The Economics of Cryptocurrencies – Bitcoin and Beyond*

Jonathan Chiu
Bank of Canada
Victoria University of Wellington

Thorsten Koeppl
Queen's University

April, 2017

Abstract

A general equilibrium monetary model is developed to study the optimal design of a cryptocurrency system based on a blockchain. The model is then calibrated to Bitcoin transaction data to perform a quantitative assessment of the scheme. We formalize the critical elements of a cryptocurrency: the blockchain to keep a history of transactions, the distributed updating of information and consensus through competition for such updating. We show that, unlike cash, a cryptocurrency system does not support an immediate, final settlement. In addition, the current Bitcoin scheme generates a welfare loss of 1.4% of consumption. Such loss can be lowered substantially to 0.08% by adopting the optimal policy which reduces mining and relies on money growth rather than transaction fees to finance mining rewards. The efficiency can potentially be improved further by adopting an alternative consensus protocols such as the proof-of-stake. A key economic feature of a cryptocurrency system is that mining is a public good, while double spending to defraud the cryptocurrency depends on individual incentives to reverse a particular transaction. As a result, a cryptocurrency works best when the volume of transactions is large relative to the individual transaction size (e.g., as in a retail payment system).

Keywords: Cryptocurrency, Blockchain, Bitcoin, Double Spending, Mining

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.

1 Introduction

Since the creation of Bitcoin in 2009, numerous private cryptocurrencies have been introduced.[1] Bitcoin is by far the most successful one. It has been getting a lot of media attention, and its total market value has reached 20 billions USD in March 2017. More importantly, a number of central banks started recently to explore the adoption of cryptocurrency and blockchain technologies for retail and large-value payments. For example, the People's Bank of China aims to develop a nationwide digital currency based on blockchain technology; the Bank of Canada and Monetary Authority of Singapore are studying its usage for interbank payment systems; the Deutsche Bundesbank has developed a preliminary prototype for blockchain-based settlement of financial assets. Many proponents believe that cryptocurrency and blockchain technology will have a significant influence on the future development of payment and financial systems.

While policy makers concern about the opportunities and challenges brought about by these technological advances, there is very little guidance provided by economic theory regarding the appropriate usage of these technologies and the optimal design of these systems. This paper attempts to provide an economic theory to help us understand the fundamental economic trade-offs and address relevant policy issues. Most existing models of cryptocurrencies are built by computer scientists who focus mainly on the feasibility and security of these systems.[2] This line of research often ignores the incentives of participants (e.g., the incentives of malicious attackers) and the endogenous nature of key variables (e.g., the real value of cryptocurrencies). More importantly, to study the optimal design of a cryptocurrency system, we need to model from first principles the behaviors of different participants, to derive the equilibrium interactions among these agents and to study the optimal usage of different policy instruments. To this end, this paper develops a general equilibrium monetary model of a cryptocurrency system to study its optimal design. This approach is desirable because the model endogenizes the value of cryptocurrency, and endogenizes the underlying trading activities and mining activities. It also provides a welfare notion for assessing alternative system designs. We will use this model to evaluate the performance of a cryptocurrency system calibrated to Bitcoin transaction statistics. We will study the optimal design of the cryptocurrency system in different settings. Furthermore, we compare the usage of different consensus protocols

[1] By July 2016, more than 740 cryptocurrencies have been introduced.

[2] The book by Narayanan et al. (2016) provides a useful overview and references of computer science studies on Bitcoin and cryptocurrency technologies.

(e.g. proof-of-work and proof-of-stake), and to evaluate the efficiency of a cryptocurrency system relative to a cash system.

The economic literature on cryptocurrencies is very thin. So far, there are only a few economic models developed to study this new payment technology.[3] These models use different frameworks to address different research questions, and often focus on different aspects of cryptocurrencies. Chiu and Wong (2015) apply the mechanism design approach to review several e-money technologies including Bitcoin, PayPal and M-Pesa and identify some essential features of e-money that can help implement constrained efficient allocations. Gans and Halaburda (2013) develop a model of platform management to study platform-specific digital currencies such as Facebook Credits. Fernández-Villaverde and Sanches (2016) model cryptocurrencies as privately issued fiat currencies and analyze whether competition leads to efficiency. Agarwal and Kimball (2015) advocate that the adoption of digital currencies can facilitate the implementation of a negative interest rate policy. Rogoff (2016) suggests subsidizing the provision of digital money to the unbanked in order to phase out paper currency which facilitates undesirable tax evasion and criminal activities. To the best of our knowledge, our work is the first paper that explicitly models the distinctive technological features of a cryptocurrency system (e.g. blockchain, mining, double-spending problems) in an equilibrium monetary model and investigates its optimal design both qualitatively and quantitatively.

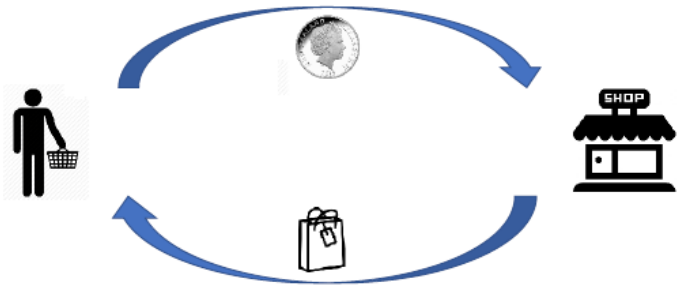
2 Cryptocurrencies: A Brief Review

For readers less familiar with cryptocurrencies, this section briefly reviews some of their key features, highlighting the main differences from traditional payment systems.

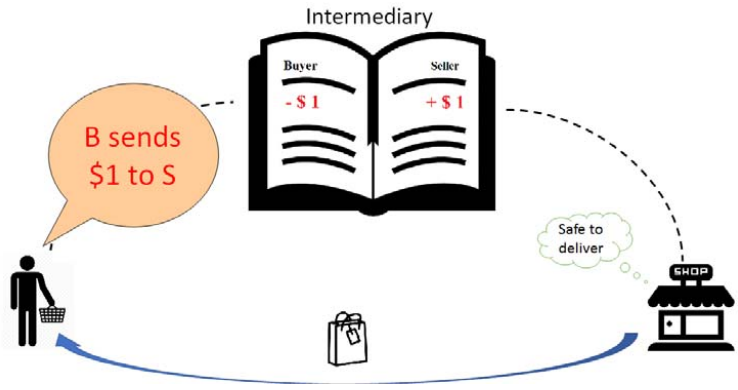
For thousand years, physical tokens have been being used as means of payment (e.g. shells, gold coins, bank notes). In such setting, a direct exchange of sellers' goods and buyers' tokens allows them to achieve an immediate and final settlement. (See Panel (a) in Figure 1). This option is unavailable, however, when the two parties are not present in the same location (e.g. e-commerce), necessitating the usage of digital tokens. In a digital currency system, the means of payment is simply a string of bits. It becomes challenging to prevent the buyer from re-using the same bit

[3]Examples of empirical research include Moore and Christin (2013), Yermack (2013) and Gandal and Hałaburda (2014), Glaser et al. (2014).

(a) Physical tokens (e.g. cash)



(b) Digital tokens with a trusted third party (e.g. PayPal)



(c) Digital tokens in a decentralized network (e.g. Bitcoin)

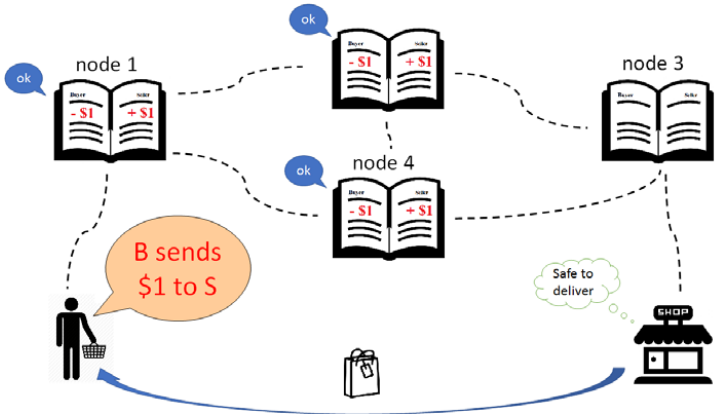


Figure 1: Different Currency Systems

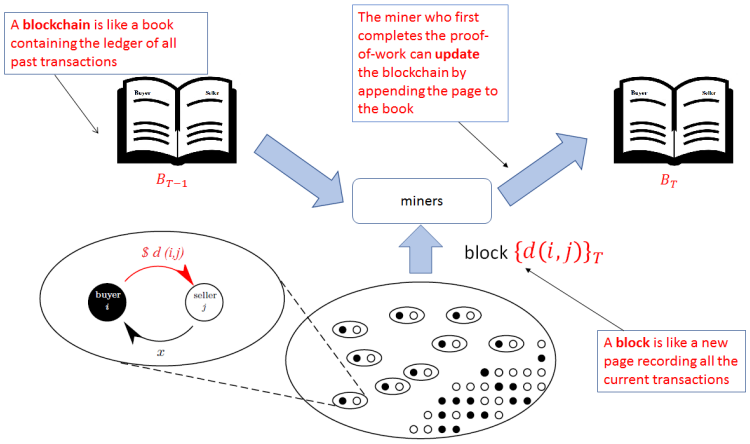


Figure 2: How the Blochchain is Updated

string over and over again. This is called the double-spending problem. This problem can be solved easily when there is a trusted third party (e.g. PayPal) who manages a centralized ledger and transfers balances by crediting and debiting buyers and sellers' accounts. (See Panel (b)). In many settings, it is infeasible to find (e.g., lack of trust) or undesirable to use (e.g., the single-point-of-failure problem) a trusted third party. In particular, cryptocurrencies such as Bitcoin are used as a digital means of payment in a distributed network in the absence of a trusted third party. (Panel (c)).

A cryptocurrency system in a decentralized network typically needs to overcome three challenges:

1. How to establish a consensus in a distributed network?
2. How to discourage double spending behaviors?
3. How to encourage proper transaction validation?

How do cryptocurrencies such as Bitcoin tackle these problems? In the absence of a central authority, the cryptocurrency relies on a distributed verification of transactions, updating and storage of the record of transaction histories. This necessitates that consensus between the users is maintained about the correct record of transactions. This trust in the currency is established by having a competition for the right to update record. This competition can take various forms. In Bitcoin, this is through a process called "mining". Miners (i.e. transaction validators) compete to solve a computationally costly problem ("proof-of-work"). The winner of this mining process has the

right to update the record and be the first to propose a new history to the network. The consensus protocol prescribes that the "longest" history will be accepted as the valid public record. Other consensus protocol are available such as the proof-of-stake used by Peercoin.

