PerpDEX Protocol

richmanbtc
richmanbtc@perpdex.com
@richmanbtc2

April 30, 2022

Abstract

We propose a new perpetual future DEX protocol called PerpDEX. There are three problems in existing perpetual future DEXs. The first problem is that it has low composability because it does not support tokenization. The second problem is security concerns in the oracle-dependent and liquidation-related parts. The third problem is that decentralization is incomplete because the source code is not open to the public or the centralized part remains. PerpDEX solves these problems. The features of PerpDEX are (1) the new method called liquidation free funding (LFF) allows positions to be tokenized with ERC4626 (2) High security due to reduced oracle dependencies and security measures such as a new feature called the two stage price limit (TSPL). (3) It is fully decentralized and developed in open source without the centralized part that existed in the conventional protocol. We will implement it in the future and demonstrate the effectiveness of this protocol.

1 Introduction

The perpetual future has the highest trading volume in the cryptocurrency market, with 43% of the cryptocurrency trading volume in 2021 being spot and 50% being the perpetual future [1]. On the other hand, the transaction volume of perpetual future in DEX is still small, and the share of perpetual future transaction volume in 2021 of dYdX [2], which has the largest transaction volume in perpetual future DEX, is 2.2% or less [1]. If DEX's trading volume share increases to 50%, perpetual future DEX has more than 20 times more room for growth.

The background of perpetual future DEX growth is the improvement of chain performance. Recently, non-ETH chains other than Ethereum have been growing. The TVL share of non-ETH chains in January 2021 is 3%, while the TVL share of non-ETH chains in April 2022 is 46% [3]. These non-ETH chains are Terra, BSC, Avalanche, Solana, Fantom, Tron, Polygon, Cronos, Arbitrum in descending order of TVL share [3]. These non-ETH chains tend to be superior to Ethereum in terms of gas costs and transaction speed. There are two points related to perpetual future DEX. The first point is the gas fee. DEX trading volume tends to move from chains with high gas prices to chains with low gas prices [4]. If the chain performance improves and the gas price goes down in the future, the transaction volume of the entire DEX including the perpetual future DEX is expected to increase. The second point is Oracle. Oracle is required to implement perpetual future DEX. As the transaction speed of the chain increases, it becomes possible to create oracles with high update frequency, and it becomes easier to implement perpetual future DEX. From these points of view, it is expected that the perpetual future DEX will continue to grow as the performance of the chain improves.

1.1 Problem

As mentioned above, while the perpetual future DEX seems to have room for growth, the current perpetual future DEX has problems. The first is the issue of composability. Composability is a property that can be freely linked with other protocols, and is one of the important properties

of DeFi. The important thing about composability is tokenization, which improves composability by tokenizing various functions of the protocol, such as LP tokens of automated market maker (AMM). However, the conventional perpertual future DEX does not support tokenization and its composability is low. The second is the Security issue. The traditional perpetual future DEX has security concerns in the oracle-dependent and liquidation-related parts. The third is the problem of decentralization. Some conventional perpetual future DEXs have a centralized part, and some cannot verify the correctness of the protocol because the source code is not open to the public. These issues are discussed in detail in Section 2.

1.2 Solution

PerpDEX solves the above problem. First, a new feature called liquidation free funding (LFF) is used to solve the problem of composability by tokenizing the perpetual future position to ERC4626. PerpDEX can issue long tokens and LP tokens that tokenize perpetual future positions with 100% capital efficiency. A long token is similar to Synthetix's Synthetic assets [5] and Squeeth long token [6], and is a token that tracks the Oracle prices referenced by the market. LP tokens are similar to AMM's LP tokens and are tokens that track the square root of the Oracle price referenced by the market and earn transaction fees. The key to tokenization is LFF. By using LFF, the perpetual future positions contained in the token will not be liquidated. This allows positions to be tokenized without rebalance.

Next, we will solve the security problem by reducing the dependence on Oracle and taking various measures including a new method called the two-stage price limit (TSPL). Oracle is vulnerable to attacks due to price manipulation and other factors. PerpDEX uses oracle only for funding rate calculations, making the protocol less susceptible to attacks or corruption. As a side effect, even if you use Oracle, which is updated infrequently, you will be able to create a market, so it will be easier for many chains to provide many markets.

Finally, the problem of decentralization is solved by eliminating the centralized part and developing it in open source. Since the source code of our protocol is open to the public, anyone can verify the correctness of our protocol. Improve decetralization by eliminating the need for the trusted liquidator, which was the centralized part of the traditional Perpetual Protocol V2 [7]. The trusted liquidator was introduced as a security measure, and its elimination can be achieved by introducing TSPL. These mechanisms will be explained in detail in Section 3.

2 Related Work

In this section, introduce the features and problems of existing perpetual future DEXs. Points are divided into composability, security, and decentralization.

2.1 Composability

Composability is a property that a protocol can freely cooperate with other protocols, and is one of the important properties of DeFi. There are four important points regarding composability.

