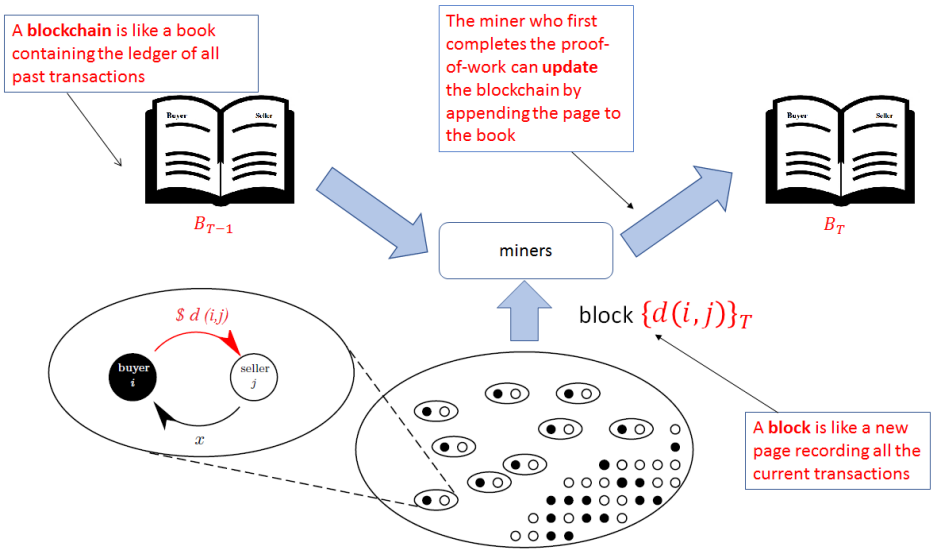


Figure 12: Blockchain Based Validation in a Cryptocurrency

Figure 12 illustrates how the blockchain is updated. To ensure consensus, validators compete for the right to update the chain with a new block. This competition can take various forms. In Bitcoin, it happens through a process called *mining*. Miners (i.e. transaction validators) compete to solve a computationally costly problem which is called *proof-of-work* (PoW).[30] The winner of this mining process has the right to update the chain with a new block. The consensus protocol prescribes then that the "longest" history will be accepted as the trusted public record.[31] Since transaction validation and mining are costly, a reward structure is needed for mining to take place. In Bitcoin, for example such rewards are currently financed by the creation of new coins and transaction fees.[32]

The main concern for users when trusting a cryptocurrency is the double spending problem: after having conducted a transaction, a user attempts to convince the validators (and, hence, the general public if the blockchain is trusted) to accept an alternative history in which some payment was

[30] Other consensus protocols are being explored which we briefly discuss in the Appendix.

[31] In general, there is no explicit requirement to follow the consensus protocol in the sense that validators and users can trust an alternative history or blockchain that is not the longest one. Well-designed cryptocurrency, however, try to ensure that there are sufficient incentives to work with the longest history. For example, in Bitcoin the reward paid to a successful miner is contained in the new block itself. Should a different chain be adopted at a later stage, this reward will be obsolete. For a game-theoretic analysis of the incentives to build on the longest chain, see Biais et. al. (2017).

[32] Huberman et al. (2017) explore the reward structure of cryptocurrencies from the perspective of the mining game, but without modelling the double spending problem.

not conducted.[33] If this attack succeeds, this user will keep both the balances and the product or service he obtained while the counterparty will be left empty handed. Hence, the possibility of such double-spending can undermine the trust in the cryptocurrency.

A blockchain based on a PoW consensus protocol naturally deals with changing transaction history backwards. The blockchain has to be dynamically consistent in the sense that current transactions have to be linked to transactions in all previous blocks.[34] Consequently, if a person attempts to revoke a transaction in the past, he has to propose an alternative blockchain (with that particular transaction removed) and perform the PoW for each of the newly proposed block. Therefore, it is very costly to rewrite the history of transactions backwards if the part of the chain that needs to be replaced is long. Hence, the "older" transactions are, the more users can trust them.