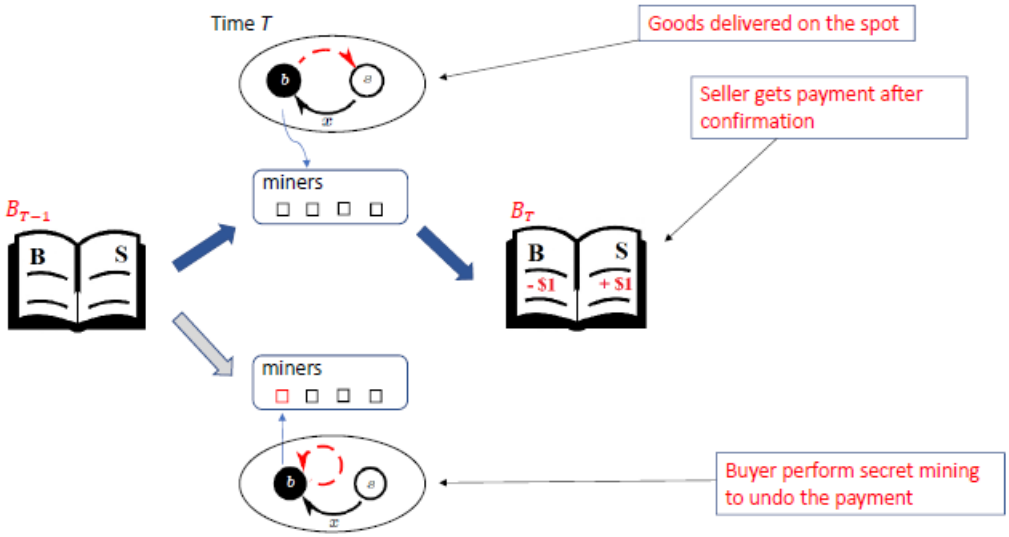
A concern in any cryptocurrency system is the double spending problem: after a transaction, the buyer attempts to convince the entire network to accept an alternative history in which the payment was not conducted. When the attack succeeds, the buyer keeps both the balances and the product while the seller will be left empty handed. The possibility of double-spending can undermine the usage of the cryptocurrency. This problem is mitigated by the usage of the blockchain and by introducing confirmation lags. Unlike cash, a cryptocurrency keeps track of the history of all transactions.[4] 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. Figure 2 illustrates how the blockchain is updated. The blockchain requires that transactions taking place in different blocks have to be dynamically consistent.[5] If a person attempts to revoke a transaction in the past, he has to solve for an alternative blockchain consistent with his proposal. This makes it very costly to rewrite the history of transactions backwards if the chain is long. This feature makes double-spending attacks costly. In addition, double spending can be discouraged by introducing a confirmation lag into the transactions. By waiting some blocks before completing the transaction (i.e. delayed delivery of goods by sellers), it becomes harder to alter transactions in a sequence of new blocks. Figure 3 and 4 illustrate why confirmation lags raise the secret mining burden of a double spender. In general, if the seller delivers the goods only after observing N confirmations of the payment, then the buyer needs to solve the proof-of-work for $N+1$ consecutive times in order to double spend successfully.

Finally, since transaction validation and mining are costly, a reward structure is needed to incentivize honest miners. In Bitcoin, the rewards are financed by the creation of new coins and transaction fees.

[4] In this sense, money is merely a partial memory as it only records the current distribution of balances and does not record how past transacions generate the current distribution.

[5] For example, if an address transfers d units in block T, it must be the case that the accumulated net flows into this address from block 0 to block $T-1$ is at least d.

Case 1: No confirmation lag (N=0). Double spending attempt fails.



Case 2: No confirmation lag (N=0). Double spending attempt succeeds.

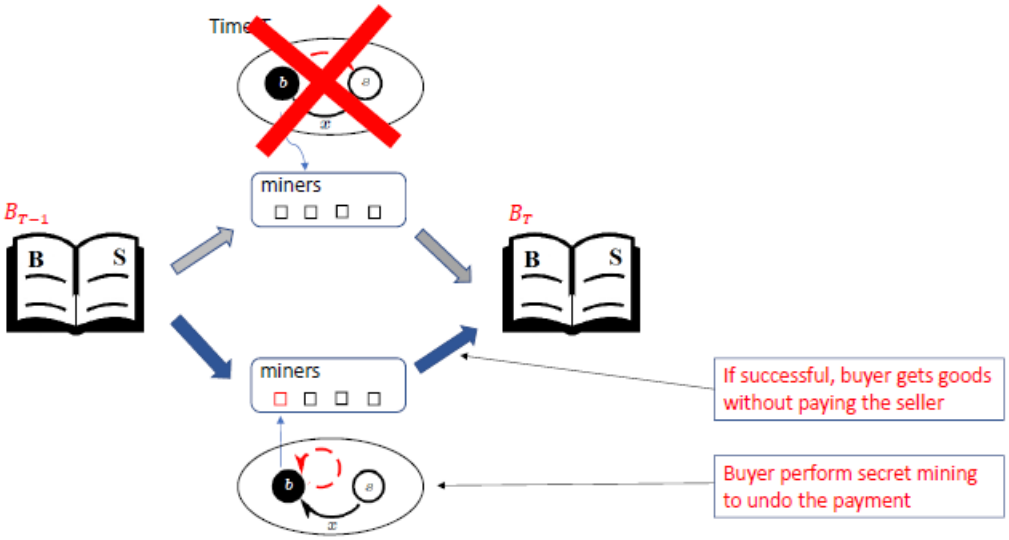
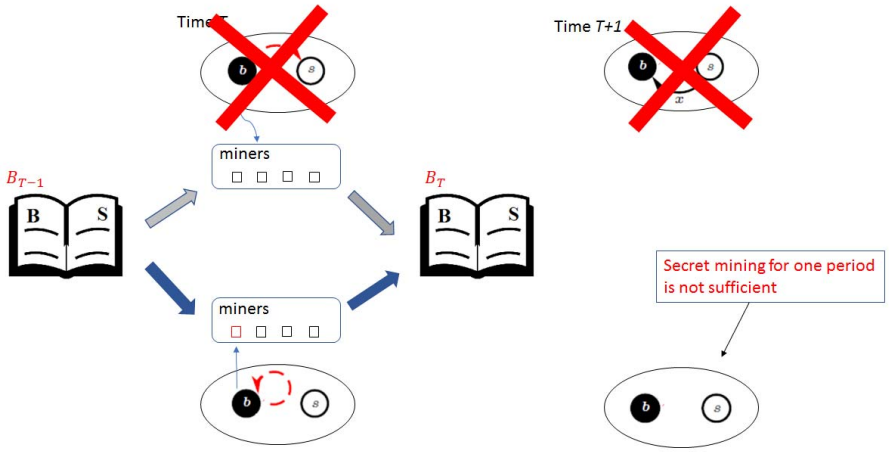


Figure 3: Double Spending Attack when N=0

Case 3: Confirmation lag N=1. Double spending attempt succeeds (with 1-period secret mining).



Case 4: Confirmation lag N=1. Double spending attempt succeeds (with 2-period secret mining).

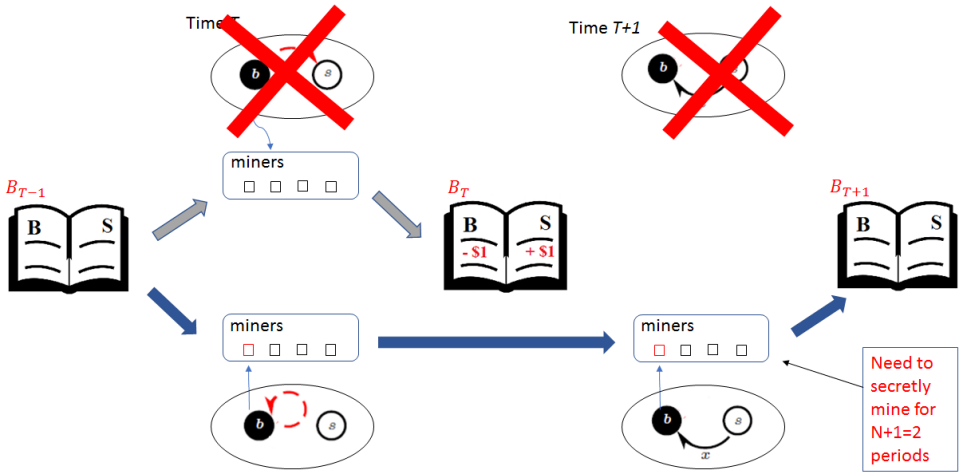


Figure 4: Double Spending Attack when N=1

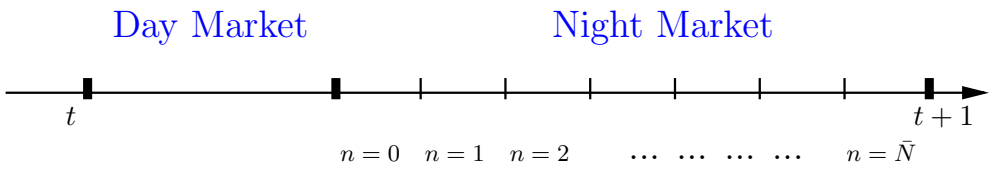


Figure 5: Time line

3 Model

3.1 Basic Set-up

Our model bases on the alternating market formulation from Lagos and Wright (2005). This framework is useful because it allows us to study frictions that give rise to the usage of money while still keeping the distribution of balances analytically tractable. Time is discrete and denoted by $t = 0, 1, 2, ...$There are a large number B of buyers and a large number $S = B\sigma$ of sellers, where $\sigma \in (0, 1)$. In addition, there are also M miners. In each period, a day market first opens and then the night market opens. The day market is a competitive market for trading a general good h to replenish balances. The night market is a decentralized market for trading a consumption good x which is produced by sellers and consumed by buyers. As shown in Figure 5, the night market is divided into $\bar{N} + 1$ consecutive trading sessions: $n = 0, 1, ...\bar{N}$ with $\bar{N} \geq 1$.[6] In the night market, trades take place bilaterally and are not monitored in the sense that obligations from the trade cannot be enforced later on. To the contrary, in the night market everyone can trade the general good. Such trades are centralized and monitored.

3.2 Cryptocurrency

Due to the anonymity in the night market, the exchange of goods necessitates a means of payment which we assume is a *cryptocurrency*.[7] A cryptocurrency is a digital record-keeping device that uses balances to keep track of the obligations from trading and that is publicly known to all traders. A cryptocurrency system is defined by two parameters: money growth rate $\mu \geq 0$ and transaction fee charge at a rate $\tau \geq 0$. As discussed, the digital nature of these balances result in the double-

[6]The setting with multiple trading sessions allows us to study confirmation lags.

[7]We assume that the cryptocurrency is the only means of payments available in the night market. For example, cash is not viable for online trades, and standard electronic payments such as debit or credit cards are not available to the unbanked. Chiu and Wong (2014, 2015) study the competitioin between cash and e-money in a different setting.

spending problem. In what follows, we describe the features of a ledger that records the transfers of these balances.

Aggregate State

Each trader is entitled to a balance. Let $m_t^D(i) \geq 0$ denote the balance associated with agent i in the period t day market. We then use $\mathcal{S}_t^D = \{m_t^D(i)\}$ to denote the entire public record of these balances, called the *(aggregate) state*. Similarly, $m_{t,n}^N(i) \geq 0$ and $\mathcal{S}_{t,n}^N$ denote the balances and the state at the beginning of the nth trading session of the period t night market. The economy starts with a given initial state $\mathcal{S}_0^D$.