The first is the architecture of the perpetual future DEX. This can be divided into those using automated market maker (AMM) and those using central limit order books (CLOB). Protocols using AMM include Perpetual Protocol V2, MCDEX [8], and Hubble Exchange [9]. CLOB-based protocols include dYdX and Mango Markets [10]. The advantages of AMM are the low gas cost and the tokenization of the liquidity provider (LP). The advantage of CLOBs is that LPs are more capital efficient than AMMs [4]. There is also a way to avoid the gas cost of CLOB by implementing CLOB off-chain as is done in dYdX. However, if implemented in off-chain, it will not be possible to link with other contracts, so it is difficult to achieve both low gas costs and composability with CLOB. From the point of view of composability, AMM is suitable because it can reduce the gas cost without using off-chain and can tokenize LP.

The second is AMM's LP. There are several types of AMM LPs. The first is fungible LP. It is used in Uniswap V2 [11] and Curve Finance [12]. The advantage is that it can be tokenized to ERC20 and no management is required. Next, non fungible LP. This is an LP that provides liquidity by specifying a price range and is used in Uniswap V3 [13]. It has the advantage of high financial efficiency because it allows liquidity to be concentrated. However, there is a disadvantage that cannot be converted into ERC20 tokens and requires management effort. The last is LP management. This is a method that automatically operates LPs such as Uniswap V3, and corresponds to Charm Finance [14] and ICHI's Angel Vaults [15]. It has the advantages of high capital efficiency, ERC20 tokenization, and no management effort. However, there is a risk of losing money due to operational failure [16]. In the case of perpetual future DEX, the capital efficiency of AMM itself is not essential because the capital efficiency can be increased by leverage. Compared to other than capital efficiency, fungible LP is suitable because it can be tokenized to ERC20, has high composability, and has low management risk.

The third is liquidation free. In this white paper, we call the property that the perpetual future position is not liquidated under any circumstances, as liquidation free. Since the position of the conventional perpetual future DEX is not liquidation free, it is difficult to tokenize it. The reason why it is not liquidation free is funding payment. The conventional funding payment mechanism is to pay funding in quote tokens and reflect it in collateral as realized PnL. While this is familiar with the method used in CEX, it has the disadvantage that any positions are exposed to liquidation risk. Specifically, if the cumulative loss due to funding exceeds the original position size, it will lead to liquidation. This makes position tokenization difficult and reduces composability. UXD Protocol [17] and Lemma Finance [18] are stablecoin protocols that build delta-neutral positions in perpetual future DEX and tokenize those positions. The advantage is that the position is delta neutral, so it is difficult for the position to be liquidated. The disadvantages are the need for regular rebalancing to maintain delta neutrality and the non-zero risk of liquidation. Squeeth [6] is a liquidation free derivative protocol. You can issue tokens whose value is linked to the square of the ETH price. The method called normalization factor used in this protocol is similar to the liquidation free funding proposed in Section 3.2 in that it uses global scaling variables to handle funding payments. The advantage is that there is no risk of liquidation. The disadvantage is that it is not a perpetual future DEX, so the degree of freedom in trading is low. In terms of composability, a liquidation free protocol such as Squeeth is suitable.

The fourth is ERC4626. ERC4626 is an extension of ERC20 that can be used to standardize vaults such as staking and Yearn Finance [19]. Like ERC20 and ERC721, ERC4626 contributes to composability. ERC4626 will be implemented in Yearn Finance, Alchemix, Balancer, Rari Capital, Fei Protocol [20]. In the perpetual protocol DEX, ERC4626 can be used to tokenize positions.

Based on these points, we will create a composabile perpetual future dex with a new method called liquidation free funding (LFF) and ERC4626 tokenization. Details will be explained in Section 3.

2.2 Security

Security is important in DeFi. Oracle and liquidation are particularly prone to security issues in perpetual future DEX.

2.2.1 Oracle

Oracle is used to implement the perpetual future DEX. Although the purpose of using Oracle differs depending on the protocol, Oracle is used for funding rate calculation, liquidation judgment, trade price calculation, and so on. All of the protocols mentioned in this white paper use Oracle to calculate funding rates. In dYdX, Perpetual Protocol V2, MCDEX, Synthetix [21], oracle is used for liquidation judgment. In MCDEX, Synthetix, Oracle is used to determine the trade price.

Oracle is used to implement the perpetual future DEX. Although the purpose of using Oracle differs depending on the protocol, Oracle is mainly used for funding rate calculation, liquidation judgment, and trade price calculation. All of the protocols mentioned in this white paper use

Oracle to calculate funding rates. In dYdX, Perpetual Protocol V2, MCDEX, Synthetix [21], oracle is used for liquidation judgment. In MCDEX, Synthetix, Oracle is used to determine the trade price.

Minimizing the use of Oracle reduces security risks. Here are two specific examples of attacks on perpetual future DEX. The first example is when an oracle is attacked with a protocol that uses oracle for liquidation determination. In this case, the attacker may forcibly liquidate other traders and generate bad debt, thereby robbing the assets of the insurance fund. The second example is when Oracle is attacked with a protocol that uses Oracle to determine trade prices. In this case, the attacker may steal the assets of other users by trading at a price favorable to the attacker.

PerpDEX solves these problems by using Oracle only when calculating the funding rate and minimizing its dependence on Oracle. Details will be explained in Section 3.6.

