VADER PROTOCOL: Incentivised liquidity, stablecoin and lending platform

Abstract. VADER is a liquidity protocol that combines a collateralized stablecoin with liquidity pools. The stablecoin, USDV, is issued by burning VADER tokens. Liquidity pools either use USDV and VADER as the settlement asset, with a zero-slippage conversion available between the two. A daily emission rate of VADER funds liquidity incentives, a protocol interest rate, impermanent loss protection and a float to facilitate protocol lending. Synthetic assets are minted from liquidity pool shares which are eligible to earn an interest rate, as well as being collateral for borrowing debt against. The collateral is used to pay an interest rate which is added into the pools to increase returns.

Introduction

Stablecoins and synthetic assets are a noble problem to solve. The key problem is a matter of liquidity and sensing the correct purchasing power of assets at all times. In addition, the use of incentives can ensure the maximum uptake of the system and the fast bootstrapping of capital. Existing stablecoins and synthetic asset designs fall short because they use oracles that are not liquidity sensitive and can be manipulated, do not incentivise the makers of liquidity properly, and are not natively interest-bearing.

VADER is a new form of liquidity protocol that seeks to be self-serving. It uses its own liquidity and awareness of asset purchasing power to support the creation of a collateralized stablecoin. It also is capable of using liquidity units as collateral for synthetic assets, of which it will always have guaranteed redemption liquidity for. It has a fair and transparent incentive strategy to maximise the depth of liquidity pools and adoption of synthetic assets. It uses a liquidity-sensitive fee to ensure safe and sustainable creation of debt, which can increase the capital efficiency of the system.

Key Features

The following are the key features of VADER Protocol:
1) Uses a collateralized stablecoin settlement asset
2) An ability to burn VADER to mint USDV
3) Impermanent Loss protection for Liquidity Providers in the pools
4) Continuous liquidity pool incentives
5) An ability to mint interest-bearing synthetic assets from pool liquidity
6) An ability to borrow debt against USDV, VADER or Synthetic Assets.

Architecture

There are two types of pools in the system, although it is abstracted to the user. The first group of pools are the "Anchor pools" which use VADER as a settlement asset. This allows the system to sense the anchor price, which is the median of the prices of the anchor pools. The second group of pools are the "Asset pools" which use USDV as the base asset, which drives liquidity and demand of the stablecoin. The Anchor pools [VADER:stablecoin] are linked to the Asset pools [USDV:Asset] via 0-slippage swaps between USDV<>VADER.

VADER Contract

VADER has extra functions to ERC20:
1) VADER is minted if VETH is burnt, 1000:1
2) VADER is minted if USDV is burnt, in accordance with the inverse Anchor price
3) Has a daily emission rate of VADER, which is sent to the USDV Contract.

USDV Contract

USDV has extra functions to ERC20:
1) USDV is minted if VADER is burnt, in accordance with the Anchor price
2)
3) Receives VADER from the VADER contract, of which it burns a proportion of it into USDV, then makes that available to be claimed by USDV stakers. The rest it sends to the Router Contract to be used by its Reserve.

VADER ROUTER and POOLS Contract

The ROUTER wraps the POOLS contract which contains the assets in the pools. POOLS maps liquidity ownership of members, which the ROUTER users to enable Impermanent Loss protection. The POOLS Contract is wrapped by the Router. The ROUTER has a reserve which facilitates:
1) Incentives for each pool
2) Impermanent Loss Protection
3) Minting/Burning of Synthetic Assets
4) Protocol Lending

VADER FACTORY and SYNTHS Contract

An ERC20 Factory is allowed to be called by the Router to deploy and register Synthetic Assets, which are vanilla ERC-20s.

VADER VAULT Contract

Allows anyone to lock up Synthetic Assets and earn an interest rate, claimable every block.

VADER UTILS Contract

The Utils contract contains logic for the system.

VADER DAO Contract

The DAO senses the weight of each member (based on share of the weight in the vault) and allows system parameters to be changed with various consensus levels.

VADER Token

There are three use cases of VADER. The first is as a common settlement asset in anchor pools to enable the system to sense the purchasing power of a group of stablecoins - this transmits the "anchor" price in USD. The second is to allow anyone to burn VADER to mint USDV at 1:1 the anchor price. The third is to allow anyone to lock up VADER as collateral for borrowing against.

