A Note on Cryptocurrency Stabilisation: Seigniorage Shares

Robert Sams
rs@clearmatics.com

First Version: October 24, 2014
This Version: April 28, 2015

Abstract

Cryptocurrencies like Bitcoin govern the supply of coin through simple and *deterministic* coin growth rules. As a result, unanticipated changes in coin demand are reflected in changes in coin price, causing volatility and discouraging usage of coin as media-of-exchange. We argue that next-generation cryptocurrencies should incorporate an *elastic* supply rule that adjust the quantity of coin supply proportionately to changes in coin market value. There are two difficult problems to solve in order to implement such a scheme. This note outlines a solution to one of those problems, and offers some suggestions on how the other problem might be solved.

Coin Supply and Volatility

> *The main volatility in bitcoin comes from variability in speculation, which in turn is due to the genuine uncertainty about its future. More efficient liquidity mechanisms don't help reduce genuine uncertainty.[5]*
> **Nick Szabo**

Cryptocurrencies like Bitcoin govern the supply of coin through simple and deterministic coin supply rules. By "deterministic" I mean that the growth rate of coin supply is completely specified in advance and is not influenced by facts outside of the system.[1] This is a significant departure from even pure commodity money systems, as the supply of a precious metal is responsive to price changes that cross the marginal cost of pulling the stuff out of the ground.

If a cryptocurrency system aims to be a general medium-of-exchange, deterministic coin supply is a bug rather than a feature. This is because changes in coin demand get translated into changes in coin price, making price volatility proportional to demand volatility. But that is only a first order effect, for *expectations* of future levels of coin demand give rise to speculation. If the expectations of the long-term rate of coin adoption are significantly greater than the rate of coin supply growth, people will buy and hold coin in anticipation of future adoption, driving up the current price of coin.

It is the nature of markets to push expectations about the future into current prices. Deterministic money supply combined with uncertain future money demand conspire to make the market price of a coin a sort of prediction market on its own future adoption. Since rates of future adoption are highly uncertain, high volatility is inevitable, as expectations wax and wane with coin-related news, and the coin market rationalises high expected returns with high volatility (no free lunch).

The problem is that high levels of volatility deter people from using coin as a medium of exchange. Given that the bullish case for buy-and-hold coin speculation is based on expectations of substantial medium-of-exchange usage in future, it might be conjectured that deterministic money supply rules are self-defeating.

[1] This is not completely correct. Accelerating growth in the hashrate means that the average interval of block times is lower than the 10 minutes targeted by Bitcoin's difficulty adjustment rule. But this influence is marginal.

Coin Demand

In light of this, it makes sense to analyse coin demand into two types:

- Transactional Coin Demand CD_T
- Speculative Coin Demand CD_S

Transactional coin demand is the desire to hold a certain quantity of coins for the purpose of making transactions. You can think of CD_T as the sum of all coin balances held with such a motive. Speculative money demand is the desire to hold a certain quantity of coins in the expectation that its price will appreciate. You can think of CD_S as the sum of all coin balances held as part of a portfolio of savings.

A given individual is likely to possesses both speculative and transactional motives for holding coin and he may not even mentally demarcate his coin balance to reflect the different motivations. But for analytical and empirical purposes, it makes sense to model coin demand as if he does and analyse aggregate coin demand as:

$$CD = CD_T + CD_S \tag{1}$$

In macroeconomic research, speculative demand for a fiat currency is often conjectured to be driven by factors such as the level of interest rates and the degree of risk aversion over risky assets. But coins are different. Because they are young and the potential growth rate of adoption is substantial, coins are among the most volatile of assets. What drives speculative demand is anyone's guess. On the most charitable, rational-expectations analysis, CD_S is driven by expectations of future levels of CD_T. Less charitably, it is driven by the evolution of the coin price itself, trend-following behaviour and "greater-fool" beliefs.