2.2.2 Liquidation

Liquidation is required to implement perpetual future DEX, but liquidation is easily targeted by attacks. Especially the liquidation attack is a problem. A liquidation attack is an attack in which an attacker profits by manipulating the price used for liquidation judgment in some way and forcibly generating liquidation of another trader [22]. Existing protocols provide countermeasures against this attack. Here are three measures taken in Perpetual Protocol V2.

The first is to use the TWAP of Oracle price for liquidation judgment. This makes it difficult to manipulate the price used for liquidation judgment, so it is effective for liquidation attacks. The disadvantage is that the liquidation price arbitrage attack (D), which is caused by the difference between the price used for PnL calculation and the price used for liquidation determination, is likely to occur.

The second is to allow only trusted liquidators authorized by admins to liquidate bad debt traders. Bad debt is a state in which the trader's asset valuation is negative. When bad debt occurs, the negative amount is compensated from the insurance fund, which leads to the outflow of assets of the insurance fund. This measure reduces the risk of protocol bankruptcy by preventing attackers from using bad debt to steal insurance fund assets. The disadvantages are that it reduces decentralization and does not prevent the liquidation attack itself.

The third is price restrictions using the rate of price change from the previous timestamp. This is a method in which all orders are allowed if the limit is not exceeded, and only partial closed orders and partial liquidation are allowed if the limit is exceeded. The advantage of this method is that outside the price limit, it is not possible to perform a liquidation attack with an open order for a position, which makes the attack difficult. There are two disadvantages. First, a liquidation attack can be performed by a closed order even outside the price limit. Second, because it does not control the priority of regular orders and liquidation orders, there is still the possibility that regular orders and MEV attacks will delay liquidation and increase bad debt.

PerpDEX proposes a new method called two-stage price limit (TSPL) to solve these problems. Details will be explained in Section 3.5.

2.3 Decentralization

Decentralization is an important value of DeFi, but some perpetual future DEXs have problems with decentralization. There are two main problems. The first is the problem that the source code of the protocol is not open to the public. Since the latest version (V3) of dYdX and Hubble Exchange are not open source, there is a problem that the correctness of the contract logic cannot be verified. The second problem is that the protocol has a centralized part. dYdX uses a centralized orderbook and matching engine [23]. In Perpetual Protocol V2, liquidation for traders in bad debt is not permissionless, and only trusted liquidators permitted by admins can liquidate. PerpDEX solves these problems and improves decentralization.

3 Proposed Methods

In this section we explain how PerpDEX works. PerpDEX is based on Perpetual Protocol V2, a perpetual future DEX that uses vAMM, with some improvements. First, we made improvements for composability. This includes an improvement to create liquidation free perpetual future positions called liquidation free funding (LFF) and ERC4626 tokenization for perpetual future positions using it. Next, we made improvements for security and decentralization. This includes a price limit called the two-stage price limit (TSPL), improvements to reduce Oracle dependence, and improvements to liquidation. Table 1 shows the differences from Perpetual Protocol V2. Figure 1 shows the architecture overview.

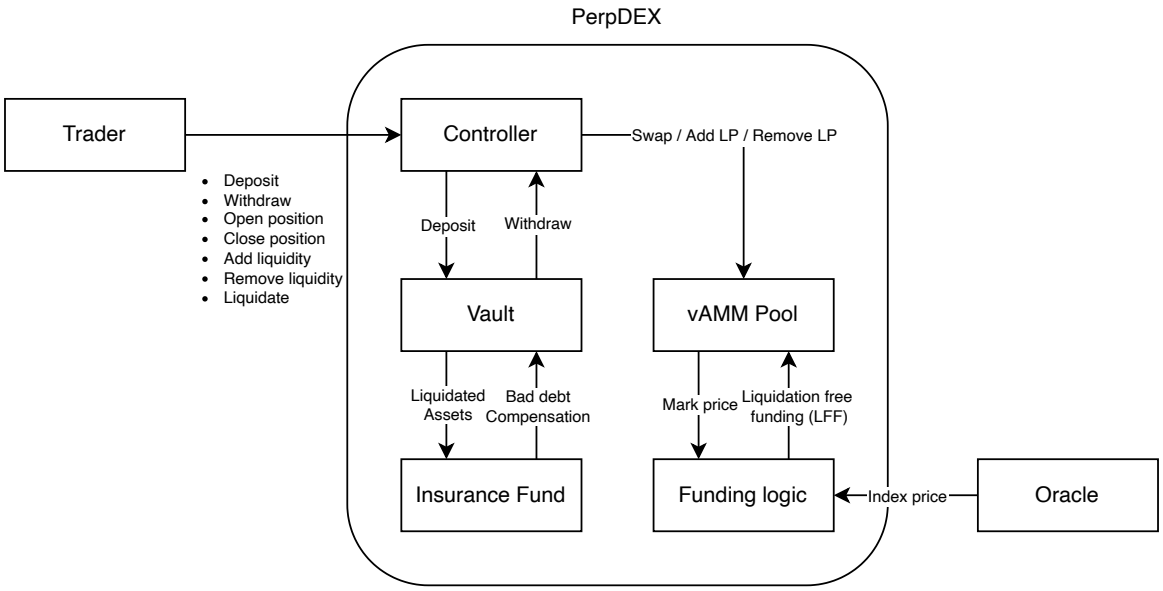

Figure 1: PerpDEX Architecture

Table 1: PerpDEX vs Perpetual Protocol V2

	PerpDEX (proposed)	Perpetual Protocol V2
Pool type	$xy = k$	Uniswap V3
Funding currency	Base	Quote
Funding target	Position or Liquidity	Collateral
Liquidation free	Yes in certain conditions	No
Price for PnL	vAMM price	vAMM price
Price for liquidation	vAMM price	TWAP of Oracle price
Price limit	Two-stage (TSPL)	Single-stage
Liquidator	Permissionless	Trusted when bad debt
Liquidation reward	A part of liquidated position	A part of liquidated position
Insurance fund fee	Remaining assets after liquidation	A part of transaction fee