Unfortunately, a blockchain does not automatically protect a cryptocurrency against a double-spending attack that is forward-looking. Figure 13 considers a spot trade between a buyer and a seller involving a cryptocurrency. The buyer instructs the miners to transfer a payment to the seller while the seller simultaneously delivers the goods. Notice that the buyer can always secretly mine an alternative history (or submit to some miners a different history) in which the fund is not transferred. The final outcome of the transaction depends on which payment instruction is incorporated into the blockchain first. If the former payment instruction is incorporated, then the double-spending attempt fails. The seller receives the payment and the buyer gets the goods. If the latter is accepted instead, then the double-spending attempt succeeds. In this case, the buyer gets the goods without paying the seller.

Such a double spending attack can be discouraged by introducing a confirmation lag into the transactions. By waiting some blocks before completing the transaction (i.e., the seller delays the delivery of the goods), it becomes harder to alter transactions in a sequence of new blocks. Figure 14 illustrates how a confirmation lag of one block confirmation raises the secret mining burden of a double spender. The seller delivers the goods only after the payment is incorporated into the blockchain at least in one new block. Again, the buyer can secretly mine an alternative history in which the payment does not happen. How successful secret mining is depends on the mining

[33] While basic cryptography ensures that people cannot spend others' balances, a miner can exclude from a block some transactions that have been initiated by other people. With positive transaction fees, a miner does not have an incentive to remove other people's transactions and lose such fees.

[34] For example, if someone transfers a balance d in block T, it must be the case that the person has received sufficient net flows from block 0 to block $T-1$ so that the accumulated amount is at least d.

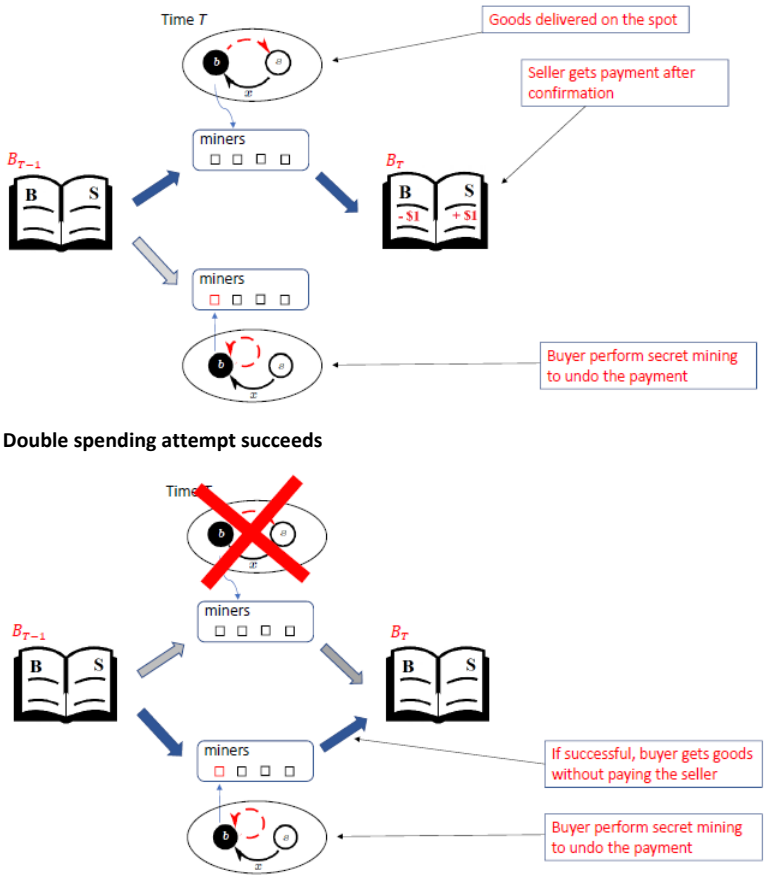


Figure 13: Double Spending Attack

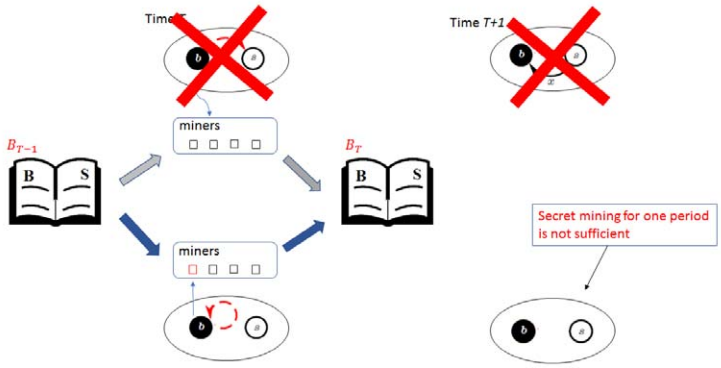
competition and the length of the confirmation lag.

