Deri Protocol V3: The Evolution of Capital Efficiency

0xAlpha
@Deri Protocol

Richard Chen
@Deri Protocol

Daniel Fang
@Deri Protocol

ARTICLE HISTORY
Compiled January 3, 2022

Abstract
This paper introduces Deri Protocol V3, a decentralized derivative protocol upgraded from Deri Protocol V2. Inheriting all the features of V2, Deri V3 extends the supported derivative universe to all funding-fee-based perpetual derivatives and has them handled consistently. One of the key features of Deri V3 is "external custody", i.e. the user capital is held in an external money market protocol instead of internal liquidity pools. This new mechanism supports multiple base tokens with substantially higher scalability and capital efficiency.

Contents

1. Introduction

In February 2021, we introduced Deri Protocol V1[*1*], a decentralized protocol for users to exchange risk exposures precisely and capital-efficiently. Ever since its launch, Deri protocol has become the DeFi way to trade derivatives: hedge risks, speculate or arbitrage, all on-chain. This is achieved through the so-called Automatic Market Making (hereafter denoted as AMM) mechanism.

Deri's AMM mechanism (specifically denoted as Deri PMM, or DPMM) works as follows: on one side, liquidity providers (hereafter denoted as LP) provide liquidity to the pool to collectively play the role of the counterparty for the traders; while on the other side, traders deposit margin to open long or short positions of futures/options. Profits and losses (hereafter denoted as PnL) are calculated according to the underlying asset prices and settled between the LPs (all together) and the traders (individually). While LPs and traders play different roles in the DPMM mechanism, in this whitepaper, we use the general term "*users*" as their collective name and "*user capital*" as the collective name for the liquidity provided by LPs or the margin deposited by traders. That is, wherever the term "*user capital*" is used, it indicates the liquidity provided by LPs and the margin deposited by traders are treated indifferently in that context.

In order to provide a higher capital efficiency for the user capital, several optimizations were introduced in Deri V2[*2*]:

(1) Accepts margin deposited by traders in one or more of the supported base tokens;
(2) Accepts liquidity contributed by LPs in one or more of the supported base tokens;
(3) Supports multiple trading symbols within one single trading pool.

With the terminology "user capital" defined above, the first two features can be consolidated into one collective attribute: accepting user capital in multiple base tokens. With these defining features of Deri V2, users can enjoy capital efficiency higher than most of the other existing derivative trading exchanges or platforms, including centralized exchanges. Such capital efficiency is achieved primarily thanks to the interactability and composability of the DeFi applications: a DeFi application can interact with another one on the same blockchain network by calling the latter's (public or external) functions just like calling its own internal functions.

As of Deri V1 and V2, the user capital is held in Deri's own liquidity pools. Within the DPMM framework, both the liquidity provided by LPs and the margin deposited by traders are essentially used as collateral for the trading business. That is, user capital is just held there and remains untouched until there is some PnL or fee settlement, upon which some PnL or fee will be added onto or subtracted from the collateral balance.

Since the launch of V1, Deri Protocol has been serving derivative trading smoothly and adequately for almost one year and processed a total trading volume of over 10 billion USD[*4*]. It has proven the effectiveness and efficiency of its mechanism. Nevertheless, it has not yet achieved the optimal capital efficiency in the DeFi world (maybe it never will be - there would always be a space for evolution). Among its major bottlenecks, Deri V2 still faces scalability issues in supporting multiple base tokens for user capital.

Therefore, once again we are upgrading Deri protocol, from V2 to V3.

And once again, we are leveraging the composability of the DeFi applications. This time we are introducing the so-called *External Custody* mechanism to tackle the scalability issue of supporting multiple base tokens.

2. Liquidity Custody

The AMM mechanism (DPMM as one example) is generally regarded as "non-custodial", in contrast to the custodial mechanism of centralized exchanges. However, the "non-custodial" nature is really referring to the fact that no person or legal entity is playing the role of "custodian". The capital still has to be stored somewhere, e.g. the liquidity pool of the AMM. In that sense, the liquidity pools (i.e. the smart contracts) are the *de-facto* custodians. Most DeFi applications (hereafter denoted as dapp) deploy their own liquidity pools as the "custodians". This is also the case for Deri V1 and V2.

The core functionalities of the Deri V2 custodians (i.e. the liquidity pools) can be summarized as follows:

- Holds and secures user capital in one or several base tokens;
- Calculates a *total dynamic effective value* of the user capital deposited.

