Cowri White Paper

November 27, 2019

Abstract

The Cowri protocol is a network of stablecoin baskets called, “shells”. Shells can be customized to contain any basket, and one shell can natively convert to any other shell. Collectively, the shell network functions as a decentralized exchange that pools liquidity for stablecoin-to-stablecoin trades. Shell token holders thus earn passive income from liquidity fees. Given that shells are diversified, inherently liquid and earn passive income, they have the potential to be the best store of value of any crypto asset. In legacy financial markets, this role is played by US Treasury bills. The goal of Cowri is to connect our species - in a way only commerce can - by creating an internet monetary system accessible to all.

1 The money myth

Mastering fire was perhaps our species’ foundational accomplishment. Prometheus, the mythic individual who brought flame down from Mount Olympus, was the original technologist. An equally important milestone was the invention of money. Money has been with us since the dawn of humanity; it is part of our identity as human beings. In this case, “Prometheus” was probably a tribe of skin divers living along the coast of East Africa many thousands of years ago. And rather than bear the torch of knowledge from on high, they carried from the depths of the ocean our species’ first currency: cowry shells.

Ancient societies from Mesopotamia, India and China all used cowry shells.[29] Cowries were a truly global currency. We can see the lasting legacy of these shells in the modern Chinese character, 貝 (“bei”), which has two meanings: “shell” or “money”.[31] This character originally started out as a pictogram carved into bone fragments and gradually evolved into its current form (see Figure 1). Cowry shells even influenced early Chinese coinage.[3][13]

The transition of cowries as a physical representation of value (shells) to increasingly abstract representations of value (pictograms) is a common pattern in the evolution of money. Before the advent of paper currency, gold specie was the primary form of money. In the 7th Century, merchants and bureaucrats in Tang China figured out how to imbue paper with the monetary properties of gold by storing gold in a secure vault and issuing paper notes redeemable for the gold held in reserve.[8][13]

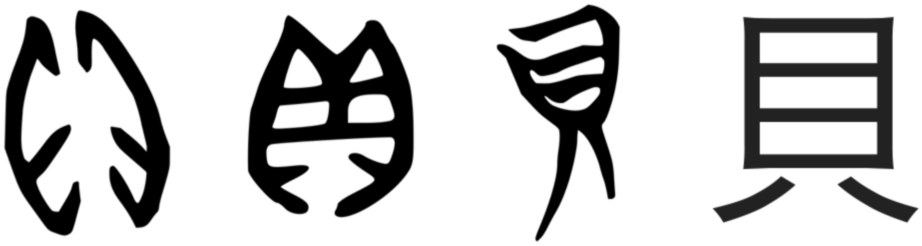


Figure 1: Over time, cowry shells changed from a pictogram on Oracle Bones (far left) to today's modern Chinese character for "money" and "shell" (far right).

The idea of paper currency was brought back to Europe by traveling merchants such as Marco Polo. Once brought to Europe, paper currency enabled the invention of stocks and bonds, pieces of paper with claims on future income flows. Complex financial contracts are only possible if accountants can manipulate ink on paper in lieu of carrying heavy pieces of metal. The financial revolution unlocked by paper currency literally underwrote the Industrial Revolution.[14] Globally, stocks and bonds were worth $177.5 trillion dollars in 2018.[21]

2 The past and future of money

In 2008, a new era of money dawned when Satoshi Nakamoto invented Bitcoin.[19] Although Bitcoin pioneered the first cryptocurrency, its self-imposed fixed supply ensured that its price would be too volatile to be used as money. Enter stablecoins, cryptocurrency designed to have a stable value over time. Stablecoins relate to fiat the same way paper currency related to gold. Although more efficient, stablecoins nonetheless had to peg themselves to fiat currency, usually the US dollar, because fiat is (for now) the standard unit of account. Paper currency tied itself to gold for the same reason.

To understand the profound advantage of stablecoins over fiat, we first need to understand the legacy system it will replace. Currently, money exists as an entry in an ad hoc confederation of private ledgers our society refers to as "the banking system". The banking system is replete with unnecessary redundancy and a commensurate level of inefficiency.

Just as paper was orders of magnitude more efficient than heavy pieces of metal, a public blockchain is dramatically superior to coordinating a cumbersome network of bank ledgers. This goes well beyond faster settlement time for payments. Money is now programmable. Financial contracts can become arbitrarily complex, limited only by the performance of micro-processors. Moreover, cryptocurrency exists natively on the internet, unconstrained by geography.

For a taste of what is to come, consider the emerging decentralized finance (DeFi) ecosystem. Decentralized lending protocols, such as Compound[17], allow lenders to pool stablecoins and lend them out to borrowers. At the time of writing, Compound holds $93.8 million in total value.[7] In return, lenders are paid interest. These protocols are live 24/7 and are available to anyone with internet access. There are no middlemen and the lending protocols operate autonomously with no need for human oversight.

