DeltaDeFi

Revolutionize Permissionless Liquidity

Version 2

Mar 2025

1. Abstract

This whitepaper outlines the protocol tailored for high-frequency trading within the UTxO ecosystem. Leveraging the unique features of UTxO architecture, DeltaDeFi protocol solves the challenges in order management fees, trading slippage, and slow finality. With DeltaDeFi protocol's innovation in on-chain logic and off-chain order handling, high-frequency traders could benefit from pure efficiency enhancement while maintaining full control of the funds. With such architecture, DeltaDeFi aims not only to enrich UTxO's financial infrastructure but also to pave the wave for big migration from centralized exchange trading by closing the gap in user requirements and experience.

Table of Contents

2. Core Problem Statement - Inefficient Non-custodial High Frequency Trading

The intersection of decentralized finance (DeFi) and high-frequency trading (HFT) presents a unique set of challenges within the current trading ecosystem, particularly when leveraging the security and robustness of layer-1 blockchains. Centralized exchanges, while currently efficient, pose significant custodial risks. The success of pseudo-decentralized exchanges in other ecosystems highlight the need for efficient, non-custodial trading solutions in UTxO ecosystems.

2.1. The Problem

2.1.1. Centralized Exchange Dominance

A significant volume of trading activity is concentrated within centralized exchanges (CEXes), which operate on a custodial basis. This model, despite its widespread use, exposes traders to risks, including potential security breaches and mismanagement of funds, evidenced by collapses of platforms like FTX. The reliance on CEXes underscores the critical demand for secure, non-custodial alternatives that can offer comparable efficiency without compromising user safety.

2.1.2. Unserved UTxO Ecosystems by Pseudo-decentralized Exchanges

There are attempts to provide support to HFT in several decentralized exchanges. Most are deemed as “pseudo-decentralized” solutions as these architectures need to rely on or build on a fast blockchain, which is either a dedicated separated blockchain or a fast but less secure layer-1 blockchain.

These pseudo-decentralized exchanges trade-off some security or decentralization in pursuit of matching user-experience with centralized exchanges. While the market has proven users accept the trade-offs, UTxO ecosystems like Cardano are unserved, mostly due to differences in technology stacks. The native solution built for UTxO blockchains is yet to be developed.

2.1.3 DEX Product Gaps & Integration Complexities

Most existing decentralized exchanges (DEXes) in UTxO ecosystems aim to address the custodial risks associated with CEXes but introduce their own set of challenges, notably in order management and fee structures. The inherent design of most DEXes results in fees for actions that are typically cost-free in

CEXes, such as order creation, modification, and cancellation. This discrepancy not only affects trading efficiency but also widens the gap between the user experiences of CEXes and DEXes, hindering broader adoption among professional traders.

The integration of trading strategies and solutions with DEXes is further complicated by technical and operational challenges. These include variable and non-deterministic confirmation times, which necessitate a high degree of technical expertise and significantly increase the engineering costs associated with DEX integration. Such complexities deter the seamless transition of liquidity from CEXes to DEXes and inhibit the potential for innovation in decentralized trading platforms.

2.2. Objective

Addressing these challenges is imperative to unlocking the full potential of DeFi trading in UTxO ecosystems, facilitating the migration of liquidity from centralized to decentralized platforms, and ensuring a secure, efficient, and equitable trading environment. By overcoming the limitations of current non-custodial trading mechanisms and bridging the gap between CEXes and DEXes, there exists a substantial opportunity to enhance market liquidity, reduce bid-ask spreads, and foster a more inclusive and resilient DeFi ecosystem.

3. DeltaDeFi Solution - Hydra Order Book DEX

To solve the 3 problems described above, DeltaDeFi introduces the L2 DEX build with the Hydra Head protocol. The L2 DEX is a fully non-custodial platform where users interact with smart contracts to safeguard their funds in the process of trading. With L2 technology, we released all the throughput bottlenecks we faced in L1 and attained extremely low transaction fees, enabling CEX-like features and products which would be practically infeasible in L1.