Here the attribute "dynamic" refers to the fact that some of the tokens (i.e. the non-stablecoins) have constantly changing dollar values, and the attribute "effective" refers to the fact that these values might be discounted for total value calculation. In this calculation, the base tokens are not discounted equally. Instead, they are discounted per their respective liquidity[1].

2.1. *Internal V.S. External Custody*

Just as in the traditional finance, custody could be *internal* or *external* in the DeFi world. Here we define "*internal*" custody as a dapp's own liquidity pools being used

[1] Please note this word "liquidity" refers to the measure of how liquid the token is.

to hold capital, whereas "*external*" custody as adopting liquidity pools of some other dapps. For Deri Protocol, theoretically the custodial mechanism could be implemented either way.

In practice, the choice between internal or external "custody" depends on many factors. When it comes to Deri Protocol, it boils down to the question of whether there is a readily available infrastructure to provide such functionalities summarized above. And the answer is yes! The money market protocols, e.g. Compound[*5*], AAVE[*6*] or Venus[*7*], have already been one of the most mature sectors of DeFi and could provide the needed functionalities very well. Simply speaking, the "custody" could be implemented as: the user capital, upon deposit, will be stored into a money market protocol. The capital could consist of several kinds of base tokens, as long as accepted by the protocol. Moreover, the money market protocol will calculate the *total dynamic effective value* of the capital consisting of several kinds of tokens, usually defined as the "borrow limit" measure per the protocol. The details of the implementation will be explained in the later sections.

In the DeFi practice, adopting external custody has not yet been popular. One possible reason is that some projects tend to hold the capital internally due to their ego or distrust of other projects. However, this might go against the philosophy of DeFi. The choice of custodial (or any other) mechanism should be based on rationale, i.e. economic or technical considerations, rather than the developer's ego of holding capital internally. That being said, adopting external custody does introduce an additional technical risk: the project will then be highly dependent on the security of the "custodian" - you fail if they fail. However, this is nothing new but a classic software engineering problem - leveraging external libraries would introduce technical risks. As a matter of fact, the modern software engineering has already answered that question. The practice of leveraging external libraries has been proved to be safer in general - you just need to choose the libraries carefully. When it comes to the choice of external custodians, you need to select the ones proved by time and scale. These are usually the leading money market or lending protocols on their respective blockchain networks, such as Compound[*5*] or AAVE[*6*] on Ethereum and Venus[*7*] on Binance Smart Chain.

2.2. *Capital Efficiency Revisited*

Improving capital efficiency has been one of the primary missions of Deri Protocol ever since V1. The whitepaper of Deri V2[*2*] points out that one of the keys to improving capital efficiency is to make use of the non-cash assets as running capital. In traditional finance, running capital is typically limited to cash or cash-equivalent. When someone (e.g. a company) only having non-cash assets needs running capital, the common practice is to borrow cash (to be used as running capital) against the non-cash assets as collateral. In the crypto world, the counterpart to cash or cash-equivalent is usually stablecoin. If the cash as an intermediate step could be skipped, the capital efficiency would be substantially optimized. In Deri V2, we introduced the support for multiple base tokens: a user can deposit user capital in one or more of the supported base tokens, including non-cash assets (i.e. non-stablecoin tokens). This is feasible thanks to the interactability and composability of DeFi applications - any token is just one swap away from the traditional “cash” or “cash-equivalent” tokens, e.g. USDC or DAI, given there is a spot AMM supporting the swap.

Ever since the launch of Deri V2, it has demonstrated a very successful capital efficiency improvement. Traders have been trading with non-cash assets (e.g. BNB or CAKE) as collateral, while LPs have provided non-cash assets as liquidity contributions. The non-cash assets have become a significant source of user capital on Deri Protocol. However, the way Deri V2 supports multiple base tokens is very “manual” and thus not entirely scalable. For the trading business to scale better, the support of multiple base tokens should be more systematic. Considering base-token-supporting is within the scope of liquidity custody, we need a more advanced custodial mechanism. This is the key issue that Deri V3 is to tackle.

Again we will leverage the composability of DeFi - most of the dapps have been built to be composable like lego blocks. Therefore, there is no need to build everything from scratch. When it comes to Deri’s need for a custodial mechanism, the money market protocols are a natural choice. That is, instead of constructing our own liquidity pools to hold user capital, Deri V3 adopts a money market protocol for this purpose. A money market protocol does not just hold capital but also calculates its dynamic effective value. When some user capital is deposited in multiple kinds of tokens, we need to know the real-time total dynamic effective value of the tokens - this logic is already implemented in the *Borrow Limit* calculation of the money market protocol.