2.1 The stablecoin conundrum

As DeFi projects already demonstrate, stablecoin-based financial systems are already capable of design patterns Wall Street and the legacy system could never replicate. The gap between the two will only accelerate in the future. In spite of stablecoins' potential, they are mired in a conundrum: on the one hand, an ecosystem with many competing stablecoins will fragment the market; on the other hand, an ecosystem with only one stablecoin will present untenable economic risk.

If there are many different stablecoins in circulation, then people will have to constantly manage individual currencies. Imagine trying to buy a cup of coffee but first having to negotiate with the barista which stablecoin both of you are willing to accept? Running a business will be maddeningly complicated, to say nothing about trading on financial markets. A fragmented stablecoin ecosystem would be dysfunctional.

It may seem as if the solution is to settle on a single stablecoin. However, that would come with a hidden cost that is perhaps even greater. Stablecoins may be superior to banks from a technological standpoint, but from an economic standpoint, they suffer from the same drawback: occasionally, they fail. In a market with many competing stablecoins, one failure will not pose an existential risk. However, with only a handful of stablecoins, or worse, a single stablecoin dominating, a failure would catastrophic. We cannot rely on stablecoins that are too-big-to-fail.

2.2 The origins of central banking

So what do we do? How do we diversify the market without simultaneously fragmenting it? The history of paper currency and the origin of central banking show us a way forward. Although banks are technologically inferior, they have had several centuries of trial and error to work out solutions to systemic economic challenges.

During the rise of paper money, each bank would issue their own notes. Hence, there were thousands of separate currencies in circulation (see Figure 2). Imagine spending currency from Hawaii while travelling through Alabama? In 1907, there were 20,000 banks in the US alone.[11] Sending money across country required numerous banks to coordinate with each other, and every middle man took a healthy cut of the transaction. Fees could be as high as 50%.[9] Tangled banking relationships exacerbated financial shocks. A minor economic shock could easily cause a number of small, local banks to collapse. Collectively, these banks may not make up a meaningful portion of the real economy. However, panics would reverberate up through the banking hierarchy, putting even major New York banks at risk.

In 1913, faced with a fragmented banking system and all its social costs, the US Congress created the Federal Reserve - not to control interest rates, inflation, unemployment, or even the aggregate money supply - to unify the banking system.[23] The mandate to pursue inflation and unemployment targets was not institutionalized until 1977.[10]

Figure 2: A $10 paper note from the First National Bank of Hawaii.

The Federal Reserve solved the fragmentation problem by creating a unified system of inter-bank deposits. Banks could voluntarily pool their deposits together at regional Federal Reserve banks and in return received Federal Reserve notes. This not only streamlined inter-bank settlement but also increased financial resiliency. The new model was so successful that by 1920 69% of all bank deposits were in the Federal Reserve system.[12]

The Fed allowed banks to seamlessly inter-operate with each other. What we need is an internet protocol that allows stablecoins to do the same. Individually, a stablecoin cannot constitute money, no more than an individual bank note. To create money, we need to unify disparate stablecoins into a single network that is accessible to all.

3 Inter-stablecoin liquidity

The Federal Reserve worked by providing liquidity between banks. A stablecoin monetary system must do the same. Inter-stablecoin liquidity is an inherently hard problem because as we add more stablecoins, the number of possible trading pairs increases combinatorially. With only 15 stablecoins, there would 105 possible trading pairs. A healthy and diversified stablecoin market could easily have thousands of trading pairs.

Centralized cryptocurrency exchanges, such as Binance[4] provide liquidity between crypto assets including stablecoins. However, these are not decentralized software protocols. Exchange protocols such as 0x[27] are decentralized but suffer from a lack of liquidity. Moreover, they require off-chain coordination to match a buyer and a seller, which complicates integration with purely on-chain protocols.

Decentralized liquidity pools have proven to be a viable means of creating on-chain liquidity between many different tokens. The most notable example of this technology is Uniswap, which allows people to deposit tokens into a contract.[1] This contract then uses these tokens to provide liquidity via an automated market maker.[5] If someone else wants to swap two tokens, they can trade against the Uniswap contract by depositing one token and withdrawing the other. The protocol's internal logic determines the price of this exchange. As of the time of writing, Uniswap has $22.5 million worth of tokens deposited and $1.9 million of daily trade volume on average.[25]

Unfortunately, Uniswap is not ideal for providing liquidity between stablecoins. Uniswap's model relies on a hub-spoke graph (see Figure 3). All trades flow through the hub token, which for Uniswap is Ether. Swapping between two stablecoins, Dai and USDC, would require not one but two trades: Dai → Ether, Ether → USDC. Relying on Ether, a volatile asset, incurs a very real cost on the platform.[20]

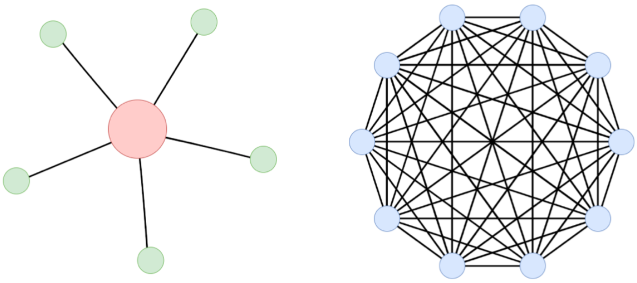


Figure 3: Network topology of a hub-spoke liquidity pool (left) versus a universal liquidity pool (right). With the same amount of capital, the hub-spoke network can support only 15 trading pairs while the universal network has 45.