3.1. Pure Trading Efficiency with Necessary Decentralization

The DeltaDeFi solution could be regarded as a design for pure efficiency. With the use of Hydra, we are able to release all blockchain fees happening on L1, making the order management costless - which is at par with CEXes and closes the UX gap. Although Hydra is fast, it is technically not fast as instant since extra networking time is still needed for hydra nodes in the same head to communicate. In order to achieve instant order confirmation, all order execution has to be approved by our system. In this way, we can make sure every order is consumed by exactly 1 counterparty, which eliminates the problem of UTxO contention in off-chain computation. Accordingly, DeltaDeFi's users are highly confident that their orders are being confirmed at the time of hitting API calls.

The pure trading efficiency features come with a scarification of users' power in interacting with the protocol - every order action has to be overseen by DeltaDeFi's system. That being said, DeltaDeFi is designed in a way where users still get full control of their funds, with or without support from DeltaDeFi's system. We would suggest users always conduct withdrawal and order cancellation through the system API to provide guaranteed order flows for other users. However, such actions related to fund autonomy can also be conducted without DeltaDeFi's permission. It is only suggested in any emergency scenarios such as service downtime, loss of keys for authentication, etc. If such a mechanism is abused in normal scenarios undermining other users' order finality, bad actors would be labeled with temporarily limited access to the protocol.

3.2. Adopt the Wisdom from Established Financial Markets

One of the key parts of DeltaDeFi protocol is to conduct order matching for users. Since order placement is separated from blockchain transactions, we can adopt the traditional wisdom from existing financial markets. On order priority, we would adopt the mainstream and battle-tested mechanism - price-time priority - the orders with the best prices would match first, when there are orders with the same prices, priority would be given to the earlier-placed orders. Such an order-matching mechanism could ensure the protocol fairness to every user.

An order matching event is equivalent to a commercial contract signing on buying and selling goods at an agreed price. DeltaDeFi aims to provide a similar commitment to users when orders are confirmed by DeltaDeFi's system but not from the blockchain confirmation. For all the aggressive goals, there is still a chance that orders pass the blockchain validation and the system API response returns a false positive. In this sense, DeltaDeFi protocol would maintain an insurance fund to compensate for such service dissatisfactory events. It is also another area that makes DeltaDeFi's service at par with CEXes.

3.3. Transparent Protocol Governance

The notation of "all transactions should be overseen by DeltaDeFi system" might sound like the protocol remains in huge control in order execution. Operationally this concern is legit. This is, nonetheless, mitigated by economically rational behavior, which DeltaDeFi allows and promotes the protocol governance by the market forces.

Also adopting the idea from established exchanges, in DeltaDeFi users can always request for their trading records and receive complete aggregated trading records for the market. The market data is in aggregated form to protect other traders' privacy since it is material information with the potential of being exploited. With such logs, the community can easily reconcile the protocol behavior, making sure the DeltaDeFi system does not abuse any power given. One example is that the DeltaDeFi system is technically possible to squeeze all potential slippage from market order (this is the slippage presented in traditional orderbook-based centralized exchange as well, not talking about the slippage introduced by AMM DEXes). However, with such trading records, the community can reveal suspicious patterns easily, and eventually leave the DeltaDeFi protocol for trading for such a reason. Given the rational behavior of every party involved, the DeltaDeFi team is incentivized to perform with integrity.

Additional transparency DeltaDeFi protocol would provide is open sourcing of critical source code, including all smart contracts and critical pieces of infrastructure which is necessary to provide confidence to the community. With all disclosure of information and records, a natural form of integrity governance could be formed to monitor the system's behavior.