Even though not a required property of the custody, there is one additional benefit of adopting money market protocol as an external custodian: the capital held by it would generate some additional yield - the interests and maybe also the protocol’s liquidity mining rewards.

3. Architectural Changes

3.1. *Vault*

The most important change from Deri V2 to V3 is that the protocol will hold the user capital in a money market protocol, except for a portion of the Base0 tokens[2] held as the liquidity reserve. This is implemented by introducing a smart contract called *Vault*. Whenever a user deposits some user capital (i.e. an LP provides liquidity or a trader deposits margin) for the first time, Deri Protocol will deploy a Vault for this user. The vault will send the user capital to the money market protocol (minus the reserved portion of B0 tokens) and hold the receipt token on behalf of the user.

With the capital deposited, the external custodian, i.e. the money market protocol, calculates a real-time *Borrow Limit* for the capital. The protocol automatically handles the cases of user capital consisting of multiple tokens with different discounting factors. This *Borrow Limit* will be converted into the real-time dynamic effective balance of the user capital, which could either be the margin deposited by a trader or the liquidity contribution by an LP.

[2] Base0 token is the primary base token in which Pnl and fees are settled. Please refer to [*2*] for more details.

With this architectural change, the multi-base-token mechanism of Deri V3 will leverage the infrastructure of the money market protocol. In general, any token accepted by the money market protocol as collateral would automatically become a supported base token of Deri Protocol V3. This would significantly improve the scalability of the base-token-supporting mechanism.

3.2. *A uniform DPMM for funding-fee-based perpetual derivatives*

As pointed out in [*3*], perpetual futures and everlasting options are just two instances of the general form of funding-fee-based perpetual derivatives requiring one long position to pay one short position $[MARK - I(S)]$ as funding fee, where $I(S)$ is a general "intrinsic value" function of the underlier price S. Another instance of such funding-fee-based perpetual derivatives is the so-called *Power Perpetuals*[*8*]. The DPMM of Deri V3 has unified the funding and pricing mechanisms for all funding-fee-based perpetual derivatives. That is, the DPMM of Deri V3 is designed to universally handle funding-fee-based perpetual derivatives rather than respectively handling perpetual futures or everlasting options. This has led to a major architectural simplification of the Deri V3 DPMM: the DPMM of Deri V3 has only one general-purpose trading pool (i.e. one single smart contract) implementing the pricing and funding fee logic. In contrast, for Deri V2 we had to implement different smart contracts of trading pools for perpetual futures and everlasting options, respectively.

Furthermore, this is not just a technical upgrade but also leads to some financial engineering optimizations. Thanks to this architectural simplification, one trading pool of Deri V3 can simultaneously support different derivative types. Consequently, the trading pools do not have to be organized by derivative types. The removal of this constraint opens up more possibilities for capital efficiency optimization. For example, it is possible to trade derivatives of different types (e.g. futures and options) with the same underlying asset against one single liquidity pool so that traders can manage the Greek letters of that underlying asset in one place. This would make both the speculations and risk management much easier.

In practice, however, we have to be very careful with mixing different derivative types within one single trading pool - we need to strike a balance between capital efficiency and risk isolation. The optimization of capital efficiency under the constraints of risk control and isolation is still a field to be comprehensively investigated. The sophisticated interactions between DeFi projects have further complicated this matter. We will explore this area within a very safe risk management measure. Whenever it involves capital safety, we will be conservative rather than aggressive.

Experiments with innovative derivative types would also be much easier, simply because we no longer have to deploy new liquidity pools and specifically prepare the liquidity for new derivative types. Instead, new derivative types could be plugged into existing (general-purpose) trading pools, and then people can start to trade them immediately. In Deri V2, we deployed a trading pool called *"Inno Zone"* to experiment on perpetual futures with innovative underlying assets. However, the keyword *"Inno"* was referring to underlying assets but had nothing to do with derivative types. Whereas in Deri V3, the keyword *"Inno"* of the *Inno Zone* will extend to cover the derivative type dimension too. Derivative innovations will thus be made much easier.