But even without Ether, the hub-spoke topology is inefficient. It dilutes liquidity by forcing half of the capital into the hub asset. A better approach is to use a universal graph[28], where each token trades against all other tokens in the network (see Figure 3). Removing the hub token frees up half the capital on the exchange. A universal graph topology can therefore support exponentially more trading pairs at half the cost (see Figure 3) versus hub-spoke.

However, the universal graph has a drawback: all depositors into the liquidity pool necessarily gain exposure to all of the tokens. The advantage of the hub-spoke is that depositors can choose to allocate capital to specific spoke tokens. On the one hand, forcing all depositors to agree on the tokens in the pool is an expensive social choice problem. On the other hand, if we allow depositors to customize their exposure, then we will have multiple pools, one for each token profile, which would fragment liquidity.

Cowri's solution is to think of the liquidity pool not as a single network of tokens, but as a network of networks. When two networks have overlapping trading pairs, their liquidity is combined, which prevents fragmentation. As we will demonstrate in Section 4.2.2, this paradigm will also allow depositors to transfer capital between networks without having to remove tokens from the exchange. The rest of this paper will have three parts: protocol formalism, how the protocol will work in practice, and potential use cases.

4 The Cowri shell protocol

The key conceptual building block to the Cowri protocol is a "shell". A shell is an object that can store value in the form of different tokens. An agent, Alice, can deposit tokens into the shell and withdraw them later. Formally, a shell is comprised of a set of tokens. The balances of the tokens in the shell are represented as $\mathbf{x}$. When Alice makes a deposit, the balance of the shell increases by $\mathbf{x_d}$. A withdrawal reduces the shell's balances by $\mathbf{x_w}$

$$shell = \{token_1,\ token_2,\ ...,\ token_n\}$$

$$\mathbf{x} = (x_1,\ x_2,\ ..., x_n)$$

$$\mathbf{x} + \mathbf{x_d} = (x_1 + x_{1d},\ x_2 + x_{2d},\ ...,\ x_n + x_{nd})$$

$$\mathbf{x} - \mathbf{x_w} = (x_1 - x_{1w},\ x_2 - x_{2w},\ ...,\ x_n - x_{nw}), \quad x_{iw} < x_i$$

Abstractly, a shell needs a way to measure value, and a way to track how much value an agent has contributed. Thus, a shell has an implicit utility function, $U(\mathbf{x})$, that assigns a scalar value to the basket of tokens held by the shell. When Alice deposits or withdraws tokens, the shell's new utility will be:

deposit: $U(\mathbf{x} + \mathbf{x_d})$

withdrawal: $U(\mathbf{x} - \mathbf{x_w})$

A shell needs to track the value deposited and withdrawn by Alice. To store this information, we can define a vector, $\mathbf{v}$, where v_i is the net value contributed by agent i.

$$\mathbf{v} = (v_1,\ v_2,\ ...)$$

$$v_i \geq 0$$

$$|\mathbf{v}| = \sum v_i, \ \text{ i.e. the total value held in the shell}$$

When an agent, i, deposits $\mathbf{x_d}$ into a shell, the value credited to the agent is equal to the percent change in the shell's utility function multiplied by the total value currently stored in the shell. The percent change in the total value is equal to the percent change in the shell's utility. The opposite holds when an agent withdraws $\mathbf{x_w}$.

$$\triangle v_i = \frac{U(\mathbf{x} + \mathbf{x_d}) - U(\mathbf{x})}{U(\mathbf{x})} * |\mathbf{v}|$$

$$\triangle v_i = \frac{U(\mathbf{x} - \mathbf{x_w}) - U(\mathbf{x})}{U(\mathbf{x})} * |\mathbf{v}|$$

Using these deposit/withdrawal mechanics, shells can not only store value, but also convert or swap between different forms of value. If Alice wants to convert an amount of $token_i$ into an amount of $token_j$, she can do so by making the following deposit and subsequent withdrawal:

$$\mathbf{x_d} = (0,\ 0,\ ...,\ x_i,\ 0,\ ...,\ 0)$$

$$\mathbf{x_w} = (0,\ 0,\ ...,\ 0,\ x_j,\ ...,\ 0)$$

In effect, Alice just swapped x_i for x_j. This can be done in such a way that after the withdrawal, the net change in value attributed to Alice is zero, $\triangle v = 0$. The exact amounts of each token will depend on $\mathbf{x}$, $\mathbf{v}$, and U. The shell's net change in utility must also be zero, $\triangle U = 0$. In terms of classical economics, we can think of this conversion of x_i to x_j as a shift along an indifference curve.[15] I.e., the shell is indifferent between its token portfolio before the swap, $\mathbf{x}$, and its token portfolio after the swap, $(\mathbf{x} + \mathbf{x_d} - \mathbf{x_w})$.