3.1 vAMM

PerpDEX implements perpetual future using vAMM like Perpetual Protocol V2. The reason for adopting vAMM is that since AMM is the mainstream in spot DEX, there is a high possibility that AMM will work in perpetual future DEX, and it will be easier to tokenize LP, which will

lead to improved composability. The pool adopted a simple $xy = k$ type. There are two reasons. First, in the case of perpetual future, unlike spots, it is possible to leverage and increase capital efficiency, so there is little need to concentrate liquidity on the AMM side. Second, if the LP is fungible, the position can be tokenized as ERC4626, leading to improved composability.

3.2 Liquidation free funding (LFF)

We propose a new funding method called liquidation free funding (LFF) that is needed to tokenize perpetual future positions and improve composability. The mechanism of funding used in the conventional Perpetual Protocol V2 is to pay funding in quote tokens and reflect it in collateral as realized PnL. While this is familiar with the method used in CEX, it has the disadvantage that any position is exposed to liquidation risk. This makes position tokenization difficult and impairs composability. In LFF, funding is paid in base tokens and reflected in positions instead of collateral. This makes it easier to tokenize positions and improves composability.

We explain how LFF works. First, the funding rate is calculated from TWAP, which is the difference between the mark price (vAMM pool price) and the index price (oracle price). The calculation method is the same as Perpetual Protocol V2, so it is omitted. Next, the funding is paid using the funding rate. The amount of funding paid by each trader is the product of each trader's position (base token balance) and the funding rate. Since funding is paid by base token in LFF, if base token is treated as a rebase token, it can be seen that funding payment can be processed by changing the total supply of base token.

We explain with mathematical formulas. We denote the funding rate at time $t \in \mathbb{Z}$, the total supply of the base token as $S(t)$, the balance of the base token as $b(t)$, and the shareholding of the base token as $h(t)$. $S(t)$ and $b(t)$ are expressed by the following equations. When the funding rate is positive, the total supply decreases, and when it is negative, the total supply increases.

$$S(t+1) = S(t) \cdot (1 - f(t)) \quad (1)$$

$$b(t) = S(t) \cdot h(t) \quad (2)$$

3.3 Liquidation free region

We show that when using LFF, the trader's position is liquidation free if certain conditions are met. This is a property not found in conventional funding payments and is required for position tokenization. In PerpDEX, traders can freely combine the taker position and the LP position (maker position). The taker position is a typical perpetual future position such as long or short. The LP position is a position that provides liquidity to vAMM. If the trader has a combined position of the taker position of leverage u and the LP of leverage v, the position in the area of Figure 2 will be liquidation free. This region is defined as a liquidation free region. For example, 1x Long and 0.5x Long + 0.5x LP are liquidation free. Note that not all positions in PerpDEX are liquidation free, and positions outside the liquidation free region are at risk of liquidation. The derivation method is described in B.

3.4 ERC4626 tokenization of liquidation free positions

We propose two types of ERC4626 tokens that tokenize liquidation free positions to improve composability. Note that these tokens are created as one of the applications using PerpDEX and are not essential to PerpDEX itself. The first token is the Long token. Long token is a tokenized version of PerpDEX's 1x long position, a token that tracks Oracle prices. The second token is the LP token. The LP token is a tokenized version of PerpDEX's 0.5x long and 0.5x LP combined positions, and is a token that tracks the square root of the Oracle price and earns PerpDEX transaction fees. The features are summarized in Table 2. The architecture overview is shown in Figure 3. See C for deriving the token value.

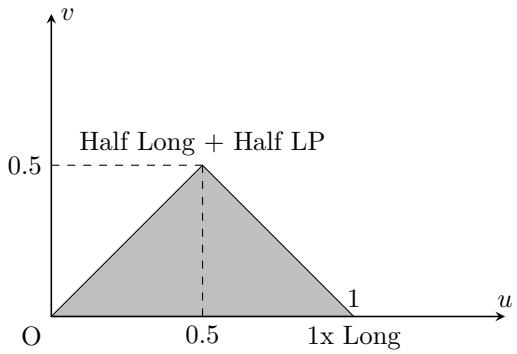

Figure 2: Liquidation free region

These tokens have three characteristics. The first feature is that the perpetual future positions contained in the token are not liquidated because LFF is used. The second feature is that it can represent all positions inside the liquidation free region. As you can see from Figure 2, these tokens are vertices of the liquidation free region, so by creating a portfolio with these tokens, all the positions inside the liquidation free region can be represented. The third feature is that it can be issued with 100% capital efficiency.

