BitOMC: An Adaptive Pricing Mechanism for Bitcoin-Settled Contracts

Joshua Doman
joshsdoman@gmail.com
bitomc.org

Abstract. In a world with ever-changing economic conditions, Bitcoin's fixed supply makes it unattractive as a unit of account in contracts where payment is due in the future. BitOMC addresses this by defining an adaptive unit of account that can facilitate price stability in a Bitcoin economy, without changing Bitcoin's core protocol. Using two interconvertible assets, Tighten and Ease, implemented as a metaprotocol on Bitcoin, BitOMC establishes a dynamic interest rate and a new unit of Bitcoin called the "util". This system allows for more predictable Bitcoin-settled payments, potentially accelerating Bitcoin's adoption in commerce while preserving its fundamental properties as a store of value.

1. Introduction

Economic prosperity depends on sound monetary policy and a good standard of deferred payment. In transactions where payment is due in the future (wages, subscriptions, purchases made on credit, etc.), it is desirable to use a unit of account whose purchasing power is relatively stable. In the US, this unit is the US dollar, and the responsibility for its stability is given to the Federal Reserve, which manages its value by controlling short-term interest rates.

Unfortunately, central banks often get it wrong, limited by poor data and the human judgment of a handful of experts. This can have disastrous economic consequences. By distorting the true cost of capital, central banks can induce asset pricing bubbles, like the "Everything Bubble" of 2021. This leads to distorted economic activity, and when the bubble inevitably bursts, investors and consumers get hurt.

What is needed is a monetary system where the price of money is set by the market, and not a central bank. With a fixed supply and global liquidity, Bitcoin is the ideal candidate for the settlement layer of this monetary system.[1] Bitcoin's fixed supply, however, makes BTC a poor standard of deferred payment, because its future purchasing power will always be uncertain.

In this paper, we propose BitOMC, a metaprotocol that facilitates the use of an adaptive unit of account for Bitcoin-settled payments. Forked from Casey Rodarmor's Runes Protocol,[2] BitOMC uses two interconvertible assets, Tighten and Ease, to set a dynamic interest rate, which in turn defines a floating unit of Bitcoin called the "util." Users can convert between Tighten and Ease according to a constant function conversion rule, allowing users to influence the interest rate by converting or buying and selling Tighten and Ease. Transactions continue to settle in Bitcoin, but parties seeking greater price stability may choose to denominate their contracts in "utils" rather than BTC or a centrally-controlled fiat currency.

2. Economic Theory

This paper is solely concerned with a theoretical future in which Bitcoin is the world's risk-off asset and its aggregate value relative to global wealth is well-established. Naturally, demand for risk-off assets rises and falls with changing economic conditions. When there is more demand for investment, risk-off assets are sold, and when there is less, risk-off assets are purchased. A future economy where Bitcoin is the risk-off asset will be no different. The interplay between demand for Bitcoin versus investment in this economy will be governed by the real rate of interest, or the real rate of return on Bitcoin. A neutral authority that provides a reliable estimate of this value can define a floating unit of BTC whose purchasing power is relatively stable. This is the purpose of BitOMC.

A key idea presented in this paper is that Bitcoin can be thought of as a perpetual bond whose interest payments are deferred forever into the future. The following sections cover the valuation of such perpetual bonds and the relationship between Bitcoin's discount rate and the relative level of Bitcoin demand.

3. A Primer on the Valuation of Perpetual Bonds

Let's consider a perpetual bond that forever returns a constant amount of value in real terms. Let's define an imaginary currency unit, called the "util," which has precisely the same value in real terms at all points in time. Now, consider a hypothetical risk-free bond, which returns 1 util per year, in perpetuity. How many utils is this bond worth today?

To calculate the present value of each coupon payment, we must discount it by the appropriate discount rate r. This is determined by the rate of return we could receive elsewhere in the market investing at an equivalent level of risk. To obtain the present value of 1 util received t periods in the future, we discount by $1/(1+r)^t$. The present value of our perpetual bond is therefore:

$$\begin{aligned} PV_{bond} &= \frac{1}{1+r} + \frac{1}{(1+r)^2} + \frac{1}{(1+r)^3} + ... \\ &= \frac{1}{r} \end{aligned} \tag{1}$$

4. Valuing a Deferred Perpetual Bond

Now, let's suppose our perpetual bond defers payment for the first period. This means that instead of receiving one util, we receive new bonds of equal value, in this case $1/r$ bonds. Does this change the value of what we hold today? Logically, no. We expect to receive the same amount of value each period, so the value of what we hold today does not change.

