Prerequisites

- A Note on Cryptocurrency Stabilisation: Seigniorage Shares
- Spencer Applebau: Potential Soros Attack against SS stablecoins

Motivation

IOU stablecoins such as USDT and USDC carry counterparty risk. Collateralized-debt Stablecoins such as DAI carry collateral, governance and price feed oracle manipulation risks. In an attempt to mitigate these risks, the idea of Seignorage Shares Stablecoins has surfaced since 2014 as a third alternative but is yet to be proven.

SS Stablecoins have so far faced the following challenges:

- SS Stablecoins require a price feed oracle operated by a central operator, which requires trusting a private entity.
- Private entities maintaining SS Stablecoin systems are subject to securities regulations worldwide. Basis was shut down in 2018 due to regulatory pressure.
- SS Stablecoins may be susceptible to Soros Attacks, where long-term confidence can be broken by short-term peg fluctuations.

Previous work on SS Stablecoins attempted to create a peg between the stablecoin and the target currency (e.g. USD) by introducing a two-token model, a stablecoin token and a seigniorage share token.

If the price of the stablecoin rises above the pegged asset price, the smart contract mints more stablecoin tokens and offers them for sale in exchange for the share token. If the stablecoin price falls under the peg, new shares are minted and are offered for sale in exchange for the stablecoin token.

The mechanism above is susceptible to the Soros Attack because the peg may temporarily break to a degree where confidence in a future peg recovery may be lost. Therefore, investors become less willing to buy stablecoins from the contract the further down the price drops from the peg.

Proposal

I propose a hard-peg USD stablecoin algorithmically-priced by a smart contract that directly regulates the price

instead of supply or demand. The on-chain price of USDC or any other IOU stablecoin may be used as an on-chain price oracle. Once deployed, no parameters or special roles on the smart contract need to be controlled by any operator or governance mechanism.

Architecture

When deployed, the stablecoin contract creates a Uniswap V2 token pair between the newly-created token and USDC. The contract then becomes responsible for ensuring that the liquidity available for both tokens is always identical.

The contract exposes a public peg()

function that can be called at any time by externally-owned accounts. Contracts are forbidden in order to prevent flash loan market manipulations.

When called, peg()

ensures that the token pair liquidity amounts are equal. If the USDC liquidity exceeds that of the stablecoin, the stablecoin contract mints the difference and donates

it to the token pair contract. If the USDC liquidity is below that of the stablecoin, the contract seizes

the difference directly from the token pair stablecoin balance and burns it. After either of these operations, Uniswap Pair.sync()

is called to force

the pair reserves to reflect the new balance.

This effectively transforms Uniswap pool tokens into seigniorage shares for the stablecoin. If market demand increases for the stablecoin, the value of a each pool share increases as more tokens are minted and donated to the pool. If market demand decreases, the value of each pool share decreases as more tokens are seized from the pool.

Soros Attack Defensibility

The goal of this design is for the smart contract to directly enforce the price of the stablecoin on the exchange by algorithmically adding and removing liquidity instead of creating any incentives for buyers. Therefore, the peg is not reliant

on investor confidence which defends it against Soros Attacks.

Please note that this is obviously a simplistic initial design. I would appreciate some thoughts on potential flaws and attacks.