Payments

We use $\Delta_t^D(i,j)$ and $\Delta_{t,n}^N(i,j)$ to denote respectively day and night transfers of balances from agent i to agent j and call these transfers *payments*. A day payment is *feasible* if

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

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

Similarly, a night payment is *feasible* if

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

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

A trader can pay positive amounts to others and the total payments are bounded by the balances one has accumulated.[8] Given any payments the state is then updated in the two markets according to

$$m_{t,0}^N(i) = m_t^D(i) + \sum_j \Delta_t^D(j,i) - \Delta_t^D(i,j) + T_t(i), \tag{5}$$

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

$$m_{t+1}^D(i) = m_{t,\bar{N}}^N(i) + \sum_j \Delta_{t,\bar{N}}^N(j,i) - \Delta_{t,\bar{N}}^N(i,j) \tag{7}$$

[8]Through cryptography, the authenticity of payments is protected by digital signatures corresponding to the sending addresses. As a result, only the owner of the digital signature can transfer balances to another address. This gives rise to the non-negativity and the cash-in-advance constraints.

where $T(i)$ is the transfer of new balances to agent i.

Blockchain

Due to public monitoring, we assume that feasible payments during the day automatically[9] update the aggregate state according to the rule (5). The new state at the start of the night market is thus given by

$$\mathcal{S}_{t,0}^N = \Psi_0^N(\mathcal{S}_t^D, \mathcal{B}_t^D) \tag{8}$$

where $\mathcal{B}_t^D = \{\Delta_t^D(i,j)\}$ is the entire set of day transfers and is called a *block*. The update takes place according to (5).

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

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

$$\mathcal{S}_{t,0}^N = \Psi_0^N(\mathcal{S}_t^D, \mathcal{B}_t^D) \tag{9}$$

$$\mathcal{S}_{t,n}^N = \Psi_n^N(\mathcal{S}_{t,n-1}^N, \mathcal{B}_{t,n-1}^N), \text{ for } n = 1, ..., \bar{N} \tag{10}$$

$$\mathcal{S}_{t+1}^D = \Psi^D(\mathcal{S}_{t,\bar{N}}^N, \mathcal{B}_{t,\bar{N}}^N), \tag{11}$$

defined according to (5)-(7). We call the sequence of blocks $\mathcal{B}_T = \{\mathcal{B}_t^D, \{\mathcal{B}_{t,n}^N\}_{n=0}^{\bar{N}}\}_{t=0}^T$ 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.

3.3 Mining

There are M miners performing mining activities to update the public ledger in the night trading sessions $n = 0, ..., \bar{N}$. In each session, miners perform a costly computational task with a random success rate by investing computing power q_n. This task is called the proof of work (PoW). As

[9]This is without loss of generality and simplifies the analysis. It is straightforward to model the mining process in the day market.

specified by the Bitcoin protocol, if the computational power of miner i in session n is $q_n(i)$, then the probability that a particular miner j will be the first one to solve the proof-of-work problem is given by

$$\rho_n(j) = \frac{q_n(j)}{\sum_{i=1}^{M} q_n(i)}.$$

In other words, the probability of wining the mining game is proportional to the fraction of computational power owned. We take this feature as given here and, in the appendix, we provide a micro-foundation for this result. By winning the competition of session n, a miner can update the blockchain (i.e., appending the nth block to the blockchain) and receives R real balances as a reward.

We use $\varpi_n(j)$ to denote the number of blocks that miner j has already solved by the end of session n. Define $\Pi_n(\varpi_{n-1})$ as the value function of a miner at the beginning of session n. In the last trading session $\bar{N}$, the value of a miner j who has already solved $\varpi_{\bar{N}-1}(j)$ blocks is

$$\Pi_{\bar{N}}[\varpi_{\bar{N}-1}(j)] = \frac{\beta}{\mu}\varpi_{\bar{N}-1}(j)R + \beta\Pi_0' + \max_{q_{\bar{N}}(j)}\left[-q_{\bar{N}}(j)\alpha + \rho_{\bar{N}}(j)\frac{\beta}{\mu}R\right]$$

where Π_0' is the continuation value in the next period. Here, the miner always receives the rewards R for the $\varpi_{\bar{N}-1}(j)$ mined blocks. The rewards are discounted by the discount factor β and the currency growth rate μ. In addition, in the last session, the miner incurs a mining cost $q_{\bar{N}}(j)\alpha$ and wins the block with probability $\rho_{\bar{N}}(j)$. For any session $n = 1, ..., \bar{N}-1$, the value function is given by

$$\Pi_n[\varpi_{n-1}(j)] = \max_{q_n(j)} -q_n(j)\alpha + [1-\rho_n(j)]\Pi_{n+1}[\varpi_{n-1}(j)] + \rho_n(j)\Pi_{n+1}[\varpi_{n-1}(j)+1].$$

The miner incurs a mining cost $q_n(j)\alpha$ to mine in an attempt to increase the number of winning blocks by 1 with probability $\rho_n(j)$. In session 0, the value function is

$$\Pi_0 = \max_{q_0(j)} -q_0(j)\alpha + [1-\rho_0(j)]\Pi_1(0) + \rho_0(j)\Pi_1(1).$$

It is straight forward to show that

$$\Pi_0 = \sum_{n=0}^{\bar{N}}\left[\rho_n(j)\frac{\beta}{\mu}R - q_n(j)\alpha\right] + \beta\Pi_0'$$

where $q_n(j)$ solves

$$\frac{\sum_{i=1}^{M} q_n(i) - q_n(j)}{[\sum_{i\neq j} q_n(i) + q_n(j)]^2}\frac{\beta}{\mu}R = \alpha.$$

Imposing symmetry, $q_n(j) = Q$ for all j, we obtain

$$\alpha Q = \frac{M-1}{M^2}\frac{\beta}{\mu}R.$$

Consequently, the total computing cost incurred by the mining community in session n is

$$M\alpha Q = \frac{M-1}{M}\frac{\beta}{\mu}R$$

The expected value of a miner is

$$\Pi_0 = (\bar{N}+1)\left[\frac{Q}{\sum_{m=1}^{M} Q}\frac{\beta}{\mu}R - Q\alpha\right] = \frac{\bar{N}+1}{M^2}\frac{\beta}{\mu}R + \beta\Pi_0'.$$

We can always normalize α to 1 by redefining the unit of computer power. To capture the fact that the mining activities are quite competitive and open to new entrants, we will assume that $M \to \infty$.[10] In that case, Π_0 converges to zero and the mining costs converges to $\beta R/\mu$.

Lemma 1. *As $M \to \infty$, the expected value of miners is $\Pi_0 = 0$, and the total computing power of the miners is*

$$C \equiv \alpha M Q = \frac{\beta}{\mu}R.$$

3.4 Double Spending and Mining Rewards

As discussed in previous sections, an important concern in a cryptocurrency system is buyers' double-spending attempts. In the day market, there is perfect monitoring in the sense the payer of a payment is liable for their authenticity of the balances. That is, if the balances get lost for the payee, the payer needs to reimburse the payee for the loss. This assumption rules out double spending in the day market.[11]

In the night market, a buyer meets with a seller with probability σ. If they agree to trade, the buyer needs to make a payment to a seller. To do so, he has to send out an instruction $\Delta_{t,0}$ to the pool of miners. However, this is insufficient to ensure that the seller receives a payment. A buyer

[10] The total number of miners is estimated to be within the range of 5000 to 100,000 (https://goo.gl/TPFBvA). In addition, according to blockchain.info, there are altogether 14 mining pools that indivudally can account for at least 1% of the total hashrate. Finally, it is feasible for miners to use their existing mining capacities to mine different cryptocurrencies. For example, ASICs (Application-specific integrated circuits) manufactured for Bitcoin can be used to mine altcoins that use SHA-256 as the hashing algorithm (e.g., Namecoin and Peercoin).

[11] This reflects the basic premise that certain parties such as merchants accepting a cryptocurrency and using it could be held legally liable for the losses sustained by other parties. In general, payers in many settings are not fully anonymous (e.g. the KYC requirement, online wallets transactions, payments to Bitcoin exchanges, payments made by well-known merchants). Hence, double spenders can be punished either formally (e.g. legal) or informally (e.g. reputational).

can engage in secret mining by attempting to mine a block in which his payment did not occur.[12] As discussed, a seller can protect himself from not receiving the payment by waiting to deliver the goods until the payment has been incorporated into the public state at night.[13] However, such confirmation of the payment in the blockchain is not enough. A buyer can secretly mine a different blockchain which could be released some periods after the seller has delivered the good and replaces the original blockchain.[14]

When the secret mining succeeds, the buyer keeps his original balances and the goods while the seller will be left empty handed. This is called *double spending*. In response, the seller can choose to postpone the delivery of the goods and wait for N confirmations. Such a *confirmation lag* can potentially deter double spending by the buyer. The idea is that, to undo a transaction with an N confirmation lag, a dishonest buyer needs to win the mining game $N+1$ times in a row. As the number of lags increases, the total proof-of-work required to revoke a transfer is increasing, making it more costly for a buyer to double spend.

The investments in mining by miners is therefore important to deter dishonest behavior as it determines the probability of success for a double spending attack. The incentives in turn depend on the mining reward R. As in the Bitcoin scheme, we assume that this reward consists of two components. First, the cryptocurrency can create new balances which are paid to miners that win the competition to update the blockchain. We denote the (gross) growth rate of new balances by $\mu \geq 1$. In addition, a fraction $\tau \geq 0$ of balances are paid to the successful miner as a fee. We assume the $\bar{N}+1$ block winners of the night market equally share the total reward. That is

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

where Z is the aggregate money balances and D is the aggregate spending in the night market.

[12] The secret mining can be done either by the buyer himself or by hiring a miner to mine a block with the instruction that the payment did not occur.

[13] This can be seen as what is called delivery-vs-payment or a quid-pro-quo exchange.

[14] Notice that, with such secret mining, the buyer cannot spend the balances of any other agent because, to spend other agents' balances, one needs to obtain others' digital signature. He can only (i) change the payment instructions of his own transaction $\Delta_{t,0}(i,j)$ and (ii) remove other payment instructions $\Delta_{t,\cdot}(j,i)$ from being mined – and, hence, confirmed – in the block. Hence, a buyer trying to double spend has to remove his own payment and all other payment instructions involving his original balance being spent.

3.5 Trading

In the day market, all buyers and sellers have a linear technology to produce a numeraire good which can be used to replenish their balances.

In the night market, each buyer meets with a seller with probability σ in trading session 0. The buyer would like to consume a good that the seller can produce at unit cost. If they agree to trade, the seller produces x in trading session 0 and commit to deliver it to the buyer in session $N \leq \bar{N}$. The buyer's preferences are given by

$$\varepsilon \delta^N u(x)$$

from consuming x with a confirmation lag of N. The discount factor between two periods is β. The discount factor across two adjacent trading sessions is δ.[15] The random variable $\varepsilon_{\min} \leq \varepsilon \leq \varepsilon_{\max}$ is known to the trading partners when the buyer enters the day market and is drawn from a distribution F_ε. To trade, the buyer makes a take-it-or-leave-it offer (x, d, N), which specifies a payment d in real balances for obtaining x goods to be delivered after confirmations of the payment in N consecutive blocks.