The problem is that CD_T itself is negatively influenced by the level of volatility in coin price. Enthusiasts and early-adopters notwithstanding, most

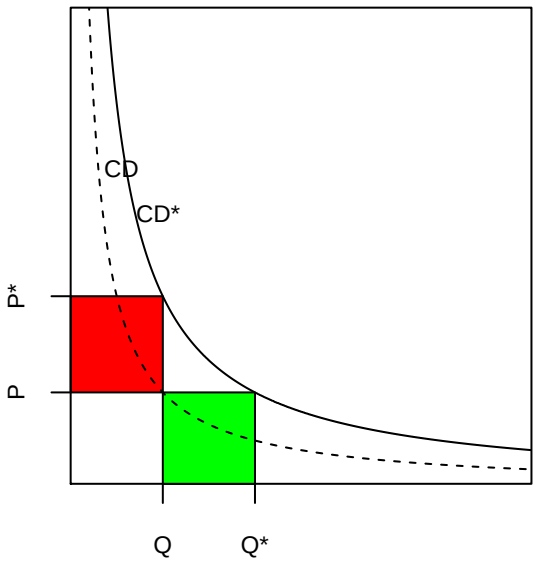

Figure 1: Money Demand

people prefer to hold stable medium-of-exchange over volatile media simply because they are risk-adverse with respect to wealth. I think the socio-economic make-up of the cryptocurrency community (primarily educated and affluent) under-estimates the relevance of risk-aversion to money demand. The poorer you are, the higher percentage of your wealth is held in money, and when entertaining prospects of mass adoption, it's important to remember that the Gini coefficient over money is lower than Gini coefficient over total wealth.

And risk aversion isn't the only reason for why we should expect CD_T to be inversely related to the volatility of coin price. Volatility also generates transaction costs. Substantial declines in coin price require the conversion of fiat currency into coins, and substantial increases in coin price require conversion of coins into fiat currency (or some other asset) in order to maintain the desired balance of coin purchasing power. Both the *frequency* of such re-balancing, and the *cost-per-transaction* of each re-balancing are correlated with coin volatility.

Coin demand is a quantity of *purchasing power* rather than a quantity of coin. That is,

$$CD = P \times Q \tag{2}$$

, where P is a coin's purchasing power (coin price of a basket of goods and services) and Q is the quantity of coins demanded with respect to a given coin price. Unlike the demand curve for ordinary goods, where price elasticity is an empirical matter, the relationship between quantity of coins demanded and coin price can be postulated apriori as in CD in figure 1.

In a protocol like Bitcoin's, when money demand shifts e.g., from CD to $CD*$, this results in a change in price from P to $P*$, as represented by the red box.

Elastic Coin Supply

But in a coin stabilisation scheme, changes in coin price stand proxy for changes in coin demand, and coin supply changes in response to changes in coin price. The idea is that an $X\%$ change in coin price, followed by an $X\%$ change in coin supply, will return coin price to its initial value, as as represented by the green box.

So the core operational principle of a protocol that aims to stabilise the market value of coin is the following rule: **at the end of some pre-defined interval of time, if the change in coin price over the interval is $X\%$, change coin supply by $X\%$.**

More specifically, let's call that interval of time the *rebase period*, defined as every n blocks. The coin supply rule mandates that:

$$Q_i = Q_{i-1} \times \frac{P_i}{P_{i-1}} \tag{3}$$

$$\Delta_i = Q_i - Q_{i-1} \tag{4}$$

, where i is the i-th rebase period, Q is coin supply and P is coin price.

Two hard problems

How do we actually implement such a scheme in a cryptocurrency? There are two hard problems to solve here:

1. How can P_i be represented inside the network in a way that requires minimal trust?

2. How is Δ_i distributed?

The first problem is hard because however the protocol defines P_i (it may be defined as the coin price of an index of commodities, consumer goods.. or simply the USD price of coin), the variable will be a fact about the world outside the system that needs to be represented inside the system via some trust-minimising mechanism. What kind of mechanism can have that property is non-obvious, to say the least. We will discuss some strategies for solving this problem in section at the end of this note, but that problem isn't the focus of this note.

