Cryptocurrency with a smoothly varying price that does not need a central regulatory entity

Uma Pessoa
PGP: 12A3 89E0 4494 A4B3 5485 6B86 94F9 1ABE 3390 B95C

2018

Abstract

The main obstacle that keeps cryptocoins as a mostly speculative currency is it's volatility. I propose two solutions to this problem that do not depend on a oracle(an external, trusted, data feed) or a regulatory entity. The approaches used are two convertible coins within a single block-chain and the use of the network hash-rate to estimate the in-exchange price variation and respond by changing the coin supply.

1 Introduction

The only way of reliably controlling the price of a good is through changes in supply and demand. This can be caused by in market mechanisms (convertible securities) or an external agent (central banks). I propose one solution of each kind and explain why previous attempts failed.

Both solutions are implementable today with existing technology. They do not depend on a mathematical breakthrough.

It's important to note that my goal is not to create a currency with constant or nearly constant price. I envision a currency with a smoothly variating price so that people don't lose 20% of their savings in a single day or a few hours.

2 Previous approaches

Up to now there have been two approaches to solving the problem of high volatility.

The first is to have a central entity peg the price of the coin to an external asset through the printing and buying of new coins or have the network do it based on data from a external feed. There have been several attempts of this kind such as [3, 4, 2]. The second is through the use of convertible securities.

Attempts of both kinds have failed for different reasons.

3 Why previous attempts failed

3.1 Price peg

Understanding why approaches of the first kind failed in the crypto world while being very successful in the real world is very simple. They defeat the whole purpose of cryptocurrencies, that is having a decentralized currency. On top of that, some, have inadequately allocated efforts between marketing and engineering.

3.2 Convertible securities

In the fist years of cryptocurrencies the only way of having convertible securities was through the means of a centralized exchange. This fails for reasons similar to the ones in the previous case. That is, its completely centralized, both parties have to trust the exchange for keeping the collateral and for not going bankrupt.

This scenario began to change in 2015 with the release of Etherum. A distributed computing platform based on the 2013 paper[1]. A trust less exchange would make convertible securities a viable option, unfortunately due to engineering challenges as of today there is no successful exchange of such kind.

4 The use of convertible coins to smooth price changes

This solution would require a single block-chain with the following properties:

1. Two different coins, coin A and coin B;
2. Coins are mined in pairs, for each coin A mined there will also be a coin B;
3. Users can have any amount(greater than zero) of both coins;
4. Both coins can be individually sent to other addresses;
5. Each coin can be transformed in the other in a proportion of 1:1 at a non zero fixed cost(conversion cost);
6. The block-chain keeps track of how many of each coin there is in existence.

4.1 Arbitrage scenario

Coin A is selling at a price one conversion cost greater than that of B. A person then buys coin B and a few things happen:

- The supply of coins B shrinks, what drives the price up;
- The person sells her coins A for a profit and with that increases the supply of coins A, what drives the prices down;

If the price different persists the person can use her profit to repeat the process and make even more money.

4.2 Minimum spread between coins A and B so that profit is possible

Suppose coin A is selling at a price greater than coin B. The costs for performing arbitrage are:

- X = exchange fee for buying coin B;
- Y = transaction cost for converting B into A;
- Z = exchange fee for selling coin A.

The total cost, C, being:

$$C = X + Y + Z$$

In the case the person already posses a coin B or the exchange fee is 0, $X = 0$, so it's possible that:

$$C = 0 + Y + Z$$

If the person decides not to sell her new coin A or the exchange fee is 0, $Z = 0$, so it's possible that:

$$C = X + Y + 0$$

But it's never possible that $Y = 0$, according to property 5. Therefore the minimum cost is:

$$C = 0 + Y + 0 = Y$$

If the spread is smaller than $C = Y$ the arbitrage operation would result in loss. Given that the price difference between the coins A and B is smaller than the minimum spread the price fluctuations will be governed by speculative forces.

4.3 Spread band

Suppose coin B is the one selling for the lowest price, P_B, in the market. Let C be the minimum spread and let the term "spread band" denote a price interval from $P_B - C$ to $P_B + C$.

4.4 Is it possible for the price of both coins to move together?

4.4.1 Scenario

The market is so efficient at keeping both price within the spread band that they effectively move as one.

4.4.2 Proposed solution

The coins cannot effectively become one. If it happens there will be no incentive for arbitrage, which is the very force making the coins effectively behave like one. Therefore this scenario cannot sustain itself for more than a few instants.

4.5 Is it possible for one of the coins to “disappear”?

4.5.1 Scenario

If a price movement is too strong, the arbitrageurs might end up converting almost all or all coins of one kind, say B.