4 Equilibrium

4.1 Day Market

Denote the value function in the night market of a buyer with real balances z and preference shock ε by $v(z;\varepsilon)$. In the day market, a buyer can work to replenish balances subject to a linear disutility function. The value of a buyer who draws ε is

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

subject to

$$h + z \geq z' \geq 0$$

where h is the amount of work at night, z' is the real balances carried to the night given ε. The FOC is

$$\mu \geq v'(z';\varepsilon). \tag{12}$$

[15] There are two ways to interpret the discount factor δ: (i) buyers prefer earlier consumption (e.g., $\delta = \beta^{\frac{1}{N+1}}$); (ii) buyer's preference will change over time so that the goods will no longer generate utility with probability $1-\delta$.

with equality whenever $z'(\varepsilon) > 0$. Linear preferences imply that

$$\begin{aligned} w(z;\varepsilon) &= z + w(0;\varepsilon), \\ w'(z) &= 1. \end{aligned}$$

The value function before the realization of ε is

$$w(z) = \mathbb{E}w(z,\varepsilon) = z + W.$$

where $W = \mathbb{E}w(0;\varepsilon)$ is a constant.

4.2 Night Market

In the night market, the buyer makes an offer (x, d, N) to the seller. We call an offer *double spending proof* (DS-proof) if the buyer has no incentive to engage in double spending after the acceptance of the offer. Otherwise, it is a *double-spending* (DS) offer. To proceed, we first study the double-spending problem after the terms of trades are agreed. We aim to construct a DS-proof equilibrium.[16] To this end, we first look at the double-spending problem after trade.

Post-trade Double-Spending Problem

Consider a trade with the terms (x, d, N). The buyer will receive the goods in session 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 to undo his payment after he has received the goods. For such double spending to be successful, he needs to solve the mining game for $N + 1$ consecutive sessions, starting from session 0 and ending in session N. Specifically, the double spender needs to be the first one who solves the proof-of-work for all $N + 1$ sessions. For each of the first N sessions, the buyer does not broadcast the blockchain immediately so that one of the miners will update the blockchain and confirm his payment. The buyer broadcasts his solutions and update the blockchain only after he receives the goods and solves the $N + 1$th block. When the double spending attack succeeds, the original payment is cancelled and the $N + 1$ rewards will be given to the buyer.

[16]The Bitcoin system is designed to discourage double spending attacks. In the current system, there is no evidence for significant double spending activities .

Given (d, N), the expected gain from an optimal double spending is given by[17]

$$\begin{aligned} D_0(d,N) &= \max_{\{q_n\}_{n=0}^N} P\frac{\beta}{\mu}[d+R(1+N)] \\ &\quad -\sum_{n=0}^{N}\left(\prod_{t=0}^{n-1}\frac{q_t}{QM+q_t}\right)q_n \end{aligned} \tag{13}$$

where

$$P=\prod_{n=0}^{N}\left(\frac{q_n}{QM+q_n}\right). \tag{14}$$

is the probability of success given the miners' equilibrium mining efforts Q. Again α is normalized to 1. If $D_0(d,N)<0$, then the contract is DS proof.

To solve the problem, it is useful to reformulate it in terms of $N+1$ sub-problems which take place in sessions $n=0,...,N$. This is illustrated in Figure 6.

We start from the session when the goods are delivered to the buyer. In session N, the buyer's payoff from a successful double spending attack with investment q_N is given by

$$\Lambda_N(q_N;d,N)=\rho(q_N)\frac{\beta}{\mu}[d+(N+1)R]-q_N. \tag{15}$$

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

$$\rho(q_N)=\frac{q_N}{QM+q_N}$$

the miner wins the competition again in the $N+1$th round so that the buyer can double spend. The revenue from double spending is given by the original payment in session 0 which in real balances in the next period is d/μ. Also, the buyer obtains the revenue from all blocks in his chain of length $N+1$. The value in terms of real balances is given by $R(N+1)/\mu$.

Define

$$\Delta=\left[\frac{d}{R}+(N+1)\right],$$

and, as $M\to\infty$, the optimal choice with respect to investment in computing power is given by

$$q_N(d,N)=\sqrt{QMR\frac{\beta}{\mu}\Delta}-QM=\frac{\beta}{\mu}R\left[\sqrt{\Delta}-1\right].$$

[17]The quantity of goods x is not an argument of D_0 as it affects neither the incentives nor the payoff in a double-spending attack.

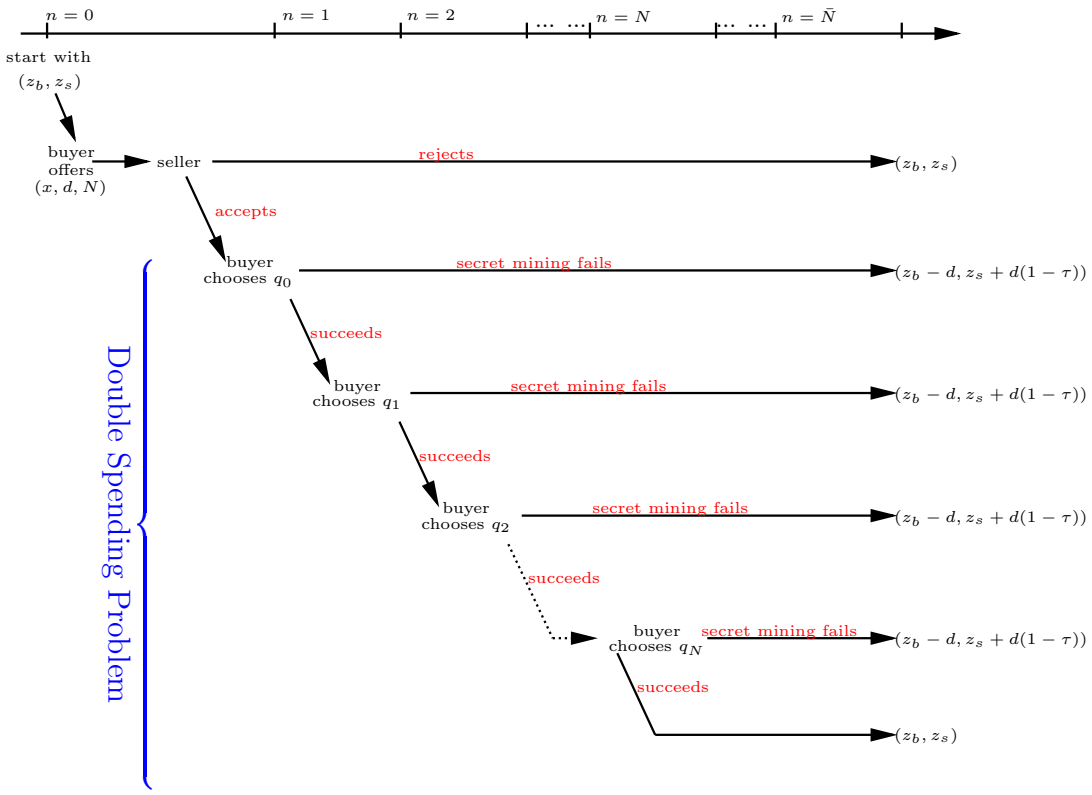


Figure 6: The Double Spending Problem in the Night Market

Therefore the probability of a successful double spending in $n = N$ is

$$\rho(q_N) = \frac{q_N}{QM + q_N} = \frac{\sqrt{\Delta} - 1}{\sqrt{\Delta}}.$$

The expected payoff in session N_t is

$$\begin{aligned} \Lambda_N(q_N; d, N) &= \rho(q_N)\frac{\beta}{\mu}[d + (N+1)R] - q_N \\ &= \frac{\beta}{\mu}R(\sqrt{\Delta} - 1)^2. \end{aligned}$$

To derive the no-double-spending constraint we work backwards. We start with the expected payoff for a double spending buyer making an optimal investment q_N in session N having been successful N times

$$D_N(d, N) = \max_{q_N} \Lambda_N(q_N; d, N) = \frac{\beta}{\mu}R(\sqrt{\Delta} - 1)^2.$$

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

$$D_n(d, N) = \max_{q_n} \Lambda_n(q_n; d, N) = \max_{q_n} \rho(q_n) D_{n+1}(d, N) - q_n.$$

Note that this takes into account that the buyer was n times successful, since if he fails once the double spend fails as well. Again $D_n(d, N)$ can only be positive if $D_{n+1}(d, N)$ is positive and $q_n > 0$. The FOC is given by

$$q_n(d, N) = \sqrt{QM \cdot D_{n+1}(d, N)} - QM.$$

By backward induction, we can obtain the following result.

Lemma 2. *As $M \to \infty$,*

$$\begin{aligned} D_{N-s}(d, N) &= \frac{\beta}{\mu}R\left[\sqrt{\Delta} - (s+1)\right]^2 \\ \rho_{N-s}(d, N) &= \frac{\sqrt{\Delta} - (s+1)}{\sqrt{\Delta} - s} \\ q_{N-s}(d, N) &= \frac{\beta}{\mu}R\left[\sqrt{\Delta} - (s+1)\right] \end{aligned}$$

Proof. This is true for $s = 0$. Suppose the result holds true for $s = n - 1$. Consider $s = n$,

$$\begin{aligned} q_{N-n}(d, N) &= \sqrt{QM \cdot D_{N-n+1}(d, N)} - QM \\ &= \frac{\beta}{\mu}R(\sqrt{\Delta} - n) - \frac{\beta}{\mu}R \\ &= \frac{\beta}{\mu}R[\sqrt{\Delta} - (n+1)]. \end{aligned}$$

$$\begin{aligned}\rho_{N-n}(d,N) &= \frac{q_{N-n}(d,N)}{QM+q_{N-n}(d,N)} \\ &= \frac{\sqrt{\Delta}-(n+1)}{\sqrt{\Delta}-n}.\end{aligned}$$

$$\begin{aligned}D_{N-n}(d,N) &= \rho_{N-n}(d,N)D_{N-n+1}(d,N)-q_{N-n}(d,N) \\ &= \frac{\sqrt{\Delta}-(n+1)}{\sqrt{\Delta}-n}\frac{\beta}{\mu}R(\sqrt{\Delta}-n)^2-\frac{\beta}{\mu}R[\sqrt{\Delta}-(n+1)] \\ &= \frac{\beta}{\mu}R[(\sqrt{\Delta}-(n+1)]^2.\end{aligned}$$

∎

It follows immediately that $D_n(d,N)$ is strictly increasing in n and, consequently, q_n is increasing in n. Hence, if it was optimal to engage in secret mining in period n, it is optimal to continue with secret mining in period $n+1$ if one has been successful in period n. So double spending is not optimal if

$$q_0(d,N)=\frac{\beta}{\mu}R\left[\sqrt{\Delta}-(N+1)\right]<0.$$

This is true when[18]

$$\begin{aligned}\sqrt{\frac{d}{R}+(N+1)} &< (N+1) \\ \frac{d}{R} &< (N+1)N.\end{aligned}$$

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

$$d<R(N+1)N. \tag{16}$$

When double spending is optimal, its expected return is

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

which is decreasing in R,N and increasing in d.