Here we are going to outline a solution to the second problem. Most cryptocurrency systems only increase coin supply, and distribution is done via the mining award. But in a stabilisation scheme, even if $E[\Delta_i]$ is positive, there are times when Δ_i is negative, and coin stabilisation needs a mechanism for *reducing* the coin supply as well as increasing it, so the mining award channel isn't a solution to the problem of how Δ_i is distributed.

How *not* to distribute Δ_i

One simple solution is to distribute Δ_i pro-rata over all coin balances. This is the approach advocated by Ametrano in his creative coin stabilisation scheme dubbed "Hayek Money"[1]. This approach has the virtue of simplicity. All wallet balances are simply multiplied by Q_i/Q_{i-1} in each period to arrive at a new wallet balance. This is very easy to implement technically, the protocol just stipulates the calculation of a rebase factor that is included in each block header. (I think that this is how Friecoin implements its demmurage rule.) It may seem a little awkward that the nominal value of one's wallet fluctuates with changes in money demand, but that might be a tolerable price to pay for a system that achieved coin price stability.

The problem is that this scheme only stabilises *coin price*, it doesn't stabilise the *purchasing power* of a wallet balance. Recall the three functions of money:

1. Unit-of-Account

2. Store-of-Value

3. Medium-of-Exchange

Price stability is not only about stabilising the unit-of-account, but also stabilising money's store-of-value. Hayek money is designed to address the former, not the latter. It merely trades a fixed wallet balance with fluctuating coin price for a fixed coin price with fluctuating wallet balance. The net effect is that the purchasing power of a Hayek Money wallet is just as volatile as a Bitcoin wallet balance. So the self-defeating dynamic of CD_S driving out CD_T remains.

One attempt to address this problem with Hayek Money is Morini's concept of Inv ("investment") wallets and Sav ("savings") wallets [4]. In short, the idea is to bifurcate coin supply into two different categories. Those coins that are in Sav wallets are immunized from any changes in coin supply, and Δ_i is instead distributed pro-rata over those coins that are in Inv wallets. Users choose how to distribute their coin between these two wallets.

This gets us closer to a solution. We can now see that a stable cryptocurrency requires that some party is willing to assume the risk of absorbing negative Δ_i in exchange for the option of receiving positive Δ_i. But the Inv/Sav wallet solution still fails. If there is any predictability in Δ_{t+1} at time t, then users will empty their Inv wallets when $E[\Delta_{t+1}] < 0$ and empty their Sav wallets when $E[\Delta_{t+1}] > 0$. The problem with this solution is that users have the option of moving coin between the two wallets at *parity*.

Seigniorage Shares

The solution to coin distribution offered here is different. I suggest that there needs to be two types of coin: coin that acts like money and coin that acts like shares in the system's seigniorage. For short, we'll just call these *coins* and *shares*. Coins and shares are identical in all respects (transaction verification works via ECDSA signature, etc) except for the process that regulates their respective supply.

Coins are the object of stabilisation, and Δ_i of coin is distributed to the holders of shares. When coin supply needs to increase, coinbase is distributed to share holders in exchange for a certain percentage of shares, which are destroyed (coin supply increases, share supply decreases). When coin supply needs to decrease, sharebase is distributed to coin holders in exchange for a certain percentage of coin, which are destroyed (coin supply decreases, share supply increases).

The mechanism of these shares-for-coin and coin-for-shares swaps is a voluntary one, a decentralised auction the rules of which are written into the protocol. The quantity of coins to create or destroy is defined via the Δ_i process in equations 3 and 4, and there is an auction at the end of every rebase period.