VADER is issued 1000:1 for holders of Vether (VETH) - which itself is distributed via a fair process of Proof-of-Value. Since VETH is acquired only by the provable burning of ETH; it is sybil-resistant, decentralised and has unforgeable costliness. VETH has a maximum supply of 1,000,000 units, issued over 10+ years.

VADER has a maximum supply of 2bn. 1bn is claimed by the holders of VETH, and the additional 1bn is paid out based on a smooth emission curve starting at 50% APY and dropping to roughly 10% after 5 years. VADER will continue to pay out until 2bn units have been issued. VADER contains a Fee-On-Transfer which is proportional to the current supply, but increases linearly the closer VADER is to its Maximum Supply in the range from 0 to 100 Basis Points. As such, the emission rate can be offset by the burn rate at some point into the future.

```
dailyEmission = (maxSupply - currentSupply) / emissionCurve
```

Liquidity Incentives

Dividends are paid out to Liquidity Pools, both in the form of VADER and USDV, as well as paying SYNTH stakers. The incentives are synced into pool balances each time a swap is made, so over time LPs realise a yield. This yield, as well as slip-based fees, offsets any Impermanent Loss, and the LP should always realise a gain after a period of time.

Impermanent Loss Protection

The deposit value for each member is recorded when they deposit. When they go to withdraw, the redemption value is computed. If it is less than the deposit value, the member is paid the deficit from the reserve. The protection issued increases from 0 to 100% linearly for 100 days. Coverage is given by the following equation:

```
coverage = (V0 - V1) + (A0 - A1) * V1/A1
V0: USDVDeposited; A0: assetDeposited;
V1: USDVToRedeem; A1: assetToRedeem;
```

Liquidity Pools

Liquidity Pools use either VADER or USDV as the common settlement asset. This allows the system to accurately price any pool, as well as sensing purchasing power of its assets. Using USDV as a common settlement asset in Asset pools takes away any friction for requiring users hold exposure to a particular asset. In fact, all of the VADER Liquidity Pools (Anchor and Asset) are stablecoin-paired pools, making Impermanent Loss easy to reason about.

The liquidity model includes a liquidity-sensitive slip-based fee. This maximises revenue for liquidity providers under high demand of liquidity, and prevents manipulation. It is also necessary to support liquidations of collateral. The algorithm is:

$$y = \frac{x*X*Y}{(x+X)^2}$$

```
x: input; X: Input Balance;
y: output; Y: Output Balance;
```

Slip-based fees break path-independence and a member can theoretically achieve better price execution by making smaller trades. This characteristic is actually favourable - smaller trades slow down the rate at which a price can change, and the base-layer gas fee paid (fixed in cost) provides a lower-level threshold of tolerance. Additionally, there is no freedom of choice of trade size of a position when it is liquidated by the protocol, arguably the most important aspect to the system.

Anchor Pools

Anchor pools are special pools that specifically have VADER as the base asset. Anyone can create a pool, but there are requirements to meet for creating a new Anchor Pool, and only some of them are allowed to be part of the pricing mechanism. The conditions to list/delist an anchor pool:

1) The protocol will start with 5 anchor pools (DAI, USDT, USDC, BUSD, UST)
2) At any stage, an anchor pool can be replaced by another, as long as the conditions are met:
   a) Is within 2% of the price of the other four
   b) The pool to be replaced is outside 5% of the other four
   c) Has a depth that exceeds the pool to be replaced (in VADER)

Synthetic Assets

Synths are issued from swapping from any other asset into Synths across the pool. Instead of the paired asset being emitted, the incoming base asset is added to the pool as liquidity, liquidity units are issued to account for the new capital and assigned to the protocol, and the synth is minted. When synths are swapped back to a token, they are burnt, a pro-rata share of units are deleted, and the base asset is moved out. Even though capital is added and removed, the liquidity units that are issued and deleted account for the capital such that other passive LPs are not affected.

$$units = \frac{P * b}{2(b+B)}$$

```
P: existingUnits;
b: inputBase; B: baseBalance;
```

Since synths always have 1:1 purchasing power they do not accrue Impermanent Loss or Yield. In fact, any loss or gain is absorbed by the other passive LPs. With IL Protection, other passive LPs should never be affected by this.

Synthetic Assets can be staked to earn an interest rate, or used as collateral for borrowing. Since synths are 50% collateralized by the real assets and 50% collateralized by USDV or VADER, heavy adoption of synths deepens the liquidity pools and allows the system to naturally scale (or contract).