4.1 Shell Implementation

Our implementation uses an ERC-20 token to record $\mathbf{v}$. We call these tokens, "shell tokens" because they are issued directly by the shell contract. When Alice makes a deposit, the contract mints new shell tokens and adds them to Alice's balance. When she makes a withdrawal, the contract burns Alice's shell tokens.

This implementation allows agents to treat their deposit as a fungible token. Alice's and Bob's contribution to the shell are directly comparable. Alice can also send Bob her deposit simply by sending shell tokens.

As for implementing the utility function, $U(\mathbf{x})$, there are multiple possibilities, and the main design constraint was the gas cost of running the smart contract code. Blockchain computations are relatively expensive, so we want to avoid a utility function that requires complex computations for deposits and swaps.

Based on these criteria, we chose to use the following specification in the current version of the protocol:

$$U(\mathbf{x}) = \prod_{i=1}^{n} x_i^{\frac{1}{n}}$$

This function is a specific implementation of what has been referred to as a "constant mean market".[2] In a constant mean market, the weighted geometric average is held constant after every swap. It is possible to specify different weights for each token in the shell, an example of which can be found in the Balancer white paper.[18] In the case of Cowri's utility function, all tokens have equal weights, i.e. all tokens in the shell are valued equally. This feature reduces computational complexity and thus the cost of running the contract code. Allowing specified weights would also fragment liquidity because shells with identical token sets but differing weights for each token would have to be treated separately. Although this fragmentation can be overcome by pooling liquidity via an aggregate swap (see Section 4.2.1), this operation would increase execution costs.

To better understand the advantages of using the geometric mean as the utility function, consider the protocol's swap implementation. Because the utility function weighs all tokens equally, we do not need to compute any exponents or roots when calculating the swap price. As stated previously, the shell's utility before the swap, U_0, and the shell's utility after the swap, U_1, are the same. Hence:

$$U_0 = U_1$$

$$U_0 = \prod x_i^{\frac{1}{n}} = (x_1 * x_2 * ... * x_i * x_j * ... * x_n)^{\frac{1}{n}}$$

$$U_1 = \prod (x_i + x_{id} - x_{iw})^{\frac{1}{n}} = (x_1 * x_2 * ... * (x_i + x_{id}) * (x_j - x_{jw}) * ... * x_n)^{\frac{1}{n}}$$

$$(x_1 * x_2 * ... * x_i * x_j * ... * x_n)^{\frac{1}{n}} = (x_1 * x_2 * ... * (x_i + x_{id}) * (x_j - x_{jw}) * ... * x_n)^{\frac{1}{n}}$$

$$(x_1 * x_2 * ... * x_i * x_j * ... * x_n) = (x_1 * x_2 * ... * (x_i + x_{id}) * (x_j - x_{jw}) * ... * x_n)$$

$$x_i * x_j = (x_i + x_{id}) * (x_j - x_{jw})$$

The product of the balances of $token_i$ and $token_j$ are the same before and after the swap. I.e., using the geometric mean as a utility function yields a swap price formula that is identical to a constant product market maker[5], which is relatively well understood and used by other protocols such as Uniswap. We can calculate the amount of x_{jw} using the following formula:

$$x_{jw} = \frac{x_j * x_{id}}{x_i + x_{id}}$$

In review, a shell can store value and convert value from one form ($token_i$) to another ($token_j$). The shell measures value according to a utility function, $U(\mathbf{x})$, and stores the value contributed by each agent in a vector, $\mathbf{v}$. In our implementation, we used the geometric mean of tokens balances in the shell as our utility function, and tracked value via an ERC-20 token.

4.2 Meta behavior of shells

Now that we have covered the mechanics of an individual shell, it is time to explain the entire protocol. The shell protocol is more than just a single shell, but a network of shells that can operate in unison. In particular, there are two collective behaviors:

1. Aggregate swaps
2. Inter-shell transfers

In an aggregate swap, multiple shells are used to swap tokens. In an inter-shell transfer, value held in one shell is converted to value held in another shell.

4.2.1 Aggregate swaps

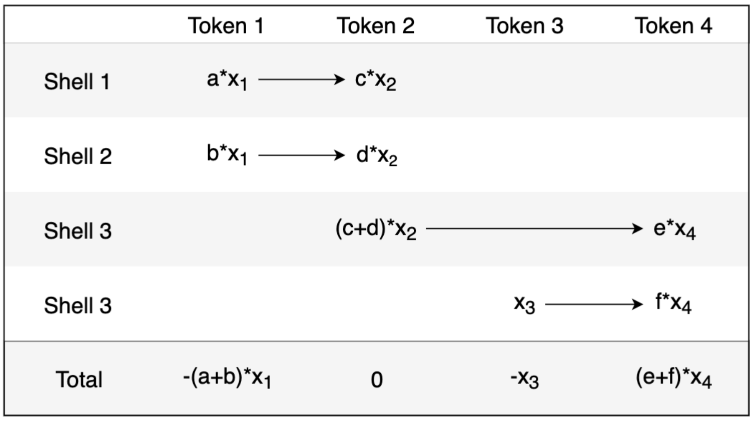
An aggregate swap is an ordered series of regular swaps chained together and executed atomically, i.e. either all of them happen or none of them happen. To illustrate how an aggregate swap works, consider the following example: Alice would like to convert her holdings of $token_1$ and $token_3$ into $token_4$. To do so, she can execute an aggregate swap using the following shells:

$$shell_1 = \{token_1,\ token_2\}$$

$$shell_2 = \{token_1,\ token_2,\ token_5\}$$

$$shell_3 = \{token_2,\ token_3,\ token_4\}$$

Figure 4 demonstrates how to compose an aggregate swap so that Alice can convert her holdings of $(a+b)*x_1$ and x_3 into $(e+f)*x_4$.



	Token 1	Token 2	Token 3	Token 4
Shell 1	$a*x_1$ →	$c*x_2$		
Shell 2	$b*x_1$ →	$d*x_2$		
Shell 3		$(c+d)*x_2$ →		$e*x_4$
Shell 3			x_3 →	$f*x_4$
Total	$-(a+b)*x_1$	0	$-x_3$	$(e+f)*x_4$

Figure 4: Three shells and four swaps are used to compose one aggregate swap. Alice converts $token_1$ and $token_3$ into $token_4$.

There are two important swap patterns to point out. First, this aggregate swap used $shell_1$ and $shell_2$ to pool liquidity when converting $token_1$ to $token_2$. We could have swapped using just one shell, but by combining them, we can reduce slippage. I.e., the swap price will be lower if you pool the liquidity of multiple shells. The ability to combine shells is important because by doing so, users can create any shell they want without fragmenting the market. Without pooling, every shell created would dilute the protocol's overall liquidity, off-setting the efficiency gains of the universal graph model versus the hub-spoke graph model (see Section 3).

The second pattern of note was using $token_2$ as a bridge to convert $token_1$ into $token_4$. None of the shells contained both $token_1$ and $token_4$. Thus, a direct swap was infeasible for Alice. The ability to use intermediate tokens as a bridge further deepens liquidity.

4.2.2 Inter-shell transfers

Alice has deposited tokens in to $shell_i$ and wants to transfer that value to $shell_j$. In order to simplify our notation, let us introduce a new deposit function:

$$d(\mathbf{x_d}|\mathbf{x},\mathbf{v}) = \triangle v$$

$$d^{-1}(\triangle v|\mathbf{x},\mathbf{v}) = \mathbf{x_w}$$

$$d_i(\cdot) = \text{deposit function for } shell_i$$

Conceptually, Alice can transfer value by first withdrawing from $shell_i$, then depositing into $shell_j$, as long as $shell_i \cap shell_j \neq 0$. The amount of $shell_j$ value tokens Alice can mint from her deposit of $\mathbf{x_d}$ into $shell_i$, and the amount she could then withdraw, $\mathbf{x_w}$, can be formally represented as:

$$\triangle v = d_j(d_i^{-1}(d_i(\mathbf{x_d})))$$

$$\mathbf{x_w} = d_j^{-1}(\triangle v)$$

Once Alice has deposited tokens into a shell, she has the ability to transfer that value into any other shell in the network. Aggregate swaps allow Alice to convert from stablecoin to stablecoin, but an inter-shell transfer allows her to convert from stablecoin basket to stablecoin basket. Additionally, Alice can enter the Cowri network via any supported stablecoin, and she can exit via any supported stablecoin. This makes Cowri an extremely fluid source of liquidity and value conversion.

5 Cowri in practice

Anyone can create a shell composed of any set of tokens. Although the protocol was designed specifically for stablecoins, that is not a hard constraint. Once created, a shell can automatically be invoked in aggregate swaps. The rest of this section explains the mechanics of how shells work in practice with the goal of explaining what a shell token actually represents.

5.1 Price discovery and arbitrage

As demonstrated in Section 4.1, the protocol determines the price of a token (i.e. its value relative to other tokens in the shell) based on the quantity of each token held in the shell, and on the amount a user wants to convert. A token that is relatively scarce will have a higher price than a token that is relatively abundant. This goes for swaps, deposits, withdrawals and inter-shell transfers.

When a user makes an exchange, they will change the quantities of the tokens in the shell, which will in turn affect the price. The token purchased by a user will become more expensive and the token sold will become less expensive. This change in price is referred to as “slippage”. Ceteris paribus, a larger transaction will have higher slippage. The more tokens in a shell (i.e. the more liquidity), the lower the slippage.

In general, prices on Cowri will track the prices on other exchanges. When there is a price differential, an arbitrage trader can exploit the difference by selling on the expensive exchange and buying on the cheap exchange. This action causes prices to converge. Arbitrage traders play an important role in the Cowri ecosystem.