When Δ_i is positive (new coins need to be created) a *coin auction* for Δ_i coins begins at the block or ledger set defining the start of rebase period $i+1$. Any holder of shares can bid for coins by signing and broadcasting a special TX describing the quantity of shares he is willing to trade for coin, and the *minimum* coins/shares price that he will accept. Winning bids are filled at whatever coins/shares price P^s that clears the quantity to be sold. So Δ_i new coins are distributed to the winning bidders and Δ_i/P^s shares are destroyed.[2]

When Δ_i is negative (some existing coins need to be destroyed), a *share auction* of Δ_i *worth of shares* begins. Holders of coin can submit bids to purchase shares, signing a special TX describing the quantity of coins he is will to trade for shares, and the *maximum* coins/shares price P^s that he will accept. Winning bids are filled at whatever price clears the quantity to be sold. So Δ_i/P^s new shares are distributed to the winning bidders and Δ_i coins are destroyed.

[2]Defining the exact details of a decentralised auction is of course a subject in itself. In order to prevent front-running by validating nodes, the auction will have to take place over three periods. In the first period, bids are hashed and broadcast, and consensus is achieved on the existence of the encrypted orders. In the second period, bidders will broadcast the unencrypted order contents and nonce, and consensus is achieved on those by validating that the submissions in the second period hash to the digests submitted in the first period. In the third period, validating nodes validate the spends and and apply the auction protocol to determine winning bids and P^S.

If the long-run demand for coin is positive, coin supply will increase with demand, but the quantity of shares will get increasingly scarce.

Valuing Seigniorage Shares

At first this scheme looks exotic, as if we need some model of "scarcity value" in order to estimate a fair value shares. But this isn't the case at all, for a position in shares is really just a claim on future coin supply growth and can be valued as if it were an income-generating asset.

Consider the following investment strategy. When Δ_i is positive, the investor sells $\Delta_i/(P_i^s Q_i^s)$ percent of his position in the auction; when Δ_i is negative, he increases his position by that percentage by buying shares in the auction. As a result, the investor maintains a position in a *fixed percentage* of the outstanding share supply and therefore has a claim on that fixed percentage of $\Delta_1, \Delta_2, \ldots$ in perpetuity. Therefore, the price of shares is nothing more than the Net Present Value of that income stream. So share price at time t is:

$$P_t^s = \frac{1}{Q_t^s} \sum_{i=t}^{\infty} \frac{\Delta_i}{(1+r_i)^i} \tag{5}$$

, where r_i is the discount rate applied to the seigniorage $i-th$ periods in future.

As with any NPV analysis, r_i must be above the growth rate of Δ_i for the series to converge to a finite value, and both of those variables are of course subjective forecasts of market participants. As with any market in cash flow perpetuities (eg, stocks), valuation metrics can be devised that calibrate forecasts of these variables to historical series of Δ_t and P_t.

If the coin is designed along the lines of a proof-of-work blockchain with a block award, then the numerator in the summation term will need to instead be $\Delta_i - \alpha_i(N)$, where α is the block award in coin and N is the number of blocks in the rebase period. Stable coin schemes have the nice side-effect that the market value of the block award (and therefore, its contribution to to network security) can be defined explicitly, instead of fluctuating with the price of coin, as is the case with a protocol like Bitcoin.

With this dual model of coins and shares, speculators now have a market that they can actually *value*, and we can now exploit the dual motivations for money demand; coins are the object of CD_T, shares are the object of CD_S, and the Janus-faced nature of CD can work in harmony like a monetary Yin and yang rather than CD_S chasing away CD_T, as is the case with monocoin schemes with deterministic coin supply rules.

Decentralised Monetary Policy

Critics of central banking (and I am among them) too often extend their criticism of the institutional failures of centralised and discretionary monetary policy to the very concept of monetary policy itself.