3.3. *Greek-based margin requirements for everlasting options*

Designing a proper margin requirement algorithm for nonlinear derivatives such as everlasting options is not easy. Traditionally, most margin requirement algorithms are essentially scenario-based: the required maintenance margin is to cover the loss of the position associated with a specific risk scenario, e.g. underlying price changing by 5%. This is straightforward for futures (or any "delta-one" derivatives). However, it is more complicated for nonlinear derivatives such as everlasting options, since the margin ratio is usually a variable depending on the underlying price.[3] We want two properties to be satisfied by the margin requirement algorithm:

(1) The required margin should be able to cover the loss of the position due to an unfavorable underlying price change (e.g. 5%)
(2) When the underlying price changes in a direction favorable to a position, if the margin requirement is to be increased, the increment should not be greater than the profit increment. Otherwise, there could be a small chance of an unreasonable scenario: price changes favorably to a position, but it gets liquidated because the increment of the margin requirement has surpassed that of the profit.

When everlasting options were rolled out in Deri V2, we adopted a margin algorithm that applies a discount for out-of-money options, per the out-of-money ratio[*3*]. While this algorithm holds the first property very well, it fails to hold the second one. Therefore, we are upgrading to a Greek-based margin system for the everlasting options in Deri V3. With Δ and Γ, we can estimate the change of the option value due to the underlying price change from S to $S + \delta S$, with a second-order Taylor expansion:

$$\delta C \approx \Delta_{call} \delta S + \frac{1}{2} \Gamma \delta S^2 = \begin{cases} \delta S + V \left[-\frac{1}{2}(u-1)\frac{\delta S}{S} + \frac{1}{8}(u^2-1)\left(\frac{\delta S}{S}\right)^2 \right], & \text{if } S > K \\ V \left[\frac{1}{2}(u+1)\frac{\delta S}{S} + \frac{1}{8}(u^2-1)\left(\frac{\delta S}{S}\right)^2 \right], & \text{if } S < K \end{cases}$$

$$\delta P \approx \Delta_{put} \delta S + \frac{1}{2} \Gamma \delta S^2 = \begin{cases} V \left[-\frac{1}{2}(u-1)\frac{\delta S}{S} + \frac{1}{8}(u^2-1)\left(\frac{\delta S}{S}\right)^2 \right], & \text{if } S > K \\ -\delta S + V \left[\frac{1}{2}(u+1)\frac{\delta S}{S} + \frac{1}{8}(u^2-1)\left(\frac{\delta S}{S}\right)^2 \right], & \text{if } S < K \end{cases}$$

where $u = \sqrt{1 + \frac{8}{\sigma^2 T}}$.

And the margin system works as follows: the trader is required to post δC (or δP) as margin for a call (or put) position, with respect to a specific risk scenario, e.g. $\delta S / S = 5\%$. Please note that the ATM point needs to be handled specially since the function is not second-order continuous at this point.

[3] This issue does not exist for classical options, for which sellers simply post margins to cover the option values.

4. Implementation

4.1. *The Architecture*

The chart below illustrates the architecture of the DPMM of Deri Protocol V3. Please note that perpetual futures and everlasting options are traded within one single DPMM. However, we would like to emphasize that this is for theoretical illustration only. In practice, the DPMM configuration of derivative-supporting will be based on several considerations, with risk management on priority.