These tokens have various applications. Here are some examples. If you make a Long token in the USD / ETH market (ETH / USD inverse), you can make stablecoins like UXD Protocol and Lemma Finance. If you have Oracle, you can use Long tokens to create tokens like Synthetix's synthetic assets. This allows you to track crypto market caps, volatility, tokens that aren't on the chain, and more. If you create a market with the power of ETH / USD price Oracle, you can issue tokens similar to Squeeth long token.

Table 2: ERC4626 tokens containing PerpDEX positions

Token	PerpDEX positions	Token value	Liquidation	Rebalancing
Long	1x Long	Price	Never	Not required
LP	0.5x Long + 0.5x LP	$\sqrt{\text{Price}}$	Never	Not required

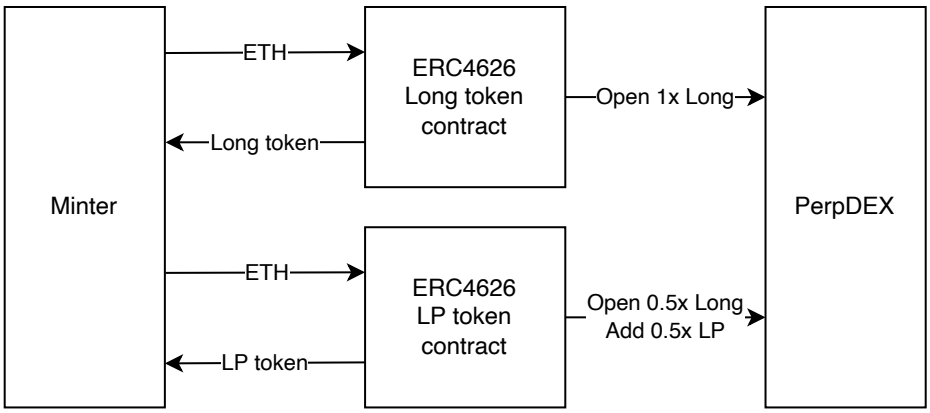

Figure 3: ERC4626 tokenization of PerpDEX positions when quote token is ETH

3.5 Two-stage price limit (TSPL)

We propose a new method called the two-stage price limit (TSPL) to improve security. Perpetual Protocol V2 introduced price restrictions using the rate of price change from the previous timestamp. This means that all orders are allowed if the limit is not exceeded, and only partial closed orders and partial liquidations are allowed if the limit is exceeded. PerpDEX uses a two-stage price limit. The rules are as follows. All orders are allowed as long as they do not violate price restrictions. If the price limit of the first stage is exceeded, only liquidation is allowed. All orders will be rejected if the price exceeds the second price limit. The price change rate from the final price of the previous timestamp is r, the limit values of the first and second stages are $c1$ and $c2$, and the details are shown in Table 3 and Figure 4.

The reason for introducing TSPL is to prioritize liquidation to prevent bad debt and improve security. Bad debt is a state in which the trader's asset valuation is negative. When bad debt occurs, the negative amount is compensated from the insurance fund, which leads to the outflow of assets of the insurance fund, so it is better that the occurrence of bad debt is small. Bad debt occurs when liquidation is delayed. With a simple one stage price limit, if normal orders are prioritized due to bulk ordering of normal orders or MEV attack, liquidation will not be processed and will be delayed, which is a problem. By setting a two-stage price limit and allowing only liquidation between the first and second stages, liquidation will not be delayed and bad debt can be prevented.

Table 3: Two-stage price limit (TSPL)

Price changes	Normal orders	Liquidation
$\|r\| \leq c_1$	Accepted	Accepted
$c_1 < \|r\| \leq c_2$	Rejected	Accepted
$c_2 < \|r\|$	Rejected	Rejected

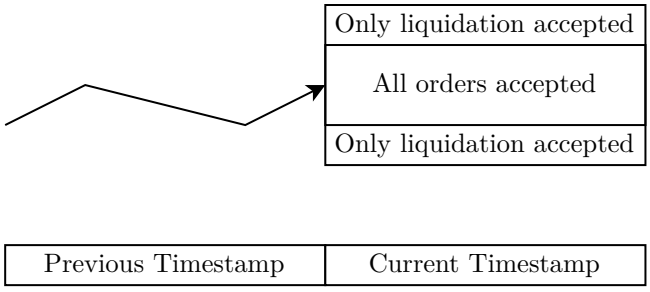

Figure 4: Two-stage price limit (TSPL)

3.6 Minimizing oracle dependencies

We have minimized PerpDEX's dependence on Oracle to improve security. In Perpetual Protocol V2, Oracle is used for two purposes. One is the funding rate calculation and the other is the liquidation judgment. PerpDEX minimized its dependence on Oracle by using the vAMM market price for liquidation determination and changing it to use Oracle only for funding rate calculations.

There are three reasons for the change. The first is to reduce the effects of oracle attacks and oracle errors. Oracle is vulnerable to attacks due to price manipulation and other factors. Reducing reliance on Oracle makes the protocol less susceptible to attacks and corruptions of Oracle. The second is to mitigate the liquidation price arbitrage attack (D) caused by the difference between the price used for PnL calculation and the price used for liquidation determination. The third is to make it easier to add a perpetual future market. Since oracles are used only in funding rate

calculations, it will be possible to create a market even if there are only oracles that are updated infrequently.