The unconditional probability of a successful double spending is

$$P(d,N)=\prod_{n=0}^{N}\left(\frac{q_n(d,N)}{QM+q_n(d,N)}\right)=\frac{\sqrt{\Delta}-(N+1)}{\sqrt{\Delta}}.$$

[18] In a setting where n_b buyers can coordinate in their double-spending attempts, the condition becomes $dn_b < R(N+1)N$. As $n_b \to \infty$, it becomes impossible to prevent double spending. This may suggest that a cryptocurrency is more secure in a decentralized environment where it is difficult to coordinate a deviation.

We first define the immediacy and finality of the settlement of a transaction as follows.

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

The above derivation implies the following theorem.

Theorem 5. *In a cryptocurrency system, a settlement cannot be both immediate and final.*

The inequality (16) provides a condition for (full) finality. Rewards help achieve finality by inducing mining activities which increase the costs of double spending attempts. Finality can also be supported by either reducing the trade size d or increasing confirmation lag N. Notice that the relationship between d and N defined by (16) is non-linear. The reason is that, while decreasing d and increasing N can both reduce the return of a given secret mining effort, increasing N has the extra effect of making mining more costly by increasing the number of mining periods.[19] Given R, there is a trade-off between trade size d, settlement lag N and finality captured by $1-P(d, N)$. As shown in Figure (7), full finality is feasible only for small, sufficiently delayed settlement. Quick settlement of large transactions is only probabilistic with the probability decreasing in d and increasing in N.

Night Value Function

After solving the post-trade double-spending problem, we can move backward to derive the night value function. For any given terms of trade (x, d, N) the value of a buyer in the day market with balances z is given by

$$\begin{aligned} v(z;\varepsilon) &= \sigma\big([\delta^N \varepsilon u(x) + D_0(d,N)] + \beta w(\frac{z-d}{\mu})\big) + (1-\sigma)\beta w(\frac{z}{\mu}) \\ &= \beta\frac{z}{\mu} + \sigma[\big(\delta^N \varepsilon u(x) + D_0(d,N)] - \beta\frac{d}{\mu}\big) + \beta W. \end{aligned}$$

Since the seller has a linear technology to produce x, the seller's payoff is given by

$$\beta\frac{d}{\mu}(1-\tau)[1-P(d,N)] - x,$$

where $P(d, N)$ is the probability of a successful double spending attack, and τ is the proportional transaction fee which we assume is paid by the seller.

[19] The DS-problem (13) indicates that q interacts with d and N in a different fashion: (i) increasing N raises mining cost at a rate which is linear in q, and (ii) increasing d raises the return at the rate P which is a concave function of q.

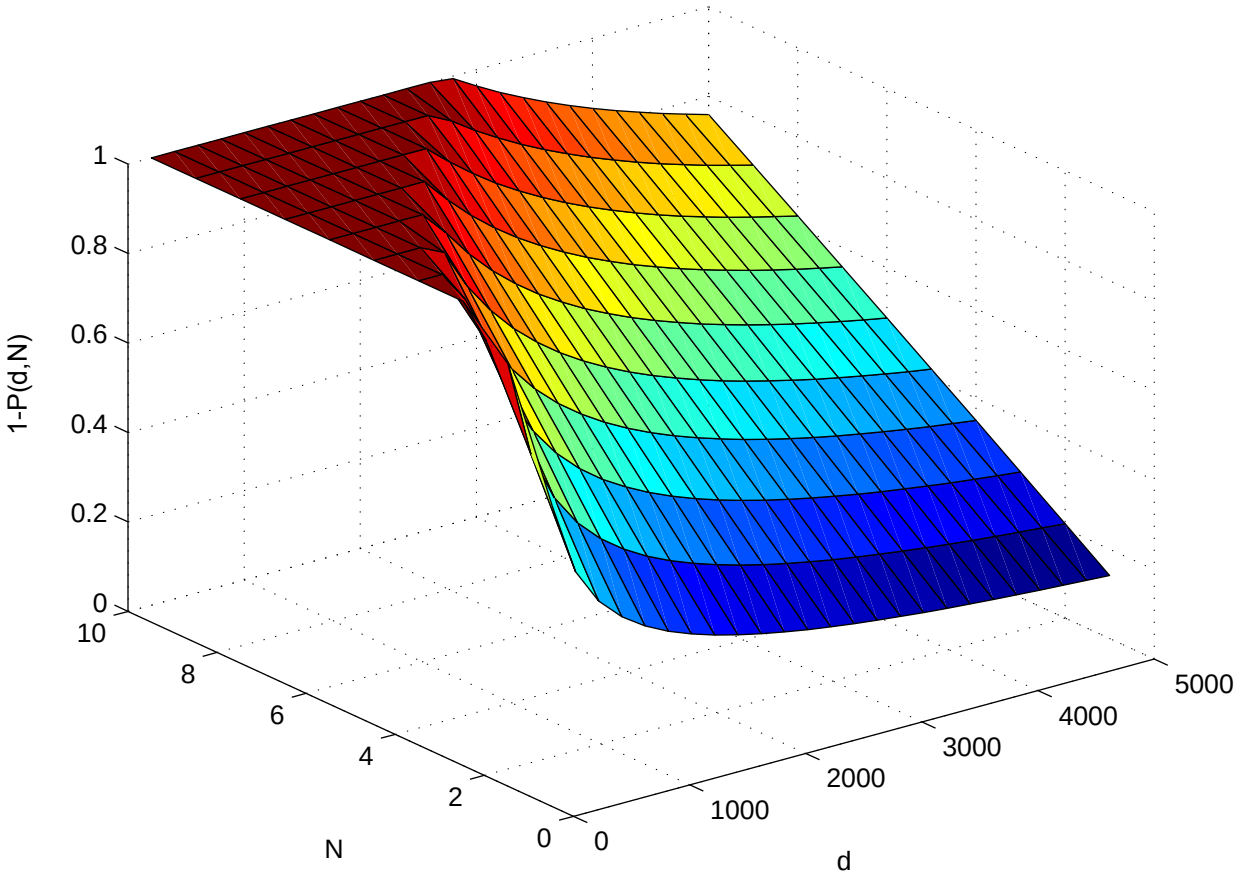


Figure 7: Trade Size, Confirmation Lag and Probabilistic Finality

4.3 Buyers' Optimal Decision

We can now solve for buyers' optimal decision regarding money demand z in the day market and the offer to sellers in the night market. In a monetary equilibrium, $z > 0$ and condition (12) is satisfied with equality. Also, since $\mu \geq 1 > \beta$, standard arguments suggest that the buyer will not bring balances that will not be spent in the night market. That is, $z = d$. So the buyer's decision is given by

$$\max_{d,N,x} -d + (1-\sigma)\frac{\beta}{\mu}d + \sigma[\delta^N \varepsilon u(x) + D_0(d,N)]$$

subject to

$$\frac{x}{1-\tau}\frac{\mu}{\beta} = d[1 - P(d,N)].$$

The values of D_0 and P depend on whether the offer is DS-proof or not:

$$\begin{cases} P(d,N) = D_0(d,N) = 0 & \text{, if } d \leq R(N+1)N, \\ P(d,N) = \frac{\sqrt{\Delta}-(N+1)}{\sqrt{\Delta}} \text{ and } D_0(d,N) = \frac{\beta}{\mu}R\left[\sqrt{\Delta} - (N+1)\right]^2 & \text{, otherwise.} \end{cases}$$

While double spending allows the buyer to receive an additional payoff $D_0(d,N)$, it also tightens the seller's participation constraint.

Note that the buyer's problem in general can have multiple maximizers. For example, sometimes a buyer can be indifferent between a DS and a DS-proof contracts. Similarly, the buyer can be indifferent between a contract with a long confirmation lag and large consumption and one with earlier but smaller consumption. Given R, define the set of optimal money demand by ε as $\Gamma(\varepsilon; R)$. For a given selection from the solution set $\Gamma(\varepsilon; R)$, the aggregate night market spending and money demand are described by

$$D = B\sigma E(d) = B\sigma \int_0^\infty z^*(\varepsilon; R) dF_\varepsilon(\varepsilon) \tag{17}$$

$$Z = BE(z) = B \int_0^\infty z^*(\varepsilon; R) dF_\varepsilon(\varepsilon). \tag{18}$$

4.4 No Double Spending

We now derive a sufficient condition under which double-spending contracts are dominated in equilibrium. Define the nominal interest rate as $i = \mu/\beta - 1$.

Lemma 6. *Only DS-proof contracts are offered if*

$$\delta\varepsilon_{\max}u'(\bar{x}_1)(1-\tau)\frac{3}{4}-1<\frac{i}{\sigma} \tag{19}$$

where $\bar{x}_1=(1-\tau)(\beta/\mu)2R$.

How to interpret this condition? Note that in general, the marginal value of an additional unit of money balances is (proportional to)

$$-i+(1-P)\sigma[\delta^N\varepsilon u'(x)(1-\tau)\mathcal{E}(x)-1]$$

where

$$\begin{aligned}\mathcal{E}(x) &= \frac{\partial x}{\partial d}\frac{d}{x}\\ &= \frac{\partial}{\partial d}[d(1-P(d,N))]\frac{1}{d[1-P(d,N)]}\\ &= \begin{cases}1 & \text{, if } x<\bar{x}\\ 1-\frac{d}{2R\Delta} & \text{, if } x\geq\bar{x}\end{cases}\end{aligned}$$

is the elasticity of consumption with respect to money balances. When the incentive constraint is not binding, $\mathcal{E}=1$. When it is binding, $\mathcal{E}<1$. Define $\bar{x}_N$ as the maximum DS-proof quantity given N. Evaluating at $\bar{x}_N$,

$$\mathcal{E}(\bar{x}_N)=1-\frac{N}{2(1+N)}$$

which is a decreasing function with its maximum equals to $3/4$ when $N=1$. The idea is that when the incentive constraint is binding, a further increase of the trade size will raise the buyer's incentive to double spend after trade, hence lowering the effective value of a marginal dollar.

This condition tends to be satisfied when (i) the cost of bringing the extra money balances is high (i is high), (ii) the probability of spending that extra balances is low (σ is low), and (iii) the utility gain from spending that balances is low (δ, $\varepsilon_{\max}$ are low or $\bar{x}$ is high).

4.5 Equilibrium

We first define an equilibrium in which the cryptocurrency has a positive value and all trades are double-spending proof.