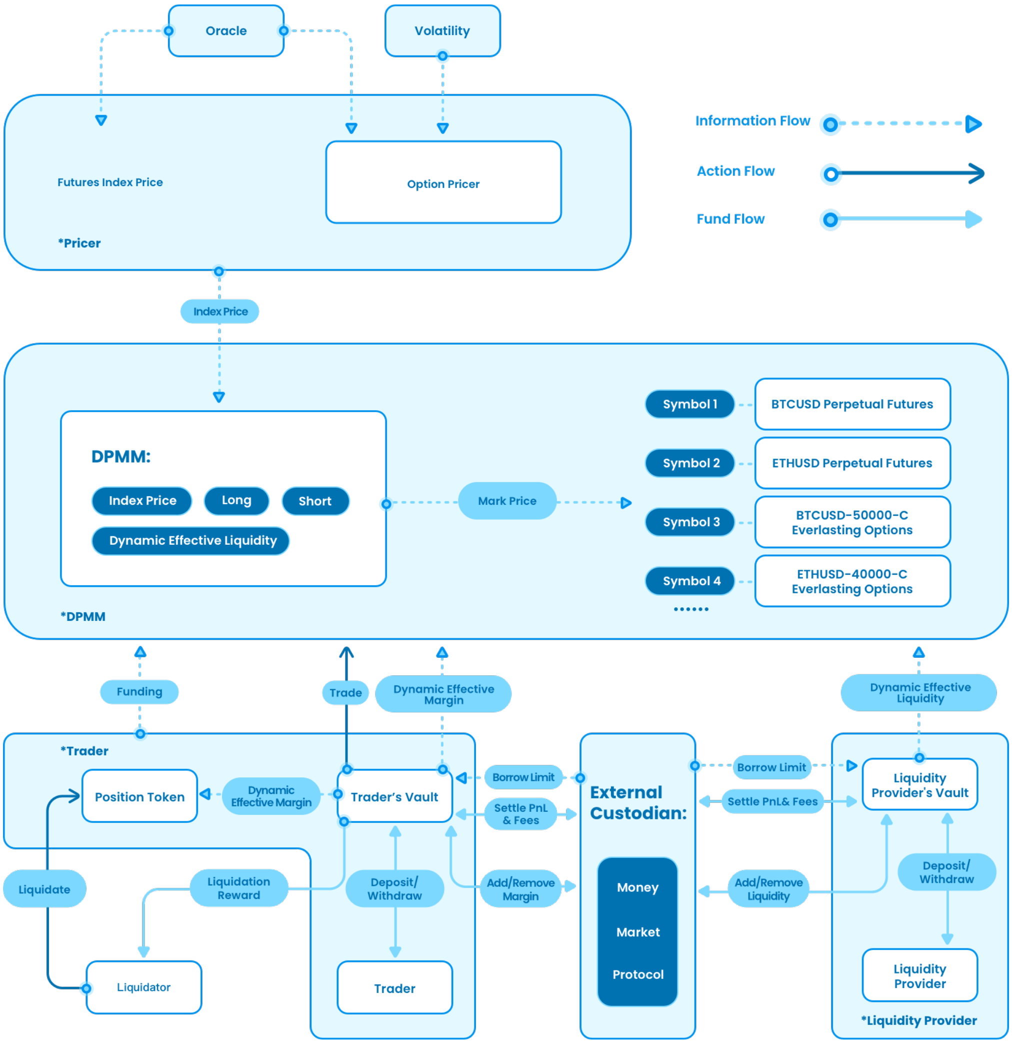

Figure 1.: Architecture of Deri Protocol V3

5. Summary

Since its launch, Deri Protocol has gone through two major version iterations and has been supporting two derivative types: perpetual futures and everlasting options. As of today, it has been deployed on multiple blockchain networks to serve traders' hedging and speculating demands and processed a total trading volume of over 10 billion USD[*4*]. Statistics show that Deri Protocol has become one of the most used DeFi derivative protocols per trading volume.

However, we never stop exploring the edge of providing an even more effective and efficient solution for people to exchange risk exposures on blockchain networks. That is why we are rolling out Deri Protocol V3. With the introduction of "external custody", Deri Protocol V3 supports multiple base tokens with substantially higher scalability and capital efficiency. Also, the DPMM of Deri V3 universally supports the funding-fee-based perpetual derivatives. Consequently, it can more flexibly organize liquidity for derivative trading across different types. Derivative innovations are thus made much easier under this framework.

Deri Protocol V3 is a defining project of DeFi 2.0 that will bring the "lego gameplay" of DeFi projects to a new level.

Acknowledgements

We would like to acknowledge BSC Core Tech Team and Leo Cheng for the constructive discussions.

References

(1) *"The Derivative Exchange Protocol"*.
https://github.com/deri-finance/whitepaper/blob/master/deri_whitepaper.pdf
(2) *"Deri V2: The Derivative Exchange Protocol with Extreme Capital Efficiency"*.
https://github.com/deri-finance/whitepaper/blob/master/deri_v2_whitepaper.pdf
(3) *"The Exchange Protocol of Everlasting Options"*.
https://github.com/deri-finance/whitepaper/blob/master/deri_everlasting_options_whitepaper.pdf
(4) https://info.deri.finance/
(5) Leshner, Robert; Hayes, Geoffrey (2019). *"Compound: The Money Market Protocol"*.
https://compound.finance/documents/Compound.Whitepaper.pdf
(6) wow@aave.com (2020). *"AAVE Protocol Whitepaper V1.0"*.
https://github.com/aave/aave-protocol/blob/master/docs/Aave_Protocol_Whitepaper_v1_0.pdf
(7) Swipe Wallet (2020). *"Venus The Money Market & Synthetic Stablecoin Protocol"*.
https://venus.io/Whitepaper.pdf
(8) White, Dave; Robinson, Dan; Koticha, Zubin; Leone, Andrew; Gauba, Alexis; Krishnan, Aparna (2021). *"Power Perpetuals"*.
https://www.paradigm.xyz/2021/08/power-perpetuals/
(9) *"Perpetual Contracts Guide"*. https://www.bitmex.com/app/perpetualContractsGuide

Appendices

Appendix A. A Brief Review of DPMM

This section briefly reviews the DPMM mechanism of perpetual futures, which was already implemented as Deri V2.1 but not covered in the previous whitepaper.