Suppose the buyer successfully solves the PoW for the block containing this alternative history. Note that the buyer has an option whether to broadcast the secretly mined block immediately or withhold it for future mining. If he decides to broadcast the block immediately, the seller will not receive the payment, but he will also not deliver the goods as shown on the top of the figure. Hence, the double-spending attack is not successful for the buyer.

Alternatively, the buyer can temporarily withhold the solved block and continue to secretly mine another block (depicted on the bottom of the figure). Specifically, the buyer needs to allow other miners to confirm the original payment to the seller, so as to induce the seller to deliver the goods. At the same time, the buyer needs to secretly mine *two blocks in a row* for which the original transaction is removed.[35] If the buyer is successful in mining two blocks faster than other miners, he can announce an alternative blockchain after the goods are delivered. In this case, the buyer gets the goods without paying the seller. More generally, if the seller delivers the goods only after observing N confirmations of the payment, the buyer needs to solve blocks $N+1$ consecutive times in order to double spend successfully.

[35]Since the consensus protocol prescribes that the longest chain is accepted by the miners, the double spender needs to mine two blocks in order to create an alternative, longer blockchain superseding the existing one which has one block solved already.

Double spending attempt fails (with one confirmation lag)



Double spending attempt succeeds (with one confirmation lag)

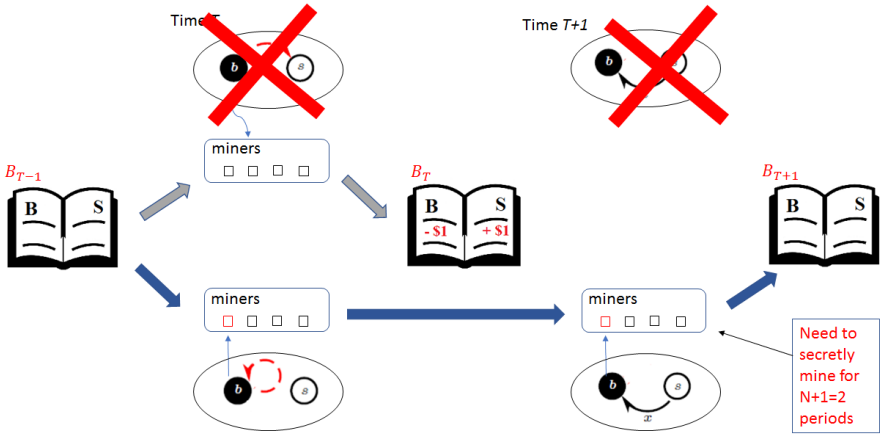


Figure 14: Double Spending Attack when Confirmation Lag

C Appendix [FOR ONLINE PUBLICATION ONLY] – A formal description of the blockchain

Aggregate State

Let $m_t^C(i) \geq 0$ denote the cryptocurrency balance held by person i in period t of the centralized market. We use $\mathcal{S}_t^D = \{m_t^C(i)\}$ to denote the entire public record of the holdings of these balances, called the *aggregate state*. Similarly, $m_{t,n}^D(i) \geq 0$ and $\mathcal{S}_{t,n}^D$ denote the balances and the state at the beginning of the nth subperiod of the period t decentralized market. The economy starts with a given initial state $\mathcal{S}_0^C$.

Payments

We use $\Delta_t^C(i,j)$ and $\Delta_{t,n}^D(i,j)$ to denote transfers of balances from agent i to agent j in the centralized and the decentralized market. We call these transfers *payments*. A payment in the centralized market is *feasible* if

$$\Delta_t^C(i,j) \geq 0 \tag{C.1}$$

$$m_t^C(i) \geq \sum_j \Delta_t^C(i,j). \tag{C.2}$$

Similarly, a payment in the decentralized market in subperiod n is *feasible* if

$$\Delta_{t,n}^D(i,j) \geq 0 \tag{C.3}$$

$$m_{t,n}^D(i) \geq \sum_j \Delta_{t,n}^D(i,j). \tag{C.4}$$

The interpretation is that any person can pay positive amounts to others and the total payments are bounded by the balances a person has accumulated.[36] A person's balance accumulates over time as a result of net payment inflows. Therefore, given any payments, the state is updated according