3.7 Liquidation

The liquidation process of PerpDEX is almost the same as Perpetual Protocol V2, but two changes have been made to improve decentralization and simplify protocol design.

The first change is to make liquidation permissionless. In Perpetual Protocol V2, bad debt traders could only be liquidated by trusted liquidators allowed by admins, but in PerpDEX anyone can now liquidate. The limitation of liquidator is to deal with bad debt, but in PerpDEX, bad debt can be mitigate by TSPL, so liquidator can be made permissonless. This change improves decentralization.

The second change is to use the remaining assets of liquidated traders as a source of income for the insurance fund. In Perpetual Protocol V2, the source of income for the insurance fund was part of the transaction fee, but in PerpDEX, if any assets remain after liquidation, they are all transferred to the insurance fund to make it a source of income. The reason for the change is to simplify the analysis of the bankruptcy probability of the insurance fund and make it easier to design the protocol. With PerpDEX, the bankruptcy probability can be analyzed as follows. Let a_i be the remaining assets of the i-th liquidated trader. If $a_i < 0$, it means bad debt. Assuming that the balance of the insurance fund after the i-th liquidation is I_i, the initial balance of the insurance fund is I_{init}, and P_{bankrupt} is the bankruptcy probability, P_{bankrupt} can be calculated as follows.

$$I_i = I_{\text{init}} + \sum_{j=0}^{i} a_i \tag{3}$$

$$P_{\text{bankrupt}} = P[\exists i, I_i < 0] \tag{4}$$

P_{bankrupt} is determined by the distribution of a_i and I_{init}. Since the distribution of a_i can be controlled by protocol parameters such as maintenance margin rate and TSPL, the protocol can be designed so that the insurance fund does not go bankrupt if the parameters are selected so that P_{bankrupt} is sufficiently small.

4 Conclusions

In this white paper we proposed a new perpetual future DEX protocol called PerpDEX. PerpDEX has improved three points. First, with the proposal of liquidation free funding (LFF), it became possible to create a liquidation free position. This made it easier to tokenize positions and improved composability. Next, security measures such as two-stage price limit (TSPL) and reduction of oracle dependency were taken to improve security. Finally, decentralization was improved by making the liquidator permissionless and removing the centralized part that existed in traditional protocols. We will implement it in the future and demonstrate the effectiveness of this protocol.

References

[1] TokenInsight Research. TokenInsight 2021 Crypto Trading Industry Annual Review, Jan 2022.

[2] dYdX. dYdX. `https://dydx.exchange/`. Accessed: 2022-04-30.

[3] Defi Llama. Total value locked all chains. `https://defillama.com/chains`. Accessed: 2022-04-22.

[4] KPMG. Crypto Insights #2. Decentralised Exchanges & Automated Market Makers – Innovations, Challenges & Prospects, Oct 2021.

[5] Synthetix. Welcome - Synthetix System Documentation. https://docs.synthetix.io/, Apr 2022.

[6] Opyn Inc. Squeeth. https://squeeth.opyn.co/. Accessed: 2022-04-30.

[7] Perpetual Protocol. Perpetual Protocol. https://perp.com/. Accessed: 2022-04-30.

[8] MCDEX. MCDEX | The Next-Gen Decentralized Perpetual Swap Exchange. https://mcdex.io/. Accessed: 2022-04-30.

[9] Hubble Exchange. Hubble Exchange. https://hubble.exchange/. Accessed: 2022-04-30.

[10] Mango Markets. Mango Markets. https://www.mango.markets/. Accessed: 2022-04-30.

[11] Uniswap Labs. Uniswap V2 Mainnet Launch! https://uniswap.org/blog/launch-uniswap-v2, May 2020. Accessed: 2022-04-30.

[12] Curve Finance. Curve.fi. https://curve.fi/. Accessed: 2022-04-30.

[13] Uniswap Labs. Introducing Uniswap V3. https://uniswap.org/blog/uniswap-v3, Mar 2021. Accessed: 2022-04-30.

[14] Charm. Charm Finance: an ecosystem of innovative products within DeFi. https://charm.fi/. Accessed: 2022-04-30.

[15] Daniel Tal. Angel Liquidity Vaults: Get Comfy this Crypto Winter. https://medium.com/ichifarm/angel-liquidity-vaults-uniswap-v3-supercharged-for-lps-and-crypto-projects-f15bc17b Nov 2021. Accessed: 2022-04-30.

[16] Shaurya Malwa. How Ichi Tokens Plunged 90% After Bad Debt Fiasco on Rari. https://www.coindesk.com/tech/2022/04/12/how-ichi-tokens-plunged-90-after-bad-debt-fiasco-on-rari/, Apr 2022. Accessed: 2022-04-30.

[17] UXD Protocol. UXD Protocol. https://uxd.fi/. Accessed: 2022-04-30.

[18] Lemma Finance. Lemma Finance. https://www.lemma.finance/. Accessed: 2022-04-30.

[19] Joey Santoro, t11s, Jet Jadeja, Alberto Cuesta Cañada, and Señor Doggo. EIP-4626: Tokenized Vault Standard. https://eips.ethereum.org/EIPS/eip-4626, Dec 2021.