Given that all tokens in a shell trade against each other, any swap between two tokens, $\{token_1,\ token_2\}$, will necessarily affect the prices for all trading pairs in the shell that contain either $token_1$ or $token_2$, e.g. pairs such as $\{token_1,\ token_3\}$ and $\{token_2,\ token_4\}$. On the one hand, arbitrage trades are now more complex. On the other hand, there are more opportunities. Aggregate swaps are a useful mechanism to exploit these opportunities because any arbitrary series of trades can be executed atomically.

Lastly, if a shell contains only stablecoins, the quantities of each stablecoin will tend to equal each other. Balanced quantities imply balanced prices, and stablecoins do not deviate in price.

5.2 Entering and exiting the network

Any time a user enters or exits the network through a deposit, a withdrawal or a swap, they will be assessed a small fee. Part of the fee will go to shell token holders (i.e. those who have capital in the platform) as an incentive. Part of the fee will go to fund the protocol and the broader ecosystem.

To make a deposit, a user only needs at least one of the shell's constituent tokens. They do not need the entire shell nor do they need to have specific ratios. Once a user has made a deposit, they can transfer their deposit to any shell that has overlapping tokens via an inter-shell transfer. This feature makes migration into new shells, for example when a new stablecoin becomes available, a seamless process. To make a withdrawal, a user can take any stablecoin in the shell. Like deposits, the user need not adhere to any specific ratios of token amounts.

Thus, a user can enter the Cowri network as long as they have at least one token supported by at least one shell. Once in the network, they can transfer value to any shell they please. And a user can then exit the network in any token supported by at least one shell. The Cowri protocol is more than just a source of liquidity for stablecoin-to-stablecoin trades. It is a liquid asset onto itself. By comparison, existing liquidity pool platforms are static whereas Cowri is dynamic. Once in the Cowri monetary system, value becomes extraordinarily fungible.

5.3 What is a Cowri shell token?

A Cowri shell token is an ERC-20 that represents a claim on the stablecoins deposited into that shell (see Section 4). That is to say, a shell token has no memory of the specific deposit made by the user, only the shell's valuation of the tokens added at the time of deposit. Once minted, a shell token can be redeemed for any of the underlying collateral. Tokens minted by one shell can also be converted into tokens minted by another shell. In that way, shell tokens are like inter-operable stablecoin baskets that are inherently liquid.

As mentioned in Section 5.2, shell token holders also earn a transaction fee whenever a user enters or exits the network. This fee grants shell tokens a source of passive income. Depositing tokens into Cowri will not only grant users access to a liquid stablecoin market, it will also earn them money. Let's consider a back of the envelope estimate: if the transaction fee is 0.2%, and the daily volume is 10% of total capital pooled on the exchange, then the annual return would be approximately 7.3%. This rate of return is notional. Actual returns could vary by a large margin. Returns for individual shells will also differ.

Given that shell tokens are diversified, inherently liquid and earn passive income, they have the potential to be the best store of value of any crypto asset. In the legacy financial system, the safest and most liquid interest bearing assets are US Treasuries. Currently, no asset fulfills that niche in crypto markets.

6 Use cases

Cowri can be thought of as a two-sided market, similar to Uber or Airbnb. On the one side, Cowri is a store of value for depositors. On the other side, Cowri is a source of stablecoin liquidity for the rest of the market. However, this is not a perfect analogy because depositors can also use Cowri as a source of liquidity via operations such as inter-shell transfers. Nonetheless, this framework can help inform our discussion of potential use cases.

6.1 Store of value

In the beginning, the only store of value available in crypto were volatile currencies such as Bitcoin. Then, stablecoins such as Tether were created.[24] This gave investors two options: high-risk, high-return; or low-risk, zero return. Crypto lending protocols such as Compound have given investors a third option. By lending stablecoins, investors can earn a positive return at a much lower risk profile.

Although these loans are over collateralized, they are not risk-free.[22] There is no guarantee that investors will be able to access their principal and may instead have to liquidate collateral. Just as banks prefer not to foreclose on a house, crypto investors prefer to not foreclose on a loan. Cowri shell tokens are different. Not only can the principal be accessed at any time, shell tokens represent a customized basket of stablecoins. Shells are more liquid and less risky than individual stablecoins, plus they earn passive income.

Any investor with idle capital who would like to minimize risk while still earning passive income can hold Cowri. These could include crypto projects and investors who keep their operating capital on-chain. Shell tokens can also function as a savings account for crypto wallets. Many decentralized exchanges pay market makers to offer liquidity on their platforms. Rather than pay for stablecoin liquidity, these exchanges can deposit into Cowri and earn income by providing liquidity for their users.

6.2 Liquidity for decentralized finance

Stablecoins are fundamental to the DeFi ecosystem, just as the banking system is fundamental to legacy financial markets. Without a robust banking system, financial markets will fail. Without a robust stablecoin ecosystem, DeFi will fail.