Let's now suppose that we defer payment until the Nth period. Each period, our bond holdings accrue "interest" at the discount rate r, and we expect to receive the same amount of value each period. Thus, the present value of what we hold today remains $1/r$.

This remains the case at the limit where N approaches infinity. In the next section, we'll use this to describe a model for the relationship between r and the value of bitcoin.

5. A Valuation Model for Bitcoin

Let's suppose bitcoin's required real rate of return is r. In equilibrium, this is the same as bitcoin's expected real rate of return.

Let's define an intermediate denomination of bitcoin, which we'll call the "e-bond." Initially, there is one e-bond per bitcoin, but every second, the number of e-bonds per bitcoin q grows at bitcoin's required real rate of return $r(t)$:

$$\frac{dq}{dt} = r(t) \tag{2}$$

As a purely illustrative example, suppose 1 bitcoin at $t = 0$ equates to 100 e-bonds and $r = 1\%$ per second. In one second, holding 1 bitcoin equates to holding 101 e-bonds. If r increases, the number of e-bonds per bitcoin grows at a faster rate, and vice versa. This is purely accounting. Holding e-bonds is no different than holding bitcoin. It is simply a different denomination.

The concept of e-bonds is important because it provides a bridge to our perpetual bond. E-bonds forever accrue interest at the discount rate r, just like the deferred perpetual bond where the number of deferral periods N approaches infinity. Thus, the present value of 1 e-bond is inversely proportional to bitcoin's discount rate:

$$PV_{e-bond} \propto 1/r \tag{3}$$

Combining (2) and (3), we obtain a deterministic algorithm for the number of e-bonds per bitcoin and the number of utils per e-bond as a function of r and t. We therefore have a model for how the number of utils per bitcoin changes over time, in the steady-state equilibrium.

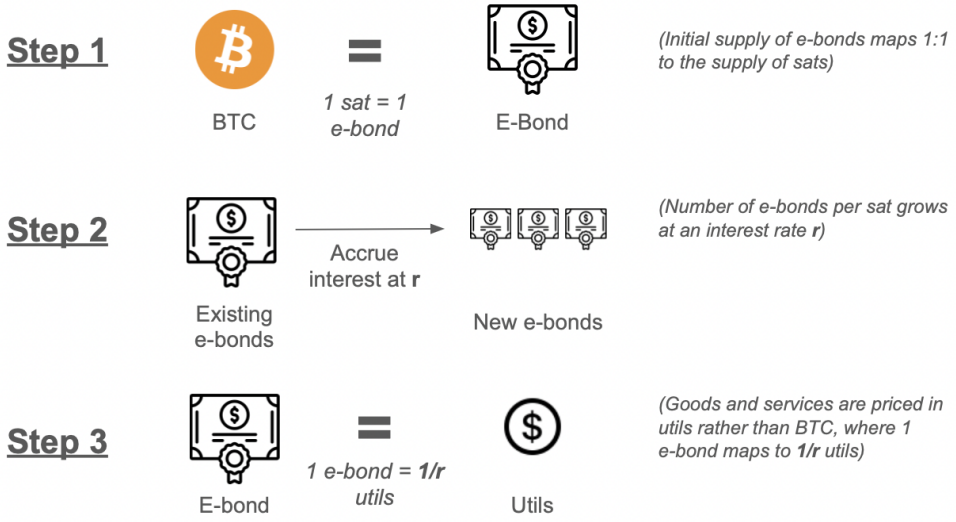


Fig. 1. The relationship between r and the number of "utils" per BTC

BitOMC uses this model and an estimate of r to define the util. We first define 1 e-bond as 1 sat, the smallest denomination of bitcoin. We then compound the number of e-bonds per sat using the BitOMC estimate of r. Finally, we define the util such that 1 e-bond is $1/r$ utils. If the estimate of r is accurate, changes in the quantity of utils will mirror changes in demand for bitcoin, creating price stability in a util-denominated economy.

6. Estimating r

The choice of governance over the estimate of r is a critical one. A robust governance mechanism should be (a) decentralized, (b) resistant to delegation, and (c) as smooth and continuous as possible. Simple majority voting can be done in a decentralized fashion, but it is prone to delegation and impractical to do continuously. BitOMC instead sets r continuously through the observed conversion rate of two convertible assets, Tighten and Ease.