[20] Sage D. Young. DeFi Giant Yearn Leads the Way on ERC-4626 Token Standard Adoption. https://www.coindesk.com/layer2/2022/04/08/defi-giant-yearn-leads-the-way-on-erc-4626-token-standard-adoption/, Apr 2022.

[21] Synthetix. Synthetix - Decentralized Perpetual Futures. https://synthetix.io/futures. Accessed: 2022-04-30.

[22] Torgin Mackinga, Tejaswi Nadahalli, and Roger Wattenhofer. TWAP Oracle Attacks: Easier Done than Said? In *4th IEEE International Conference on Blockchain and Cryptocurrency, Virtual Conference*, May 2022.

[23] dYdX. dYdX V4 Full Decentralization. https://dydx.exchange/blog/v4-full-decentralization, Jan 2022. Accessed: 2022-04-30.

A vAMM with Rebase token

As explained in Section 3.2, LFF treats funding payments as a base of base tokens. Here we analyze how vAMM behaves when the base token is a rebase token. Let p be the price, S be the total supply of base tokens, b be the reserve amount of base tokens, and q be the reserve amount of quote tokens. In the case of a rebase token, b and S increase and decrease at the same rate at the time of rebase, so the invariant of vAMM of type $xy = k$ can be written as follows. LP fees are ignored.

$$b \cdot q = S \cdot k \tag{5}$$

From eq. 5, the relationship between total supply, price, and reserve is as follows. It can be seen that the reserve amount is determined only from the total supply S and the price p.

$$\begin{aligned} p &= \frac{q}{b} && (6)\\ b &= \sqrt{\frac{k \cdot S}{p}} && (7)\\ q &= \sqrt{k \cdot S \cdot p} && (8) \end{aligned}$$

B Liquidation free region

We explain the derivation of the liquidation free region introduced in Section 3.3. Suppose the trader has a combined position of the taker position of leverage u and the LP of leverage v. u and v are arbitrary real numbers and $v \geq 0$. When u is positive, it represents long, and when u is negative, it represents short. Suppose the trader creates a position at time $t = 0$. Let $S(t)$ be the total supply of base tokens at time t, and $p(t)$ be the price determined by vAMM. The trader's collateral is $p(0)$. Using eq. (7) and eq. (8), the trader's base token balance $b(t)$ and quote token balance $q(t)$ are as follows. LP fees are ignored.

$$\begin{aligned} b(t) &= (u - v) \cdot \frac{S(t)}{S(0)} + v \cdot \sqrt{\frac{p(0)}{p(t)}} \sqrt{\frac{S(t)}{S(0)}} && (9)\\ q(t) &= -u \cdot p(0) - v \cdot p(0) + v \cdot \sqrt{p(t)p(0)} \sqrt{\frac{S(t)}{S(0)}} && (10) \end{aligned}$$

Assuming that unrealized PnL is $R(t)$, total account value is TotalAccountValue(t), total position notional is TotalPositionNotional(t), and margin fraction is MarginFraction(t) at time t, these can be written as follows.

$$\begin{aligned} R(t) &= b(t) \cdot p(t) + q(t) && (11)\\ \text{TotalAccountValue}(t) &= p(0) + R(t) \\ &= p(0) + b(t) \cdot p(t) + q(t) && (12)\\ \text{TotalPositionNotional}(t) &= |b(t) \cdot p(t)| && (13)\\ \text{MarginFraction}(t) &= \frac{\text{TotalAccountValue(t)}}{\text{TotalPositionNotional(t)}} \\ &= \frac{p(0) + b(t) \cdot p(t) + q(t)}{|b(t) \cdot p(t)|} && (14) \end{aligned}$$

If we define $z = b(t) \cdot p(t)$, $w = p(0) + q(t)$, MarginFraction(t) can be written simply as follows.

$$\begin{aligned} \text{MarginFraction}(t) &= \frac{z(t) + w(t)}{|z(t)|} \\ &= \text{sign}\left(z(t)\right) + \frac{w(t)}{|z(t)|} \end{aligned} \tag{15}$$

Inserting eq. (9), eq. (10) into $z(t)$, $w(t)$ gives the following.

$$\begin{aligned} z(t) &= b(t) \cdot p(t) \\ &= \left((u - v) \cdot \frac{S(t)}{S(0)} + v \cdot \sqrt{\frac{p(0)}{p(t)}} \sqrt{\frac{S(t)}{S(0)}} \right) \cdot p(t) \qquad (16) \\ w(t) &= p(0) + q(t) \\ &= (1 - u - v) \cdot p(0) + v \cdot \sqrt{p(t)p(0)} \sqrt{\frac{S(t)}{S(0)}} \qquad (17) \end{aligned}$$

If MarginFraction(t) is greater than or equal to the maintenance margin rate (any constant $\in (0, 1)$) for any p, S, then the position is liquidation free. Consider the cases separately by $z(t)$.

When $z(t) = 0$, there is no position, so obviously liquidation free.