Currently, DeFi is dominated by one stablecoin, Dai. At the time of writing, roughly 50% of all value in DeFi is held in Dai's parent protocol, Maker.[6] Although Dai is issued by a decentralized protocol (Maker), in economic terms, Dai is nonetheless a central point of failure. The US banking system, with 8 major banks, is in some ways more decentralized than DeFi (ironically).[30] DeFi protocols should not rely on infrastructure that is too-big-to-fail. It is vital to decentralize the stablecoin market as a whole.

As discussed in Section 3, inter-stablecoin liquidity is the only way to decentralize DeFi. By offering this liquidity via a simple on-chain API, Cowri can help DeFi protocols diversify their stablecoin exposure and help mitigate existential risk. Because Cowri can execute an arbitrary combination of trades atomically, it is composeable with other smart contracts.

Additionally, Cowri enables novel financial products. For example, a stablecoin liquidity pool combined with a stablecoin lending protocol can create carry trades, where a trader borrows in a low-interest stablecoin and simultaneously lends in a high-interest stablecoin, pocketing the difference. A carry trade is like interest rate arbitrage. Such a strategy is only feasible with ample liquidity between stablecoins. Not only does Cowri help DeFi become safer, it also creates new business opportunities.

6.3 Crypto payments

Payments will inevitably be a critical part of the crypto economy (indeed, payments are the foundation of any economy). Payments rely on a stable medium of exchange, i.e. money. As discussed in Section 2.1, stablecoins on their own cannot constitute money. Market fragmentation will be prohibitively complicated for end-users. Imagine having to manage a wallet with dozens of stablecoins? This problem will be exponentially more complicated for developers, who will have to manage stablecoins for potentially millions of users.

Shell tokens are a natural solution to stablecoin complexity. Rather than manage many individual currencies, users can choose a basket of stablecoins they are willing to accept (or have developers make that decision on their behalf). If users prefer different baskets, they can still transact seamlessly with each other via an inter-shell transfer. Neither end-users nor developers need to even be aware of the individual stablecoins everyone else is using.

6.4 New stablecoin adoption

To have a healthy ecosystem, new stablecoins must be able to enter the market. Merely decentralizing stablecoin market share is not enough. No industry can thrive if incumbents dominate by default. Innovation and progress rely on a continual infusion of fresh entrants. In many industries, incumbents will squelch would-be competition by monopolizing distribution channels. Distribution will be especially important for stablecoins, a market driven by network effects.

Cowri functions as a decentralized distribution platform for new stablecoins. Adding liquidity is as simple as adding the stablecoin to a new shell. Once a shell has been created, users can migrate their capital via an inter-shell transfer.

Such a service may be valuable for stablecoins that already have some traction but would like to branch out into broader crypto markets. For example, Terra[16] has developed a stablecoin specifically for retail payments in Asia. However, Terra is not used outside of its ecosystem. With Cowri, novel stablecoins such as Terra can grow beyond their initial user base and expand adoption.

7 Conclusion

Throughout history, nothing has connected our species more than commerce; people coming together to exchange goods and services. Cowry shells and paper money transformed our society by enabling new forms of commerce. Cryptocurrency is poised to do the same. The vision for Cowri is to connect our species by creating an internet monetary system accessible to all. Unifying stablecoins into one network is the first step in a long journey.

The final transition to fiat (backed by nothing) came in 1971.[26] US President Richard Nixon, with the backing of the Federal Reserve, announced to the world that the US would no longer peg its dollar to gold. Instead, the US dollar would float freely on the open market. Whether this event was ultimately in the rest of the world's interest is a matter of debate. But it was nonetheless a significant milestone for our species: no longer did we require physical scarcity to create money, but abstract scarcity. Money as an abstract technology came into its own. And once an abstract concept, money became programmable. Cryptocurrency, a natural implementation of programmable money, is the next step.

What does the past tell us about the long-term future of cryptocurrency? For the time being, stablecoins will be a necessity. But what will happen after cryptocurrency consolidates and entirely supersedes the legacy system, just as the banking system superseded gold in 1971? We cannot know for certain. Perhaps, one day we will detach from the fiat peg and let our money float freely through the internet.

References

[1] Adams, Hayden (2019, July 5). “Uniswap white paper”. https://hackmd.io/C-DvwDSfSxuh-Gd4WKE_ig#DEX-inside-a-Whitepaper

[2] Angeris et al (2019, November). “An analysis of Uniswap markets”. https://arxiv.org/abs/1911.03380

[3] Ashkenazy, Gary. “Bei - Cowrie shells used as coins and various imitations”. Retrieved 2019, July 21: http://chinesecoins.lyq.dk/Bei/BeiNetpakke/index.htm

[4] Binance (March, 2018). “Binance Exchange White Paper”. https://whitepaperdatabase.com/binance-coin-bnb-whitepaper/

[5] Buterin, Vitalik (2018, March). “Improving front running resistance of x*y=k market makers”. *ETH Research.* Retrieved 2019, November 24: https://ethresear.ch/t/improving-front-running-resistance-of-x-y-k-market-makers/1281

[6] DeFi Pulse (2019, November 25). “DeFi Pulse: Total Value Locked (USD) in DeFi”. Retrieved 2019, November 25: https://defipulse.com/

[7] DeFi Pulse (2019, November 26). “DeFi Pulse: Compound”. Retrieved 2019, November 26. https://defipulse.com/compound

[8] Extra Credits (2016, Oct 8). “The History of Paper Money - Extra History - #2.” YouTube. Retrieved 2019, July 21: www.youtube.com/watch?v=rPHTmGjoe2k.