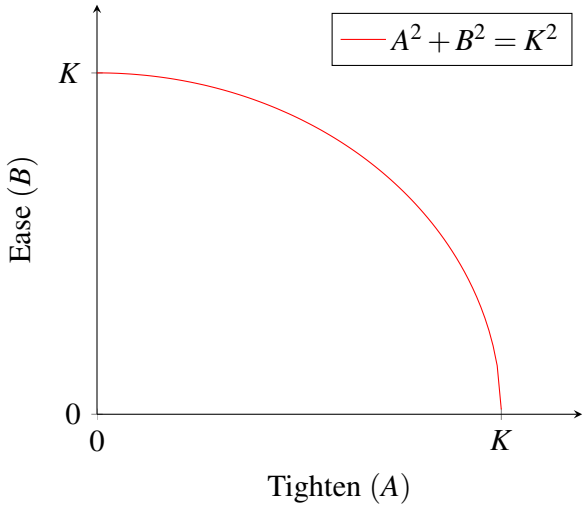
To understand how this mechanism works, let A be the outstanding quantity of Tighten, and let B the outstanding quantity of Ease. We define r in terms of A and B, such that $r = 0$ when $A = B$:

$$r = \frac{A - B}{A + B} \tag{4}$$

Users can freely convert between the two assets, according to the invariant:

$$A^2 + B^2 = K^2 \tag{5}$$

where K is some constant. This is akin to the constant product rule used by Uniswap and ensures that the conversion rate moves against the trader as they convert.[3] Visually, conversions occur occur along a quarter-circle with radius K:



As we can see, the conversion rate worsens as Tighten is converted to Ease, and vice versa. The marginal conversion rate from Tighten to Ease at (A, B) is given by the negative slope of the curve, which is A/B:

$$\begin{aligned} -\frac{dB}{dA} &= -\frac{d}{dA}\left(\sqrt{K^2 - A^2}\right) && \text{(since } B = \sqrt{K^2 - A^2}\text{)} \\ &= \frac{A}{\sqrt{K^2 - A^2}} \\ &= \frac{A}{B} \end{aligned}$$

By symmetry, the rate at which Ease can be converted to Tighten is B/A. Assuming no arbitrage or transaction costs, these conversion rates reflect the relative market price of Tighten and Ease. To illustrate, suppose that A/B is not the price of Tighten relative to Ease. Without loss

of generality, let's assume the relative price is above A/B. Arbitrageurs can buy Ease, convert to Tighten at B/A, and sell in the market for a profit. Doing so will increase A and decrease B, causing A/B to rise. This will continue until A/B equals the relative market price.

For this reason, it is the relative price of Tighten and Ease that determines the value of r. Converting itself will not change r for very long, since arbitrageurs will reverse the conversion. For an attacker to artificially raise r, they must first acquire the entire supply of Tighten, which will be a majority of the value of the system. Artificially lowering r is similarly expensive, requiring the attacker to purchase the entire supply of Ease.

Let's calculate the percentage of the value of the system that Ease represents at a given value of r. Let $x = A/B$, the relative price of Tighten to Ease. The percentage p of the value of the system that Ease represents is:

$$\begin{aligned} p &= \frac{B}{B + x \cdot A} \\ &= \frac{1}{1 + x^2} \end{aligned} \tag{6}$$

Using (4), we have $r = (x-1)/(1+x)$, or $x = (1+r)/(1-r)$. Substituting into (6) and simplifying, we obtain:

$$p = \frac{1}{2} - \frac{r}{1 + r^2} \tag{7}$$

Thus, if $r = 5\%$, an attacker must acquire 45% of the value of the system before they can lower the interest rate. If $r = 20\%$, they must acquire 31%, and so forth. In short, the system is secure as long as $1 - p(r)$ of the value of the system is held by "honest" actors, where r is the required real rate of return on bitcoin.

7. The Full Protocol

We need to make a few adjustments to use this protocol in practice. First, we cannot allow r to be less than zero, nor can we allow r to be easily manipulated. We therefore define r as the median value of $\frac{A-B}{A+B}$ over the last 100 blocks where $A > B$.