When $z(t) > 0$, MarginFraction(t) is as follows. When $w(t) \geq 0$, MarginFraction$(t) \geq 1$ always holds, so liquidation free. When $w(t) < 0$, MarginFraction(t) can be made arbitrarily small by making $S(t)$ small for any $p(t)$, so it is not liquidation free.

$$\text{MarginFraction}(t) = 1 + \frac{w(t)}{z(t)} \tag{18}$$

When $z(t) < 0$, MarginFraction(t) is as follows. MarginFraction(t) can be arbitrarily approached to -1 by increasing $S(t)$ for any $p(t)$, so it is not liquidation free.

$$\text{MarginFraction}(t) = -1 - \frac{w(t)}{z(t)} \tag{19}$$

Therefore, it becomes as follows.

$$\text{Liquidation free} \Leftrightarrow \forall p, \forall S, z(t) \geq 0 \text{ and } w(t) \geq 0 \tag{20}$$

From eq. (16) and eq. (17), the following holds.

$$\forall p, \forall S, z(t) \geq 0 \quad \Leftrightarrow \quad u - v \geq 0 \tag{21}$$

$$\forall p, \forall S, w(t) \geq 0 \quad \Leftrightarrow \quad u + v \leq 1 \tag{22}$$

Therefore, the liquidation free region is as follows.

$$\text{Liquidation free} \Leftrightarrow u - v \geq 0 \text{ and } u + v \leq 1 \tag{23}$$

C Token value

This section describes how to derive the value of the ERC4626 token introduced in Section 3.4. LP fees are ignored. First, consider the value of Long tokens. Since this is a tokenized version of the 1x long position, substitute $u = 1, v = 0$ for eq. (9), eq. (10).

$$b(t) = \frac{S(t)}{S(0)} \tag{24}$$

$$q(t) = -p(0) \tag{25}$$

Since the total account value represents the token value, substituting eq. (24), eq. (25) into eq. (12) gives:

$$\text{TotalAccountValue}(t) = p(t) \cdot \frac{S(t)}{S(0)} \tag{26}$$

Next, consider the value of LP tokens. Since this is a tokenized composite position of 0.5x long and 0.5x LP, substitute $u = 0.5, v = 0.5$ for eq. (9), eq. (10).

$$b(t) = 0.5 \cdot \sqrt{\frac{p(0)}{p(t)}} \sqrt{\frac{S(t)}{S(0)}} \tag{27}$$

$$q(t) = -p(0) + 0.5 \cdot \sqrt{p(t)p(0)} \sqrt{\frac{S(t)}{S(0)}} \tag{28}$$

Since the total account value represents the token value, substituting eq. (27), eq. (28) into eq. (12) gives:

$$\text{TotalAccountValue}(t) = \sqrt{p(t)p(0)} \sqrt{\frac{S(t)}{S(0)}} \tag{29}$$

From eq. (26), eq. (29), if $S(t)$ does not change, the value of the Long token tracks $p(t)$ and the value of the LP token tracks $\sqrt{p(t)p(0)}$. The value of both tokens is uniquely determined when $S(t) \cdot p(t)$ is determined. The token value when $S(t)$ does not change is plotted in Figure 5.

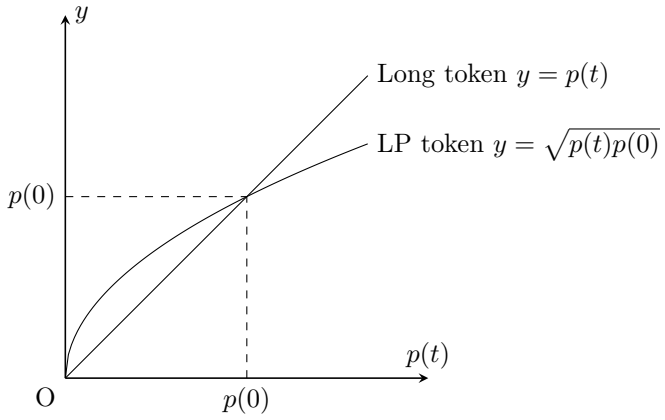

Figure 5: Token value without funding

D Liquidation price arbitrage attack

We describe an attack we call the liquidation price arbitrage attack. This is an attack that occurs when the price used for liquidation judgment is different from the price used for opening / closing a position or calculating profit and loss. We will call the former price for liquidation (p_{liq}) and the latter price for PnL (p_{pnl}). Here are two problematic cases. The first is the problem of opening infinite positions. Suppose $p_{\text{pnl}} < p_{\text{liq}}$. If the divergence between p_{pnl} and p_{liq} is large enough, when you open a long position, you will be profitable on the p_{liq} basis. This profit makes the liquidation judgment false, so you can open an infinite number of positions. The second problem is that bad debt is likely to occur during sudden price changes. For example, Perpetual Protocol v2 uses TWAP for liquidation determination, so p_{liq} will be delayed relative to p_{pnl} if the price changes suddenly. This causes the problem of delayed liquidation and increased bad debt. The difference between p_{liq} and p_{pnl} in perpetual futures can be seen as putting the trader, which should be liquidated, on hold. This puts the insurance fund at the risk that the trader should take. In many cases, the reason why p_{liq} and p_{pnl} are different is to make p_{liq} a more stable price than p_{pnl} so that it will not be liquidated by sudden changes in p_{pnl}. It's a kind of investor protection and has some benefits, but it should be done with caution from a security perspective.