When Deri V2 was first launched, the mechanism of perpetual futures trading was implemented as a simple AMM but was later upgraded to the DPMM mechanism in Deri V2.1. This upgrade was to adopt the better-known and easier-to-understand funding mechanism, i.e. the funding based on the spread of $(mark - index)$, which is in line with most centralized derivative exchanges.

The simple AMM of Deri V2 for perpetual futures trading works as follows: whenever there is a trade, the AMM takes the index price from the oracle as the trading price. That is, there is no slippage from the index price for the trade. Whereas in Deri V2.1, we switch to DPMM to process trades of perpetual futures. With DPMM, when the net position is 0 (i.e. the equilibrium state), the mark price equals the index price (fed by the oracle). Whenever there is a trade, it pushes the mark price toward the trading direction (i.e. a buying trade pushes the price up while a selling pushes it down). The price change due to the trade is proportional to the trade size. For example, if the current mark price is P and someone places a trade of size x, then the mark price is pushed to $P + \Delta P$, where $\Delta P = a \cdot x$ with a determined by the pool liquidity and the pool parameters. Since mark price is the trading price for a trade of infinitesimal size as of the current state (similar to the mid price of an orderbook), the trading price of the trade of size x is the average from P to $P + \Delta P$, roughly $P + \Delta P/2$. The exact trading price is calculated by the trading cost, an integral from P to $P + \Delta P$, divided by the trading volume.

As the trading volume pushes the mark price linearly, the price spread and the mark price are determined by the total net position, as follows:

$$\begin{aligned}&\Delta P/i = a(l - s),\\ &P = i + \Delta P = i[1 + a(l - s)]\end{aligned}$$

where i is the index price, l and s are the total long and short positions (thus $(l - s)$ is the total net position), and a is a coefficient determined by the pool liquidity and parameters.

Funding Fee

Within the DPMM algorithm, the funding fee mechanism has been switched to the better-known funding fee based on the spread of the mark price over the index price, i.e. $(mark - index)$. Every second, one long position pays one short position funding fee as follows:

$$F = f \cdot (P - i)$$

where P is the mark price, i is the index price, and f is the funding fee coefficient. This is the same funding fee mechanism as that of most centralized derivative exchanges, e.g. BitMEX[9].

Please note that the DPMM funding mechanism is mathematically equivalent to that of the simple AMM of Deri V2. In Deri V2, we have the funding fee of 1 long contract per second proportional to the net position $(l - s)$:

$$F_{SimpleAMM} = [b \cdot (l - s)]i$$

where b is the funding fee coefficient of V2 and i is the index price but also the contract value of one natural unit of contract (e.g. the USD-value of 1BTC). Whereas with DPMM, we have

$$\begin{aligned} F_{DPMM} &= f \cdot [i(1 + a(l - s)) - i] \\ &= [fa \cdot (l - s)]i \end{aligned}$$

It is easy to see that, as long as $fa = b$, the two funding fee mechanisms are equivalent. In fact, the simple AMM of Deri V2 can be thought of as a special DPMM with $a \to 0$ while fa is kept as a finite constant (the coefficient b).

Forced Liquidation

Please note that Deri Protocol adopts an account-level cross-margin framework for the margin requirement. That is, a total required margin is calculated against all the open positions by one account (i.e. the position token) within a trading pool and used to compare with the dynamic effective balance of the account. Should the latter fail to fulfill the maintenance margin requirement, the whole account would be forcedly liquidated and thus all account balance would be lost. In other words, after a forced liquidation, the account balance will become 0. From the technical perspective, this means the corresponding position token will be reset to the initial state: all the positions and margin will be set to 0.

An account's dynamic effective balance contains all the unrealized PnLs of its open positions. During any unrealized PnL calculation for liquidation purposes, the unrealized PnL is calculated based on the index prices for perpetual futures or the theoretical prices for everlasting options. This is in line with the common practices of the centralized exchanges[9]. It is especially important to do so in the DeFi scenario as the mark price of DPMM is subject to flashloan manipulation. If the unrealized PnL were calculated based on the mark price for liquidation purposes, attackers would be able to forcedly liquidate healthy accounts by adopting flashloan.