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

1. *money demand and the offer maximizes a buyer's utility;*
2. *the mining choice maximizes a miner's utility;*
3. *the day money market clears;*
4. *Condition (19) is satisfied.*

Notice that nominal balances are growing at rate μ and so do prices. Our definition has only used real balances which stay constant across time. Finally, we define social welfare as the average utility per period or

$$\mathcal{W} = B\sigma \int_0^\infty [\delta^{N(\varepsilon)} \varepsilon u(x(\varepsilon)) - x(\varepsilon)] dF_\varepsilon(\varepsilon) - C(\bar{N} + 1).$$

The first term is the trade surplus, while the last term are the aggregate costs of mining.

4.6 Existence

Lemma 8. *A DS-proof cryptocurrency equilibrium exists for a sufficiently large* B*.*

This lemma establishes the existence of a monetary equilibrium. There are potentially multiple equilibria. If we focus on the equilibrium that supports the maximum level of Z (i.e. the equilibrium with the highest value of money), then we can show that an increase in B will increase the value of money but also increase the cost of mining.

5 Numerical Analysis based on Bitcoin

5.1 Parameterization

We assume that buyers' utility function is

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

with $b \approx 0$. We pick the following parameter values for the benchmark model.

	values	targets
β	0.999916553598325	period length = 1 day
δ	0.999999420487088	$\delta = \beta^{1/(1+\bar{N})}$
μ	1.0003	money growth rate
τ	0.000088	transaction fee
B	6873428	max. no of average-sized transactions
σ	0.0178	velocity per block (block length = 10 mins)
α	1	normalization

The length of a period is a day and the length of each trading session is 10 minutes (i.e. average block time). Setting $\beta = 0.9999$ gives an annual discount factor of 0.97. The average Bitcoin supply in 2015 is 14342502.95. So the money growth rate per day in 2015 is $\mu = (1+25/14342502.95)^{6\times 24} = 1.0003$. Numbers in the following table are averages in 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

We set $\sigma = 0.0178$ to match the average fraction of Bitcoins spent per day, and set $\tau = 0.15596529$ /1769.744292 = 0.000088129 to match the transaction fees data. The average transaction size is $\tau = 1769.744292$ /848.1232877 = 2.086659237. So we set $B = 6873428.441$ which is the maximum number of average-sized transactions that the existing stock of Bitcoin can support.[20] The distribution $F(\varepsilon)$ is set to capture the shape of the empirical distribution of transaction size reported in Ron and Shamir (2013).

Figure (9) plots the density function of the preference shocks ε and Figure (10) plots the optimal N for each ε.

5.2 Effects of Money Growth and Transaction Fees

Before deriving the optimal policy (μ, τ), we first study the equilibrium effects of a partial change in the money growth rate and the transaction fee around the benchmark equilibrium. Given the

[20]This is quite close to the number of blockchain wallet users which is 5439181 in 2015 (Source: blockchain.info, year-end number).

Figure 8: Size Distribution of Bitcoin Transactions (Ron and Shamir, 2013)

Larger or equal to	Smaller than	Number of transactions in the graph of entities	Number of transactions in the graph of addresses
0	0.001	381,846	2,315,582
0.001	0.1	1,647,087	4,127,192
0.1	1	1,553,766	2,930,867
1	10	1,628,485	2,230,077
10	50	1,071,199	1,219,401
50	100	490,392	574,003
100	500	283,152	262,251
500	5,000	70,427	67,338
5,000	20,000	6,309	6,000
20,000	50,000	1,809	1,796
50,000		364	340

Figure 9: Preference Shock Distribution $f(\varepsilon)$

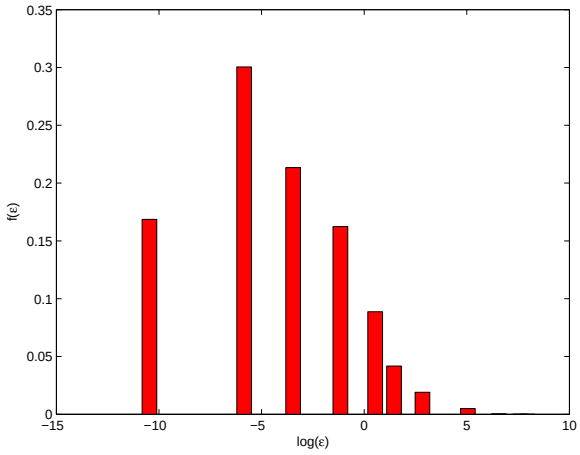


Figure 10: Confirmation Lag $N(\varepsilon)$

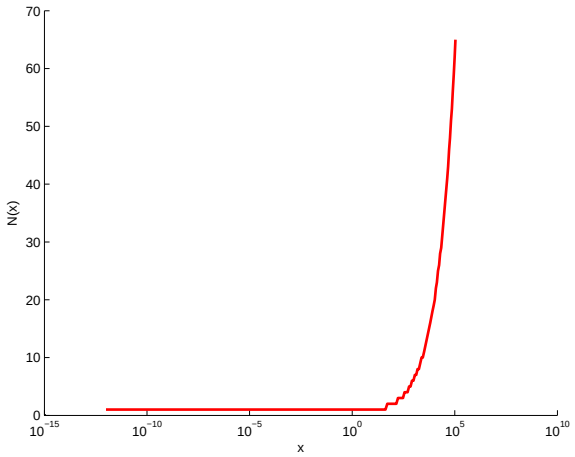
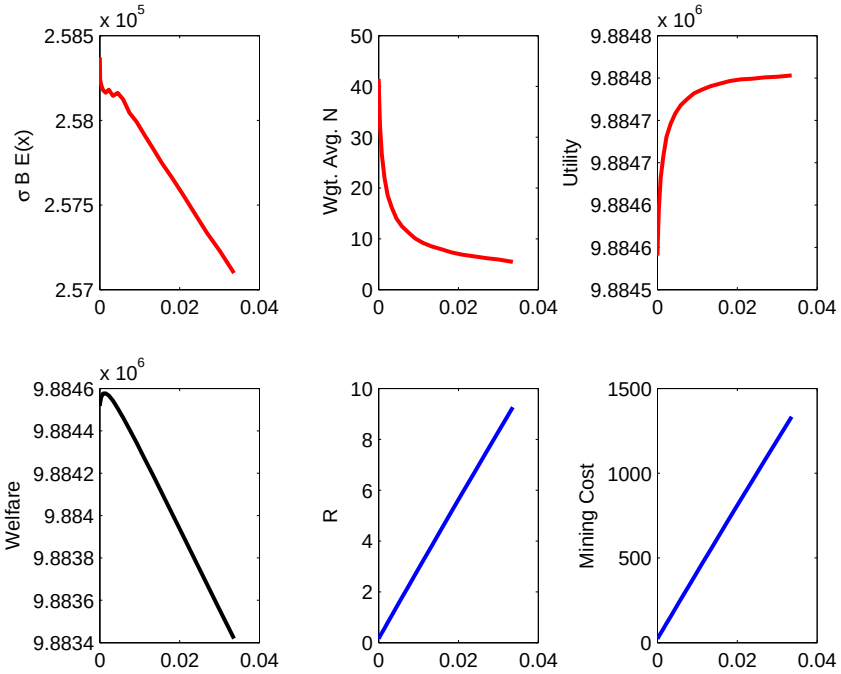


Figure 11: Effects of money growth



benchmark level of τ, Figure (11) shows the effects of μ on aggregate trade, average confirmation lags, utility, welfare, rewards and mining costs. By inducing mining activities, a higher μ lowers confirmation lags but increases inflations. The net effect on consumption and utility is positive. Also, a higher μ raises rewards, computational efforts and overall mining costs. The former effect improves welfare while the latter effect reduces welfare. So the two effects result in a hump shape response of welfare to money growth. Positive inflation is optimal.

Given the benchmark money growth, Figure (12) shows the effects of τ. By inducing mining activities, a higher τ lowers confirmation lags but also distorts consumption. The net effect on consumption and utility is negative, because the benchmark μ is already rather high. Also, a higher τ raises rewards, computational efforts and overall mining costs. Given these two negative effects, τ is always welfare reducing. Given the benchmark μ, zero τ is optimal.

5.3 Efficiency of Cryptocurrency Systems

Given the underlying preference shocks, Table 1 compares the welfare costs (as a fraction of First-best consumption) under different cash and cryptocurrency systems. A cash system under the

Figure 12: Effects of transaction fees

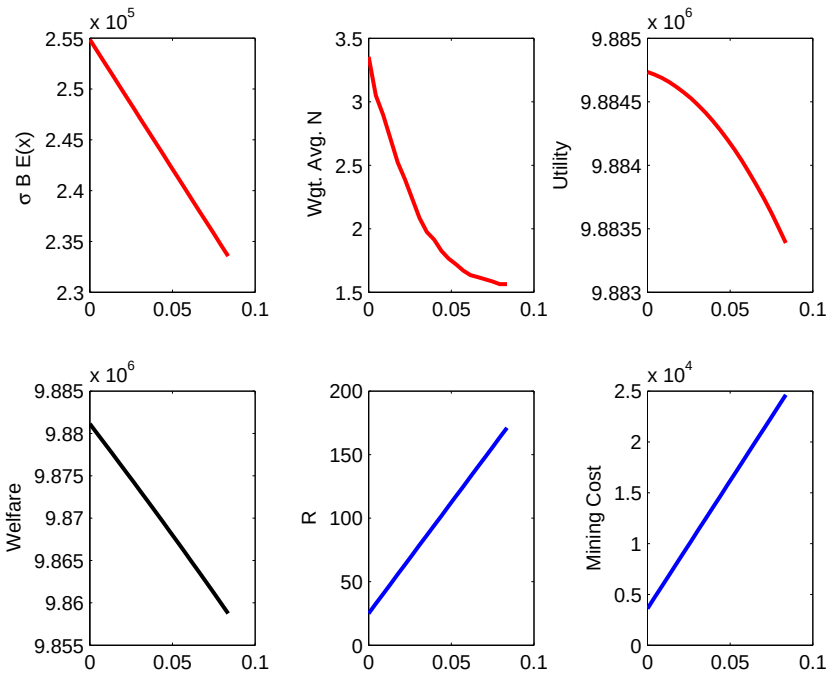


Table 1: Welfare Comparison between Cryptocurrency and Cash Systems

	Welfare Cost (% of comsumption)
Cash (Friedman Rule)	0.000%
Cash (2% inflation)	0.003%
Bitcoin (benchmark)	1.410%
$\mu - 1 = 9.5\%, \tau = 0.0088\%$	mining cost: \$1.57 billions
Bitcoin (optimal policy)	0.080%
$\mu - 1 = 0.17\%, \tau \approx 0.0000\%$	mining cost: \$27.73 millions

Table 2: Welfare Comparison between Retail and Large-value Systems

	Retail Payments (US Debit cards)	Large Value Payments (Fedwire)
avg transaction size	\$38.29	\$6552236
annual volume	59539 millions	135 millions
optimal μ	0.032%	0.58%
optimal τ	0%	0.0021%
confirmation lag	2 mins	13 mins
welfare loss	0.0006%	0.59%
mining cost (per year)	\$3.73 millions	\$18.92 billions

Friedman rule has zero welfare costs. A cash system under 2 percent money growth rate generates a very small welfare cost of 0.003%. In contrast, the cryptocurrency system welfare is much less efficient, generating a welfare loss of 1.4%. The main source is the huge mining costs, which is estimated to be 1.57 billions USD per year. It is inefficient to set the money growth rate and transaction fees too high. The optimal policy is to reduce inflation and transaction fees substantially to discouraging mining. In addition, the optimal policy should rely on money growth instead of transaction fees. The reason why inflation tax is more efficient is that it is paid by all buyers while transaction fees are paid only by traders who are liquidity constrained.