4.5.2 Proposed solution

This scenario is not a problem. Once one of the coins’ supply becomes significantly smaller than the other it will also begin to respond differently to variations in supply and demand. In another words, the volatility of one of the coins, say B, will increase in relation to the other’s, say A. This increase in volatility also increases the opportunities for arbitrage from coin A to coin B and from coin B to coin A. This increase in arbitrage will once again smooth the price variation.

5 Variable supply of new coins based on network hash-rate

This solution consists on using the network hash-rate as an in-chain indicator of the coin’s price changes and variating coin supply through destroyed transaction fees and block rewards.

$$H_c = C_r + C_h$$

$$E = M_p + H_c$$

Suppose miners reinvest all their profits, M_p, in mining power.

The exchange ratio E can variate due to changes on the market premium per coin M_p - determined solely by supply and demand - and due to changes on hardware costs H_c.

Hold H_c constant. If M_p increases, the market premium per coin, E, also increases. This represents extra profit for the miner, which reinvests it in mining hardware, the hash-rate increases. If the reward per block is increased, the coin supply increases and E drops back.

Hold H_c constant. If M_p decreases, the market premium per coin, E, also decreases. This represents a decrease in profit for the miner, some machines have to be turned off, the hash-rate decreases. If the reward per block is decreased and the destroyed fees are increased, the coin supply shrinks and E raises back.

If H_c increases, the miner will turn some less efficient machines off, reducing coin supply, increasing E and M_p. This increase in M_p raises the miner's profit, that can then turn the machines back on. The coin supply increases, E and M_p drop. The net effect will be a decrease in hash-rate, followed by an increase, ending with more or less the same E. In this case is not desirable to artificially variate the hash-rate.

If H_c decreases, the miner will turn some less efficient machines on, increasing coin supply, decreasing E and M_p. This decrease in M_p reduces the miner's profit, that then has to turn the machines back off. The coin supply decreases, E and M_p raise. The net effect will be a increase in hash-rate, followed by a decrease, ending in more or less the same E. In this case it's not desirable to artificially variate the hash-rate.

If the hash-rate change was due to variations in H_c, it will also be brief, since the market will self regulate. Therefore any implementation of this system should not react immediately to hash-rate variations.

5.1 Implementation

An implementation of this system would need an activation threshold based on time and hash-rate variation.

The time factor is important to cover the scenarios where the market will self regulate. The network should not try to interfere with oscillations that do not persist for longer than D days.

The activation threshold is important because there will inevitably be small variations on hash-rate. Those will be due miners entering and leaving business, power outrages around the world and other externalities.

A volatility goal is what effectively makes the system work. It should be something on the lines of: The hash-rate should not sustain an oscillation greater than 5% for longer than D days based on a W days hash-rate moving average.

It's necessary to take into account the effects of Moore's law. The hash-rate will naturally increase exponentially. Therefore if the upward and downward volatility goals are to be effectively the same, the former should be the equal to the later plus the exponential growth of the hash-rate over the same time-frame.

5.2 Protocol specificities

A protocol that supports this solution will need a few characteristics:

1. Infinite coin supply that does not decrease with time;

2. Transaction fees are destroyed;

3. The hash-rate history is recorded on the block-chain.

The hash-rate for the last M minutes would be inserted in the block by the miner who discovers it. It can be easily calculated so other peers only accept the block if it's within a certain small boundary of their own results.

5.3 Interesting consequences

Since transaction fees are destroyed there won't be a struggle between miners and other users ultimately leading to 10 dollars fees. If necessary the transaction fees can be regulated using the hash-rate as a parameter in an analogous way to the block reward.

This system also allows for a variable block size. On conventional cryptocurrencies the block size can't be made large enough to accommodate all or almost all transactions, since an "unlimited supply of transactions" would drive the fee to zero and bring spam/DOS problems. The result will be acceptable transaction latencies.

6 Final considerations

I have no intention of creating a new coin or work on a existing one to implement those solutions. I leave that task to others. Nevertheless I do want my ideas to succeed, so I warn people to be cautious of ICOs and pre-mined coins.

7 Conclusion

I proposed two solutions for the problem of high volatility, one based on convertible coins and arbitrage, the other on using hash-rate changes to estimate in-exchange price variations and adjust coin supply accordingly. The later approach effectively creates a distributed and trust less central bank and has the advantage of allowing for low transaction fees and variable block size.

References

[1] https://github.com/ethereum/wiki/wiki/White-Paper

[2] https://www.saga.org/static/files/saga-whitepaper.pdf

[3] https://www.basis.io/basis_whitepaper_en.pdf

[4] https://makerdao.com/whitepaper/Dai-Whitepaper-Dec17-en.pdf