Lending

Members can lock up collateral, which includes VADER, USDV or any synthetic asset. VADER/USDV is taken from the reserve and swapped out to the debt asset. The member's collateral is added to a collateral pool, and their debt recorded. Since there is global collateral pool, members can not adjust collateral rate themselves - all debt is issued at 150% collateralization when minted. Interest is charged against the collateral and paid into the pools of the associated debt asset. A member can unlock their collateral by returning their debt amount. If the collateral drops below the debt value, then anyone can partially liquidate the pool of collateral, set at 10% each time.

The Interest Rate is set to be proportional to the debt loading of the system, so the more debt is issued, the higher the rate, which provides a natural throttle. Interest payments are made daily and synced into pool balances, which should drive up yield for LPs and encourage deeper liquidity. This will in turn reduce the interest rate:

```
interestRate = debtIssued / assetDepth
```

The interest paid is tracked, such that when a member leaves, their redeemed collateral is their deposit minus their share of the paid interest. If the paid interest for a member is within 99% of their collateral, anyone can purge that member and claim the final 1% collateral. This will stop a build up of members who have consumed all their collateral in interest, but never leave, which adversely affects the accounting for the remaining members.

The share of interest is tracked by an accumulator which multiplies time by collateral sizing (priceTime). This is updated each time a member borrows or repays, and their share of the interest corresponds to their share of the accumulated priceTime.

```
interestShare = memberPriceTime / (systemPriceTime +
accumulatedPriceTime)
```

Liveness

The system is able to conduct its own servicing:
- Pay the Daily Emission Rate
- Pay the Dividends
- Liquidate collateral 10% at a time
- Pay the Daily Interest Rate

This is done by piggy-backing VADER transfers, USDV transfers and ROUTER actions which query a condition and execute. The member making the transfer/action will subsidise the gas cost at that time. If they don't wish to pay the gas, they can wait for someone else to do it.

Reserve

The VADER Vault contract has a reserve, which is topped up by emissions every day, and consumed by incentives, IL Protection and borrowers. This is to ensure the emission rate of VADER is programmed and immutable, and is not dependent on downstream requirements. As such, under heavy demand, the Reserve can go to 0, in which case incentives, protection and lending will stop for the rest of the day. This is a throttle on the system, preventing run-away inflation.

Governance

System parameters can be tweaked by DAO Governance, but this is strictly limited. The DAO can be purged by itself if the system is stable and no more parameters need tweaking. During the bootstrap period, the DEPLOYER will retain the ability to skip DAO governance, which can be purged by anyone after 3 months. A liquidity limit will be in place at the same time. The reserve of USDV in the USDV contract pays USDV stakers, who can decide to pay out grant funding to anyone in the ecosystem, (limited to 10% at a time).

Conclusion

VADER is an incentivised, governance-minimal and cohesive liquidity protocol that can scale itself sustainably. A stablecoin is issued by burning VADER, which itself is priced against the median of a group of anchor pools. This stablecoin allows the bearer to earn interest, as well as being the settlement asset in most of its liquidity pools. Liquidity providers are entitled to Impermanent Loss protection whilst they are in the pools. All pools receive daily liquidity incentives. A lending design allows members of pools and holders of VADER to lock up collateral to borrow debt against.

Appendix A - Code Architecture

VADER - ERC20

* ERC20
* Mints when given VETH
* Mints when given vUSD
* Daily Emission Rate to vUSD
* Knows USDV contract address

USDV - ERC20

* ERC20
* Mints when given VADER
* Pays dividends to stakers
* Pays dividends to ROUTER, can compute share of rewards
* Knows ROUTER address

ROUTER

State
* Has LP deposits, IL tracking
* Has collateral, debt tracking

Functions
* Liquidity logic: Swap, Add, Remove
* Dividend Logic: pay incentives per swap
* IL Logic: pay deficit
* Lending logic: Mint, Burn, Liquidate, Purge
* Anchor logic: listing, pricing

POOLS

State
* Has reserve balance
* Has pool balances
* Has LP balances,

Functions
* Liquidity logic: Swap, Add, Remove
* Synth logic: Mint, Burn

FACTORY

State
* Synth Contract
* Synth registry

Functions
* Deploy Synth

SYNTHS - ERC20

* Programmatically deployed

DAO

State
* Has Proposal Tracking

Functions
* Create, Vote, Replace Proposal
* Action Proposal

DAO

Functions
* System
* Pricing
* Lending
* Core-math
* Helpers