[9] Extra Credits (2016, Oct 29). “The History of Paper Money - Extra History - #5.” YouTube. Retrieved 2016, 29 Oct: www.youtube.com/watch?v=LrB9bS2VOLE.

[10] The Federal Reserve Bank of Chicago (2019, July 12). “The Federal Reserve’s dual mandate”. Retrieved 2019, July 21: https://www.chicagofed.org/research/dual-mandate/dual-mandate

[11] Friedman, M., & Schwartz, A. J. (1963). *A monetary history of the United States, 1867-1960.* Chapter 4: Gold Inflation and Banking Reform (page 169). Princeton University Press.

[12] Friedman, M., & Schwartz, A. J. (1963). *A monetary history of the United States, 1867-1960.* Chapter 5: Early Years of the Federal Reserve System (page 190). Princeton University Press.

[13] Goetzmann, W. N. (2016). *Money changes everything: How finance made civilization possible.* Part 2, Chapter 8: China’s Financial World. Princeton University Press.

[14] Goetzmann, W. N. (2016). *Money changes everything: How finance made civilization possible.* Part 3: The European Crucible. Princeton University Press.

[15] Huthinson, Emma (2016). *Principles of Microeconomics*. Chapter 6.2: The Indifference Curve. University of Victoria. https://pressbooks.bccampus.ca/uvicecon103/chapter/6-3-how-changes-in-income-and-prices-affect-consumption-choices/

[16] Kereiakes et al (2019, April). "Terra Money: Stability and Adoption". https://s3.ap-northeast-2.amazonaws.com/terra.money.home/static/Terra_White_paper.pdf?201904

[17] Leshner, Robert and Geoffrey Hayes (2019, February). "Compound: The Money Market Protocol". https://compound.finance/documents/Compound.Whitepaper.pdf

[18] Martinelli, Fernando and Nikolai Mushegian (2019, September 19). "Balancer: A non-custodial portfolio manager, liquidity provider, and price sensor." https://balancer.finance/whitepaper.html

[19] Nakamoto, Satoshi (August, 2008). "Bitcoin: A Peer-to-Peer Electronic Cash System". https://bitcoin.org/bitcoin.pdf

[20] Noyes, Charlie. (2019, October 15). "Market Completeness and Arbitrage Freeness in DeFi". Youtube. Retrieved 2019, November 26. https://www.youtube.com/watch?v=mpx1KRMeFjA

[21] SIFMA (2019, September). "Capital Markets Fact Book 2019". Retrieved 2019, November 26. https://www.sifma.org/wp-content/uploads/2019/09/2019-Capital-Markets-Fact-Book-SIFMA.pdf

[22] Soleimani, Ameen (2019, September 4). "What You Should Know Before Putting Half a Million DAI in Compound". Medium. Retrieved 2019, November 26. https://medium.com/@ameensol/what-you-should-know-before-putting-half-a-million-dai-in-compound-fafdb2645f77

[23] Sprague, O. M. W. (1914). The federal reserve act of 1913. *The Quarterly Journal of Economics*, 28(2), 213-254.

[24] Tether (2016, June). "Tether White Paper". https://tether.to/wp-content/uploads/2016/06/TetherWhitePaper.pdf

[25] Uniswap.Info (2019, November 26). "Uniswap Overview". Retrieved on 2019, November 26. https://uniswap.info/home

[26] The US Department of State: Office of the Historian (2018). "Nixon and the End of the Bretton Woods System, 1971–1973". Retrieved 2019, July 21: https://history.state.gov/milestones/1969-1976/nixon-shock

[27] Warren, Will and Amir Bandeali (2017, February). "0x: An open protocol for decentralized exchange on the Ethereum blockchain". https://0x.org/pdfs/0x_white_paper.pdf

[28] Weisstein, Eric W. "Complete Graph." *MathWorld–A Wolfram Web Resource.* Retrieved on 2019, November 25. http://mathworld.wolfram.com/CompleteGraph.html

[29] Wikipedia contributors. (2019, July 3). “Shell money”. *Wikipedia, The Free Encyclopedia.* Retrieved 2019, July 21: https://en.wikipedia.org/w/index.php?title=Shell_money&oldid=904661921

[30] Wikipedia contributors. (2019, November 25). “List of systemically important banks”. *Wikipedia, The Free Encyclopedia.* Retrieved on 2019, November 25.https://en.wikipedia.org/w/index.php?title=List_of_systemically_important_banks&oldid=927453646

[31] 貝. (2019, July 15). Wiktionary, The Free Dictionary. Retrieved 2019, July 22: https://en.wiktionary.org/w/index.php?title=%E8%B2%9D&oldid=53661334