5.4 Best Usage of Cryptocurrency Technology

We now evaluate the efficiency of using the cryptocurrency system for retail and large-value settlement systems. In Table 2, the first column reports the case when we use 2014 US debit cards payment data to calibrate the model. A period is set to be 30 minutes and the block length is 1 minute. First of all, we pick $B = 30.16$ millions to match the number of debit cards.[21] and set $\sigma = 0.540853348$ to match the volume of transactions per card per day. Finally, ε is picked so that the average size of transaction equals \$38.2890. In the second column, we try to calibrate the model to match Fedwire data. We assume $B = 7866$ to match the number of participants in 2014, and set $\sigma = 0.9795$ which is the average volume of transactions for a participant in 30 minutes.

[21]Source: http://www.bis.org/cpmi/publ/d152.pdf

Again, ε is chosen to match the average transaction size.

Recall that the no-double-spending constraint is

$$d \leq RN(1+N).$$

Mining is a public good (captured by $R = Z[(\mu - 1) + \sigma\tau]$), while double spending depends on individual incentives to reverse a particular transactions (captured by d). This implies that a cryptocurrency works best when the volume of transactions is larger relative to the individual transaction size. As shown in the above table, in a retail system the average transaction size is much smaller and the volume is much bigger than in a large-value payment system. As a result, under the optimal design, the required money growth and transaction fees are both lower while the confirmation lag is also shorter. Therefore, the welfare cost and the actual mining cost are both significantly lower.

6 Alternative Consensus Protocols

In this section, we study whether replacing proof-of-work by alternative consensus protocols can improve efficiency.

6.1 Proof of Stake

Under the proof-of-stake (PoS) protocol, the probability that a cryptocurrency holder is granted the right to receive the newly issued currencies (called "minting") and to update the blockchain is proportional to the fraction of currency held. Specifically, we assume that the probability that an agent i wins a block in the night market is equal to

$$\frac{z(i)}{Z}$$

where $z(i)$ is the balances that agent i brought from the day market and Z is the total balances.

Day Value Function

The day value function is given by

$$w(z;\varepsilon) = \max_{\hat{d},d,h} -h + v(\hat{d}, d; \varepsilon) \tag{20}$$

subject to

$$z + h \geq \hat{d} + d \tag{21}$$

where h is the amount of work in the day, d is the real balances to be spent in the night market (if there is a match), and $\hat{d}$ is the real balances that will not be spent in the night but carried to the next day. The only reason for carrying $\hat{d}$ is to increase the chance of wining the block. The FOC is

$$\begin{aligned} 1 &\geq \frac{\partial}{\partial \hat{d}} v(\hat{d}, d; \varepsilon) \\ 1 &\geq \frac{\partial}{\partial d} v(\hat{d}, d; \varepsilon) \end{aligned}$$

with equality whenever money holdings are positive. Note that

$$\begin{aligned} w(z) &= z + W, \\ w'(z) &= 1. \end{aligned}$$

where W is a constant. We want to find a condition under which $\hat{d} = 0$.

Night Value Function

The night value function is

$$\begin{aligned} v(\hat{d}, d) &= \frac{\beta}{\mu}(\hat{d} + d) + \frac{\hat{d} + d}{Z}(\bar{N} + 1)\frac{\beta}{\mu}R \\ &\quad + \sigma[\delta^N \varepsilon u(x) - \frac{\beta}{\mu}d] + \sigma\frac{(\hat{d} + d)^{N+1}}{Z^{N+1}}\frac{\beta}{\mu}d + \beta W. \end{aligned}$$

Here, the first term represents the continuation values of balances in the next day. The second term is the expected minting rewards from $\bar{N}+1$ blocks. When there is a match, the third term captures the buyer's trade surplus and the fourth term is the expected reward from double spending. Here we assume that if an agent wins a block, he can claim the rewards even when the double-spending attack fails. If we relax this assumption, the incentive to double spend will be even lower.

In equilibrium, the participation constraint of the seller is

$$x = \frac{\beta}{\mu} d(1 - \tau)(1 - \frac{(\hat{d^e} + d)^{N+1}}{Z^{N+1}}).$$

Here we use $\hat{d^e}$ to denote the equilibrium value of $\hat{d}$. Since $\hat{d}$ is not observable by the seller, its value does not directly affect x. Of course, in equilibrium $\hat{d^e} = \hat{d}$.

Incentive Constraint

Given a contract (x, d, N), we now consider the buyer's decision on the balances $\hat{d}$.

$$\max_{\hat{d}} -\hat{d} + \frac{\beta}{\mu}(\hat{d}+d) + \frac{\hat{d}+d}{Z}(\bar{N}+1)\frac{\beta}{\mu}R$$
$$+\sigma[\delta^N \varepsilon u(x) - \frac{\beta}{\mu}d] + \sigma\frac{(\hat{d}+d)^{N+1}}{Z^{N+1}}\frac{\beta}{\mu}d + \beta W.$$

The FOC is

$$-1 + \frac{\beta}{\mu} + \frac{1}{Z}(\bar{N}+1)\frac{\beta}{\mu}R + \sigma\frac{(N+1)(\hat{d}+d)^N}{Z^{N+1}}\frac{\beta}{\mu}d \leq 0,$$

or simply

$$-i + \frac{1}{Z}(\bar{N}+1)R + \sigma\frac{(N+1)(\hat{d}+d)^N}{Z^{N+1}}d \leq 0.$$

Since $R = \frac{Z[\sigma\tau+(\mu-1)]}{\bar{N}+1}$, the condition becomes

$$-i + [\sigma\tau + (\mu-1)] + \sigma\frac{(N+1)(\hat{d}+d)^N}{Z^{N+1}}d \leq 0. \tag{22}$$

When this is satisfied with strict inequality, we have $\hat{d} = 0$.

We now look at the choice of d:

$$\max_{d} -d + \frac{\beta}{\mu}(\hat{d}+d) + \frac{\hat{d}+d}{Z}(\bar{N}+1)\frac{\beta}{\mu}R$$
$$+\sigma\left[\delta^N \varepsilon u\left(\frac{\beta}{\mu}d(1-\tau)(1-\frac{(\hat{d}^e+d)^{N+1}}{Z^{N+1}})\right) - \frac{\beta}{\mu}d\right] + \sigma\frac{(\hat{d}+d)^{N+1}}{Z^{N+1}}\frac{\beta}{\mu}d + \beta W.$$

Therefore, the FOC is

$$-1 + \frac{\beta}{\mu} + \frac{1}{Z}(\bar{N}+1)\frac{\beta}{\mu}R + \sigma\frac{(N+1)(\hat{d}+d)^N}{Z^{N+1}}\frac{\beta}{\mu}d$$
$$+\sigma\frac{\beta}{\mu}\left[\delta^N \varepsilon u'[x(\varepsilon)](1-\tau)\left(1 - \frac{(\hat{d}^e+d)^{N+1} + d(N+1)(\hat{d}^e+d)^N}{Z^{N+1}}\right) - 1\right] + \sigma\frac{(\hat{d}+d)^{N+1}}{Z^{N+1}}\frac{\beta}{\mu} = 0,$$

implying

$$-i + [\sigma\tau + (\mu-1)] + \sigma\frac{(N+1)(\hat{d}+d)^N}{Z^{N+1}}d \tag{23}$$
$$+\sigma\left[\delta^N \varepsilon u'[x(\varepsilon)](1-\tau)\left(1 - \frac{(\hat{d}^e+d)^{N+1} + d(N+1)(\hat{d}^e+d)^N}{Z^{N+1}}\right) - 1\right] + \sigma\frac{(\hat{d}+d)^{N+1}}{Z^{N+1}} = 0.$$

Combining conditions (22) and (23), we have $\hat{d} = 0$ when

$$\delta^N \varepsilon u'[x(\varepsilon)](1-\tau)\left(1 - \frac{(\hat{d}^e+d)^{N+1} + d(N+1)(\hat{d}^e+d)^N}{Z^{N+1}}\right) - 1 + \frac{(\hat{d}+d)^{N+1}}{Z^{N+1}} > 0. \tag{24}$$

Below, we want to show that, when $B \to \infty$, we have $d/Z \to 0$ for $\tau = 0$, $N = 0$, $\mu = 1$, $\hat{d} = 0$. Note that the payoff of a buyer who carries balances z into the night market is bounded by

$$v(z;\varepsilon) - v(0;\varepsilon) - z < \sigma\left(\varepsilon u(x_\varepsilon^*) - x_\varepsilon^*\right) - (\frac{\beta}{\mu} - 1)z.$$

Since $\mu > \beta$, there exists a maximum level $\bar{z}$ of balances any buyer would consider carrying into the night market. By the cash-in-advance constraint, we have that $d \leq z \leq \bar{z}$. Therefore,

$$\frac{d}{Z} \leq \frac{\bar{z}}{Z} = \frac{\bar{z}}{B\mathbb{E}(d(\varepsilon))}$$

where $d(\varepsilon)$ is the equilibrium money demand for ε when $\tau = 0$, $N = 0$, $\mu = 1$,$d/Z = 0$. That is,

$$\frac{1}{\beta} - 1 = \sigma[\varepsilon u'(\frac{\beta}{\mu}d) - 1].$$

Obviously, when $B \to \infty$, $\frac{d}{Z} \to 0$. The LHS of condition (24) becomes

$$\varepsilon u'[x(\varepsilon)] - 1 = \frac{1-\beta}{\beta\sigma},$$

which is obviously positive. Hence it is optimal for agents to hold $\hat{d} = 0$. Therefore, when $B \to \infty$, PoS supports the best possible allocation with $\tau = 0$, $N = 0$, $\mu = 1$. Since there is no mining cost, the welfare must be higher than that under PoW.

Theorem 9. *As $B \to \infty$, PoS strictly dominates PoW. Settlement is immediate and final.*

This theorem states that, when the economy is sufficiently large, a cryptocurrency with PoS is as efficient as a cash system with zero inflation. By setting $\mu = 1$ and $\tau = 0$, the minting rewards become zero. When the economy is sufficiently large, the probability that an individual buyer will be chosen to update the block converges to zero. As a result, there is no incentives to double spend even when $N = 0$.