Second, we issue the supply of Tighten and Ease steadily over time, so that the initial distribution is fair. Each block, Tighten and Ease can be minted, such that a reward R is split between $A \cdot R/K$ Tighten and $B \cdot R/K$ Ease, where A and B are the respective supplies at the end of the previous block. The value of R is initially $R = 50$, and this halves every 210,000 blocks, following the same issuance schedule as Bitcoin. This split prevents r from changing due to issuance and ensures that K, the maximum quantity of Tighten and Ease, never exceeds 21 million.

The protocol is implemented as a fork of Casey Rodarmor's Runes protocol, a lightweight metaprotocol for asset issuance on Bitcoin. Assets are limited to Tighten and Ease, and the Runestone, the protocol-specific data in the `OP_RETURN` output, is encoded slightly more efficiently, given that certain features of the Runes protocol are not needed.

A conversion is defined as a transaction where one asset is transferred in excess of its balance and one asset is burned, according to the same rules as the Runes protocol. The asset ID of the final "edict" determines which asset is to receive the remainder post-conversion. If the ID is the output asset, for example, the input amount is exact and the excess output is the minimum

allowable output post-conversion. If the minimum cannot be satisfied, the input is unburned and the conversion is cancelled.

To prevent network congestion from many mints fighting to appear first in a block, each mint transaction is required to signal replaceability, leave an output that anyone can spend starting in the next block (as a P2WSH for `OP_1 OP_CHECKSEQUENCEVERIFY`), and spend the output of the previous mint, if unspent. In addition, the minter receives any outstanding burned balances. This may allow Tighten and Ease holders to tip miners directly in the future. Lastly, if a block lacks a mint transaction to receive the newly issued supply, that supply is burned and rolled over to the next block.

To prevent MEV (miner extractable value) from miners having control over the order of conversions, each conversion transaction is required to signal replaceability, leave an output that anyone can immediately spend (as a P2WPKH for a known private key), and spend the output of the conversion immediately preceding it, if unspent. This makes chained conversions deterministic, in the sense that the converter will know the outstanding supply at the time their conversion is executed. In other words, there is no slippage on chained conversions, and miners cannot sandwich transactions. The choice of P2WPKH for the output is to make it unprofitable to spend the output at normal fee rates. In the event that the output of the last conversion has been spent prior to the current block, the requirement to spend the output of the preceding conversion is lifted, but only for one block. This temporarily reintroduces the potential for MEV but prevents the mass invalidation of conversions. The output of the first conversion in this block provides the anchor to the next conversion in subsequent blocks.

There are two drawbacks worth noting with this design. First, converters must monitor the mempool for replacements and choose a fee unlikely to be replaced. Second, present mempool policies limit the number of chain conversions that can exist in the mempool. The former is a relatively small cost and the latter can be resolved in the future if needed, without requiring changes to the protocol. In the future, if Bitcoin introduces an opcode that allows for transaction introspection, the protocol can be improved so that a new anchor must be created in order to spend a previous one.

8. Conclusion

For Bitcoin to be competitive as money, it needs a unit of account that reflects a consistent amount of value in real terms. This requires an interest rate and a protocol to set monetary policy. While US monetary policy is set by a closed committee, BitOMC sets monetary policy through the open market. Anyone can influence monetary policy by buying or selling Tighten and Ease, democratizing a process that is opaque and somewhat arbitrary today.

As stated earlier, BitOMC is most effective in a future where Bitcoin is the world's risk-off asset. Until then, while the "util" may prove useful as a way to denominate some cross-border transactions, it will take time before it can widely substitute existing currencies like USD. What's important is that BitOMC provides a rebuttal to those who say that Bitcoin can never be widely adopted as a currency due to its fixed supply. A monetary layer like BitOMC could change this narrative and accelerate the adoption of Bitcoin as a global monetary standard.

Acknowledgments

Thank you to Yuga Cohler, Alana Levin, Patrick Lung, Akshay Malhotra, and others for helpful discussions during the development of this paper.

Notes and References

[1] Nakamoto, S. "Bitcoin: A Peer-to-Peer Electronic Cash System." (2008) (accessed 17 June 2023) `https://bitcoin.org/bitcoin.pdf`.

[2] `https://rodarmor.com/blog/runes/`.

[3] Angeris, G., Kao, H.-T., Chiang, R., Noyes, C., Chitra, T. "An Analysis of Uniswap markets." *Cryptoeconomic Systems* **0.1** `https://doi.org/10.21428/58320208.c9738e64`.