Any monetary regime has a monetary policy. A commodity money regime like a gold-standard has a monetary policy dictated by the cost of pulling new supply of the commodity out of the ground. It has the virtue of a being a monetary policy driven by rules rather than human discretion. But it has the downside of fixing the supply function to an arbitrary process–cost of mining gold–that may not serve the goal of stabilising the purchasing power of money particularly well. Bitcoin has a monetary policy too, but it's arguably worse than the monetary policy of commodity money, as its supply function isn't influenced by the value of Bitcoin at all.[3]

In a sense, this dual model of coins and shares embodies the functionality of a fiat money central bank–without the centralisation...or the bank[4]. Setting aside the complexity of monetary policy transmission in an economy with fractional reserve banking, the essence of a central bank's operations is the use of its balance sheet to adjust the supply of money. Money supply is expanded by purchasing assets with newly created money. Money supply is decreased by selling assets, thereby extinguishing part of the money supply. For all the mystique that surrounds central banking, and the current fashion for targeting interest rates rather than money supply, this is the essence of what a central bank does.

In this way, a central bank can increase the money supply without limit. But its ability to shrink the money supply is constrained by the value of the *assets* it holds. Seigniorage shares are like the asset side of a central bank's balance sheet. The market capitalisation of shares at any point in time fixes the upper limit on how much the coin supply can be reduced.

Solving the Other Problem

So far we have just assumed that our network has a mechanisms for achieving consensus on what P_i is. This is a separate hard problem in its own right. Here's a brief sketch on how I think this problem might be solved.

Firstly, I think that there are two families of design here, one that works with *exogenous* data sources for P_i and a more restrictive strategy that defines P_i in terms of information that is purely *endogenous* to the network itself.

Without specifying what types of values P_i could be, we've sort of assumed the *exogenous* design in this note. The idea here is that, P_i would be defined as some specific price index,

$$P_i = \frac{p_1^1 w^1 + \ldots + p_1^n w^n}{p_i^1 w^1 + \ldots + p_i^n w^n} \tag{6}$$

, where p^j is the coin market price one of n specific goods/assets and w^j is its index weight. For example, the index might consist of the coin prices of a basket of commodities (crude oil, wheat, copper, etc). The goal here is to make the value of coin stable *with respect to* some pre-defined target level of P_i, e.g., $P_i \approx 1$ for all times i.

[3] except at the margin, when non-linear growth in network hash power causes block intervals to average below 10mins.

[4] Even calling central banks "banks" is an anachronism, homage paid to their historical evolution, for in the current fiat money world, a central bank is unique in being the only institution where the "liability" side of its balance sheet is a liability in name only; fiat money isn't a claim on anything, by definition, and the holders of national bank notes are not creditors of the central bank like the holders of a bank deposit are a creditors of a commercial bank.

Exogenous models

One strategy for solving this problem for exogenous designs is a "Schelling point" scheme. The goal is to have a mechanism to incentivise people to submit accurate estimates of P_i to the network.

One mechanism for doing this is to have a periodic *Shelling competition*, where people submit encrypted estimates of P_i to the network, along with F coins, the price for participating in this mechanism. The incentive for participating is that after all the submissions are collected and decrypted, everyone who submitted a value inside the inner two quartiles of the estimate distribution wins $2 \times F$, whilst all those with estimates in the outside two quartiles lose F. So the incentive is to bet what you think the consensus will be.

The idea here is that the "rational" expectation of what consensus will be is whatever is common knowledge of the *salient* answer. A Schelling point is a qualitative equilibrium solution based on common knowledge of salience. In our context, the hope is that *truth* is the Schelling point.

Various suggestions along these lines have been suggested before by Sams[3], Buterin[2], Ametrano[1] and probably others. I'm uneasy about the robustness of the truth-as-Schelling-point assumption. For example, what if the majority of participants in the competition would prefer the consensus to be different from the truth, and this was common knowledge? Why would truth be the Schelling point in this scenario?

We can do better than this by exploiting the incentives of share holders. Share holders want to maximize the value of of shares P^s. And there is a compelling argument that the value of shares is maximised when the P_i communicated to the network is the truth. Here is the argument.