[36] Through cryptography, the authenticity of payments is protected by digital signatures that only the owner of the balance controls. As a result, only the owner of a balance can transfer it to another person. This gives rise to the non-negativity and the cash-in-advance constraints as used in these definitions.

to

$$m^D_{t,0}(i) = m^C_t(i) + \sum_j \Delta^C_t(j,i) - \Delta^C_t(i,j) + T_t(i), \tag{C.5}$$

$$m^D_{t,n}(i) = m^D_{t,n-1}(i) + \sum_j \Delta^D_{t,n-1}(j,i) - \Delta^D_{t,n-1}(i,j), \text{ for } n = 1, ..., \bar{N} \tag{C.6}$$

$$m^C_{t+1}(i) = m^D_{t,\bar{N}}(i) + \sum_j \Delta^D_{t,\bar{N}}(j,i) - \Delta^D_{t,\bar{N}}(i,j) \tag{C.7}$$

for payments in the centralized and decentralized market, where $T(i)$ denotes the potential transfers of new balances created by the cryptocurrency to person i in the centralized market.

Blockchain

Since we assume public monitoring in the centralized market, feasible payments during this market automatically update the aggregate state according to the rule (C.5). The new state at the start of the decentralized market is thus given by

$$\mathcal{S}^D_{t,0} = \Psi^D_0(\mathcal{S}^C_t, \mathcal{B}^C_t) \tag{C.8}$$

where $\mathcal{B}^C_t = \{\Delta^C_t(i,j)\}$ is the entire set of transfers in the centralized market and is called a *block*. The update function Ψ^D_0 is defined according to the payments as describe in equation (C.5).

Payments made during the decentralized market, however, enter the state through a process we call *mining*. When agent i makes a payment to agent j in the decentralized market, he needs to send out an instruction for a feasible payment $\Delta^D_{t,n}(i,j)$ to miners who compete to update the state with a new block of feasible payments in subperiod n of the market. The set of feasible payment instructions $\mathcal{B}^D_{t,n} = \{\Delta^D_{t,n}(i,j)\}$ is the nth block of period t payments in the decentralized market.

A sequence of blocks $\{\mathcal{B}^C_t, \{\mathcal{B}^D_{t,n}\}^{\bar{N}}_{n=0}\}^T_{t=0}$ iteratively generates a sequence of states $\{\mathcal{S}^C_t, \{\mathcal{S}^D_{t,n}\}^{\bar{N}}_{n=0}\}^{T+1}_{t=0}$ according to

$$\mathcal{S}^D_{t,0} = \Psi^D_0(\mathcal{S}^C_t, \mathcal{B}^C_t) \tag{C.9}$$

$$\mathcal{S}^D_{t,n} = \Psi^D_n(\mathcal{S}^D_{t,n-1}, \mathcal{B}^D_{t,n-1}), \text{ for } n = 1, ..., \bar{N} \tag{C.10}$$

$$\mathcal{S}^C_{t+1} = \Psi^C(\mathcal{S}^D_{t,\bar{N}}, \mathcal{B}^D_{t,\bar{N}}), \tag{C.11}$$

where the update functions Ψ^D_n, Ψ^C are defined by the equations (C.5)-(C.7). We call the sequence of blocks $\mathcal{B}_T = \{\mathcal{B}^C_t, \{\mathcal{B}^D_{t,n}\}^{\bar{N}}_{n=0}\}^T_{t=0}$ a *blockchain*. Determined by the process of mining, one specific blockchain is used to construct the public state and can be observed by everyone in the economy at all times.

D Appendix [FOR ONLINE PUBLICATION ONLY] – Proof-of-Stake (PoS) Consensus Protocol

We assume that the right for updating of the blockchain is allocated randomly across people. The probability that one can update the chain is proportional to the unused balances in the night market.

This is identical to pledging balances that are not used in transactions in order to gain a right to update the chain. For simplicity, we assume that people can pledge balances after they learn whether they have a match or not.

Denote the balances chosen in the day market for transaction purposes as d and the additional amount of balances chosen for having a stake in the voting procedure as z'.

Whenever, $z' > 0$, we have that there is a strictly positive probability that there is a double spend, since the buyer in a transaction will propose a block that undoes the payment whenever he is allowed doing so.