6.2 Practical Byzantine Fault Tolerance (To be added)

7 Appendix

7.1 Micro-foundation for the proof-of-work problem

Miners perform their mining between trading sessions. By investing computing power $q(m)$, the probability that a miner m can solve the computational task within a time interval t is given by an exponential distribution with parameter μ_m

$$F(t) = 1 - e^{-\mu_m t}$$

where $\mu_m = q(m)/D$. The parameter D captures the difficulty of the proof-of-work controlled by the system. The expected time needed to solve the problem is thus given by

$$\frac{D}{q(m)}.$$

Aggregating over all M miners, the first solution among all miners, $\min(\tau_1, \tau_2..., \tau_M)$, is also an exponential random variable with parameter $\sum_{m=1}^{M} \mu_m$. Hence the expected time needed to complete the proof-of-work by the pool of miners is[22]

$$\frac{D}{\sum_{m=1}^{M} q(m)}.$$

Furthermore, any particular miner m will be the first one to solve the proof-of-work problem with probability

$$\rho_n = \frac{q(n)}{\sum_{m=1}^{M} q(m)}.$$

7.2 Proof of Lemma 6

DS-proof contract

Fix N. The optimal DS-proof contract is a solution to

$$\max_{d,x} -d + (1-\sigma)\frac{\beta}{\mu}d + \sigma\delta^N \varepsilon u(x) \tag{25}$$

[22]In practice, the parameter D is often adjusted to maintain a constant time for completing the proof-of-work problem given any changes in the total computational power.

subject to

$$\frac{x}{1-\tau}\frac{\mu}{\beta} = d \tag{26}$$

$$d \leq R(N+1)N \tag{27}$$

Note that the participation constraint of the seller is always binding.

If the incentive constraint is not binding, then the FOC is then given by

$$1 = \frac{\sigma}{i}[\delta^N \varepsilon u'(x)(1-\tau) - 1], \tag{28}$$

where $i = \mu/\beta - 1$ is the nominal interest rate. We denote this solution by (x^*, d^*) where it is understood that the solution depends on N.

If the incentive constraint is binding, then we have that

$$\bar{d} = R(N+1)N \tag{29}$$

$$\bar{x}(N) = \frac{\beta}{\mu}(1-\tau)R(N+1)N \tag{30}$$

and

$$1 < \frac{\sigma}{i}[\delta^N \varepsilon u'(\bar{x}(N))(1-\tau) - 1] \tag{31}$$

We denote this solution by $(\bar{x}, \bar{d})$ where it is understood that the solution depends on N.

DS contract

Fix N. The optimal DS contract is a solution of the problem

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

subject to

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

$$1 - P(d,N) = \frac{(N+1)}{\sqrt{\Delta}} \tag{34}$$

$$d \geq R(N+1)N \tag{35}$$

Note that the participation constraint for the seller is binding again.

Some preliminaries first.

$$\frac{\partial D_0(d,N)}{\partial d} = \frac{\beta}{\mu} P(d,N) \tag{36}$$

$$\frac{\partial P(d,N)}{\partial d} = \frac{1}{2R\Delta}(1 - P(d,N)) \tag{37}$$

$$\frac{\partial d(1-P(d,N))}{\partial d} = \left(1 - \frac{d}{2R\Delta}\right)(1 - P(d,N)) \tag{38}$$

We look next at the function $d(1 - P(d,N))$. First, note that this expression is only valid when $d \geq \bar{d}$. Its minimum is achieved at $\bar{d}$. It is strictly increasing in d and strictly concave.

Differentiating the objective function w.r.t. to d, we obtain up to a factor of $\frac{\beta}{\mu}$

$$-i + \sigma(1 - P(d,N)) \left(\delta^N \varepsilon u'(x)(1-\tau)\left(1 - \frac{d}{2R\Delta}\right) - 1\right)$$

Case 1:

Suppose now that for a given N we have $x^* < \bar{x}$, that is at the best DS-proof contract the constraint is not binding.

Since $(1 - P(d,N)) < 1$ and $(1 - d/(2R\Delta)) < 1$, we have immediately that the objective function is decreasing in d. Hence, the best DS contract has $d = \bar{d}$. But this is worse than (x^*, d^*).

Case 2:

Suppose now that for a given N we have $\bar{x} < x^*$ so that the constraint is binding for the optimal DS-proof contract.

Note first that the objective function is strictly concave. This implies that there is a unique maximizer and – by the previous argument – the solution needs to satisfy $\hat{x} \in [\bar{x}, x^*)$.

A sufficient condition is thus that the objective function is decreasing at $\bar{x}$. The first-order condition at $\bar{x}$ is given by

$$-i + \sigma\left(\delta^N \varepsilon u'(\bar{x})(1-\tau)\left(1 - \frac{1}{2}\frac{N}{N+1}\right) - 1\right) \tag{39}$$

Finally, note that this equation is strictly decreasing in N as $u'(\bar{x})$ is decreasing in N. Hence, a sufficient condition is that the objective function is decreasing at $N = 1$ and $\bar{d}$ or, equivalently, that

$$\delta\varepsilon u'[(1-\tau)(\beta/\mu)2R](1-\tau)\frac{3}{4} \leq \frac{i+\sigma}{\sigma}.$$

So we can conclude that DS is never optimal when

$$\delta\varepsilon_{\max}u'(\bar{x}(1))(1-\tau)\frac{3}{4}-1<\frac{i}{\sigma}$$

where $\bar{x}(1)=(1-\tau)(\beta/\mu)2R$.

7.3 Proof of Lemma 8

We briefly sketch the proof here. We first show that the individual money demand given R, $z^*(\varepsilon;R)$, is an upper-hemicontinuous correspondence function. Since integration preserves upper-hemicontinuity, the aggregate money demand correspondence

$$S^Z(R)=\{B\int_0^\infty z^*(\varepsilon;R)dF_\varepsilon(\varepsilon)|\ z^*(\varepsilon;R)\in\Gamma(\varepsilon;R)\}$$

is also upper-hemicontinuous. Note that in a symmetric equilibrium where agents of the same type choose the same money demand, this set is non-convex in general. However, we can convexify it by allowing agents of the same type to pick *asymmetric* choices. As a result, the equilibrium money demand becomes the convex hull

$$\tilde{S}^Z(R)=conv[S^Z(R)]=\{B\int_0^\infty\sum_{i=1}^{|\Gamma(\varepsilon;R)|}\theta_i(\varepsilon;R)z^*(\varepsilon;R)dF_\varepsilon(\varepsilon)|\theta_i(\varepsilon;R)\geq 0,\ \sum_{i=1}^{|\Gamma(\varepsilon;R|}\theta_i(\varepsilon;R)=1,\ z^*(\varepsilon;R)\in\Gamma(\varepsilon;R)\}.$$

Define a correspondence function

$$\Omega(R)=B\frac{[(\mu-1)+\sigma\tau]}{\bar{N}+1}\tilde{S}^Z(R),$$

which, taking R as exogenous, gives the per block mining reward generated by agents' choices. Next, we want to apply Kakutani's fixed point theorem to show existence: Let S_R be a non-empty, compact and convex subset of some Euclidean space R^n. Let $\Omega: S_R\to 2^{S_R}$ be a set-valued function on S_R with a closed graph and the property that $\Omega(R)$ is non-empty and convex for all $R\in S_R$. Then Ω has a fixed point.

We do it in a few steps:

1. Define a lower bound for S_R. Pick a small $R_{\min}>0$. We have $z^*(\varepsilon;R_{\min})>0$ for all ε. We can then pick B sufficiently large so that

   $$B\inf_{R\geq R_{\min}}\{\frac{[(\mu-1)+\sigma\tau]}{\bar{N}+1}\tilde{S}^Z(R)\}>R_{\min}.$$

   We denote this as $\bar{B}$. This ensures that for all $R\geq R_{\min}$, $\Omega(R)>R_{\min}$.

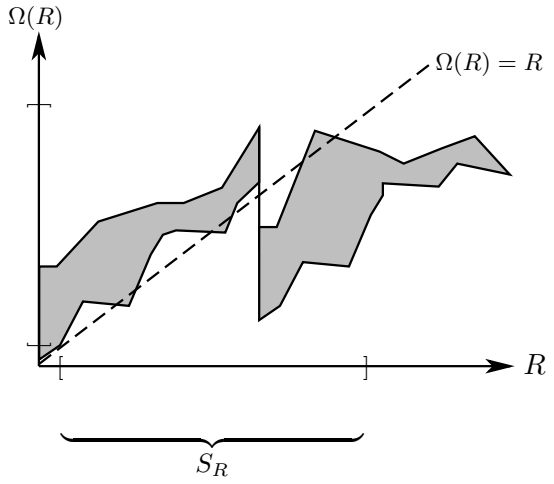


Figure 13: $\Omega(R)$

2. Define an upper bound for S_R. Set $B = \bar{B}$. Notice that the net surplus of a buyer bringing balances z into the night market is bounded by

$$\beta\sigma[\varepsilon u(x^*) - x^*] - (1 - \frac{\beta}{\mu})z.$$

Since $\mu \geq 1 > \beta$, there exists a maximum level $\bar{z}(\varepsilon)$ of balances any buyer would consider carrying from the day market. Therefore, for all

$$R, \Omega(R) \leq R_{\max} \equiv \bar{B}\frac{[(\mu - 1) + \sigma\tau]}{\bar{N} + 1}\int_0^\infty \bar{z}(\varepsilon)dF_\varepsilon(\varepsilon).$$

3. Define $S_R = [R_{\min}, R_{\max}]$. It is a non-empty, compact and convex subset. By construction, for any $R \in S_R$, $\Omega(R) \subset 2^{S_R}$.

4. Since $\Omega(R)$ is a non-empty, closed and convex set for all R, and Ω is upper hemicontinuous, we know that $\Omega(R)$ has a closed graph.

5. Therefore, Ω has a fixed point, which defines a (potentially asymmetric) cryptocurrency equilibrium.

■

References

Agarwal, R. and M. Kimball, (2015). "Breaking through the Zero Lower Bound."IMF Working Paper WP/15/224. Washington: International Monetary Fund, October.

Chiu, J., and T. Wong. (2014). "E-Money: Efficiency, Stability and Optimal Policy." Bank of Canada Working Paper No. 2014-16.

Chiu, J., and T. Wong. (2015). "On the Essentiality of E-Money." Bank of Canada Working Paper No. 2015-43.

Fernández-Villaverde, J. and D. Sanches, (2016). "Can currency competition work?"National Bureau of Economic Research No. w22157.

Gandal, N., and H. Halaburda (2014). "Competition in the Cryptocurrency Market."Bank of Canada Working Paper No. 2014-33..

Gans, J., and H. Halaburda, (2013). "Some Economics of Private Digital Currency," Bank of Canada Working Paper No. 2013-38.

Glaser, F., K. Zimmermann, M. Haferkorn, M. Weber and M. Siering, (2014). "Bitcoin – Asset or Currency? Revealing Users' Hidden Intentions," available at http://papers.ssrn.com/sol3/papers.cfm?abstract id=2425247

Lagos, L. and R. Wright, (2005). "A unified framework for monetary theory and policy analysis, " Journal of political Economy, 113.3 (2005): 463-484.

Moore, T. and N. Christin, (2013). "Beware the Middleman: Empirical Analysis of Bitcoin-Exchange Risk," in Financial Cryptography and Data Security.

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.

Yermack, D. (2013) Is Bitcoin a real currency? An economic appraisal. No. w19747. National Bureau of Economic Research.