First, let's simplify the set-up a bit (hopefully, without loss of generality) and say that expected demand growth for coin is g, so

$$\Delta_t = \Delta_{t-1}(1+g) \tag{7}$$

, where $g < r$. The sum of infinite summation in equation 5 can then be defined in terms of g and r and the equation reduces to:

$$P_t^s = \frac{1}{Q_t^s}\frac{\Delta_t}{r-g} \tag{8}$$

.

Now, the question is: if shareholders had the power to represent P_i to the network, do they have an incentive to be untruthful? Let ρ_i be the true market value of the coin index and let $\epsilon = P_i/\rho_i$ be the measure of untruthfullness. It might be argued that share holders have an incentive to make ϵ large, that is, to exagerate the value of coin in order to force the network to bid for more shares and increase the current period's coin "dividend".

If we hold MD constant, the first implication of this policy is that the value of coin will decline with the size of the error,

$$\hat{P}_t = P_t\frac{1}{\epsilon} \tag{9}$$

, this follows from equations 2 and 3.

Endogenous models

An entirely different strategy is that which I call an endogenous stabilisation model. Stabilising coin with respect to the coin market value of some index of goods/assets isn't the only form that stabilisation can take. The goal isn't indexing per-se, but stability itself. As we intimated in the previous section, what coin should ideally be stable against isn't necessarily a known parameter and can change through time. But throughout this note we have implicitly defined "stability" in a relative sense, stability with respect to a price index. But why not make the goal some notion of robust stability, a coin that is usually stable with respect to any randomly selected, broad index of goods/assets.

And perhaps information endogenous to the protocol itself is sufficient to achieve this more general goal. If the coin is based on a proof-of-work blockchain, two candidate variables stand out: the average level of *transaction fees* and hashing *difficulty*. Now what information these two variables contain about the outside world depends upon what particular fee model and hashing algorithm the protocol defines. But I

believe they do contain information relevant to coin stability.

Fees. If there is no block size limit, or the limit doesn't generally force miners to ration transaction inclusion, then we can postulate that fees will converge to the average cost of validating transactions times N, the number of validating nodes. So holding constant the computational cost of transaction validation, it stands to reason that changes in average fee levels signal either changes in coin value and/or changes in N.

Difficulty. Similar pattern. Holding GHs/kWh constant, a change in difficulty signals change in coin value, as hashing power is turned on/off until hashing costs equal the market value of mining award.

The market information contained in these two variables has been hugely obscured by the explosive progress in hardware optimisation of hashing. But is it not reasonable to think that rates of optimisation will slow down to roughly predictable levels and eventually be governed by the same constraints (photo-lithography, etc) that CPU and GPU design is subject to?

If so, perhaps the most robust solution is to define P_i in terms of fees and difficulty, deflated by some hard-coded Moore's Law-like assumption. This would avoid the need to represent market facts about the outside world at all, keeping the stable coin scheme autonomous and self-referential.

But all of these suggestions for tackling problem 1 are speculative and I have no convictions here whatsoever. This is a hard problem.

References

[1] Ferdinando M. Ametrano, *Hayek Money: The Cryptocurrency Price Stability Solution (August 19, 2014)*. Located at `http://ssrn.com/abstract=2425270` or `http://dx.doi.org/10.2139/ssrn.2425270`

[2] Vitalik Buterin, *SchellingCoin: A Minimal-Trust Universal Data Feed (March 28, 2014)*. Located at `https://blog.ethereum.org/2014/03/28/schellingcoin-a-minimal-trust-universal-data-feed`

[3] Robert Sams, *What would a trusted data feed look like (February 5, 2014)[Reply to forum posting]*. Located at `https://forum.ethereum.org/discussion/31/what-would-a-trusted-data-feed-look-like/p1`

[4] Massimo Morini, *Inv/Sav Wallets and the Role of Financial Intermediaries in a Digital Currency*. Located at `http://ssrn.com/abstract=2458890`

[5] Nicholas Szabo, email, September 18, 2014