Fix N. The problem is given by

$$\max_{d,z'} -(z'+d) + \sigma\left[\delta^N \varepsilon u(x) + \frac{\beta}{\mu}\left(z' + \left(\frac{z'}{(1-\sigma)Z}\right)^{N+1} d + \left(\frac{z'}{(1-\sigma)Z}\right) R(\bar{N}+1)\right)\right] + (1-\sigma)\frac{\beta}{\mu}\left[z' + d + \left(\frac{z'+d}{(1-\sigma)Z}\right) R(\bar{N}+1)\right]$$

When having a transaction, a buyer is only allowed to update the chain if $z' > 0$. Given N, he needs to win the right to update $N+1$ in order to double spend. If a buyer has no transaction, he will pledge all his balances as a stake. Note that participation of the seller requires

$$x = \frac{\beta}{\mu} d(1-\tau)\left(1 - \left(\frac{\mathbb{E}(z')}{Z}\right)^{N+1}\right).$$

Notice that the buyer's choice of z' is not observed by the seller, so x depends on seller's expectation $\mathbb{E}(z')$ instead.

Lemma D.1. *The objective function is strictly convex in z' for all $N \geq 1$ and linear in z' for $N = 0$.*

Proof. Differentiating the objective function, we obtain

$$-i + \left(\frac{1}{1-\sigma}\right)(\sigma\tau + (\mu - 1)) + \left(\frac{\sigma}{1-\sigma}\right)(N+1)\left(\frac{z'}{(1-\sigma)Z}\right)^N \left(\frac{d}{Z}\right) \tag{D.1}$$

Differentiating again, we obtain

$$\left(\frac{\sigma}{1-\sigma}\right)(N+1)N\left(\frac{z'}{(1-\sigma)Z}\right)^{N-1}\left(\frac{d}{Z}\right) > 0 \tag{D.2}$$

or 0 if $N = 0$. □

We are only looking for a sufficient condition so that we can support the policy $N = 0$, $\mu = 1$ and $\tau = 0$. Interestingly, for $N = 0$, we only need to check the slope at $z' = 0$. This case implies settling of the trade on the spot without delay. The idea is that the buyer has zero probability of updating the chain and, hence, can never double spend.[37]

Importantly, setting $\mu = 1$ and $\tau = 0$ implies that people have nothing at stake when updating the chain. The do not receive a reward and there are no penalties. In turn, setting positive rewards increase the incentives for traders to acquire more balances.

Proposition D.2. *Set $N = 0$, $\mu = 1$ and $\tau = 0$. Then the PoS protocol avoids double-spending for B sufficiently large and σ sufficiently small.*

Proof. For the policy parameters, (D.1) implies that the objective function is decreasing in z' if

$$\left(\frac{\sigma}{1-\sigma}\right)\left(\frac{d}{Z}\right) \leq i = \frac{1}{\beta} - 1$$

which is satisfied for B sufficiently large and σ sufficiently small since d is bounded. □

Note that, even though a system based on PoS can avoid the double spending attacks studied above, it may still be subject to "long range attacks". For example, if all coins are owned initially by one issuer, then he can easily propose a long fork starting from the initial block to claim all existing balances in any future dates. By assuming that the blockchain cannot be rewritten after the end of the night period, our model has ruled out these types of long-range attacks. This captures the feature of checkpoints introduced in Bitcoin.

Interestingly, these are exactly the concerns and proposals raised by practitioners. For example, BitFury Group (2015) pointed out that "In PoW-type systems, this [long range] attack is prevented

[37] We might need to look at $N = 1$ since the buyer can still prodcast a different transaction so that the seller at least needs to see one transaction in the chain to hand over the good. This will, however, not influence the analysis, since the condition for $z' = 0$ being optimal is identical except for that we do neet need to check the slope, but the value of the function at $z' = (1-\sigma)Z$. Or we could require a negative derivative at $z' = (1-\sigma)Z$.

by the enormous amount of computational power needed to build the blockchain from scratch; however, this task is within the realm of possibility with proof of stake. As the attacker is able to move coins freely in the blockchain he is building, he has a much higher dimensionality of the search space ... To prevent a long range attack, the protocol can specify the maximum allowed depth of a branching point."

Finally, note that there are issues with PoS other than double spending. Consensus could be a problem especially if there is a nothing-at-stake problem or if we need to ensure that new parties can decide unambiguously between competing chains. Our model is not developed to address these issues.