4. Product Architecture

On product architecture, we have 3 major objectives - (1), architecturing the infinitely scaling Hydra DApp, (2) solving the UTxO contention issue and (3) maintaining the fairness of order processing. DeltaDeFi aims to build an infrastructure compatible with UTxO blockchains. Considering the technical nuances among different blockchains within UTxO ecosystems. In this session, we will describe how we designed the product on the Cardano blockchain with Hydra technology to achieve all 3 of the objectives.

4.1. Infinitely Scaling Hydra DApp

The most straight-forward way to run a Hydra DApp is to invite all users to become one of the Hydra node operators. This way can also ensure the protocol remains fully decentralized as every single participant obtains the power of rejecting snapshot updates for the Hydra instance. However, current Hydra Head protocol can only support standalone instances with very limited participants as direct node operators. Furthermore, expecting normal end users to run separate Hydra nodes on their own consume non-trivial resources that fades most DApps' economics away.

Thus, any Hydra DApp which desires mass adoption by the general community must maintain the nodes in a controlled set of operators, similar to the concept of App chain in other ecosystems. Under the line of logic, DeltaDeFi comes up with the below design choices to support infinitely scaling Hydra DApp, which makes DeltaDeFi technically ready for mass adoption.

4.1.1. Reimplement account-based model on UTxO blockchain

One nuance about implementing a user account on the UTxO blockchain is that concurrency cannot be supported. Usually, this is a bad design pattern on a typical Cardano DApp. However, given the Hydra environment for scalable DApp is usually in a controlled environment, this assumption can be released in a way the DApp operator can control exactly the transaction flow. DeltaDeFi's user account is built with this philosophy to simplify the on-chain state.

4.1.2. Aggregating information at committing and decommitting UTxOs

Given limited participants in Hydra, at the time of committing UTxOs we have to condense all the DApp information or state into a few UTxOs' datum. Considering DeltaDeFi might support up to tens of thousands of users, it is impossible to commit those UTxOs plainly as if we are building a L1 DApp.

Taking advantage of customizable protocol parameters in the Hydra environment, we can release execution units (ExUnits) limitation in Hydra and configure it to be nearly infinite. Given these released parameters, any DApps can pass through massive DApp states into Hydra by building condensed states in L1 and expanding the condensed information inside Hydra, which takes ExUnits normally exceeding L1 limitation.

In DeltaDeFi, we aggregate all DApp states inside Merkle Trees and only commit the Merkle root hashes into Hydra. Right after UTxOs are committed, we will trigger complex transactions to split the Merkle information back to entire DApp states as if we are running a L1 DApp. Meanwhile, everything can still be safeguarded by validators or smart contracts, making sure states are not lost.

Same logic applies to decommitting UTxOs from Hydra. With this approach, DeltaDeFi's contracts are forward looking as if they can support infinitely large user states.

4.2. Solving UTxO Contention

Building decentralized applications on the UTxO model must solve the UTxO contention issue - where there are possibilities of multiple parties attempting to consume the same UTxO. In that case, only one of those attempts would eventually succeed. Although it would not cause any loss of funds, leaving the UTxO contention issue unresolved would hugely undermine the user experience.

4.2.1 Off-chain order book engine governing order UTxO contention

All orders at DeltaDeFi will go through the off-chain order book engine, which is responsible for matching orders. The order book engine will serve as a single source of truth on match-making, and then making sure no order will be double-filled, to prevent order UTxO contention at its base level.

4.2.2. Single thread processing hydra transactions to prevent UTxO contention from simultaneous update

In scenario of the same order being (partially) filled multiple times, or same account balance

We introduced a smart contract account layer, where users have to deposit their funds into the “account” before trading on DeltaDeFi protocol. It could prevent users from innocently signing other transactions and voiding the throttled transactions.

4.3. Order Processing Fairness

While the VOB is powerful at providing pure efficiency to traders, DeltaDeFi indeed retains the discretion to process the orders. In the previous session, we covered the high-level mechanism to retain protocol governance. Here we cover the technical procedure where we could produce a fair order processing backend.

4.3.1. Single priority rule handling all orders

As aforementioned slightly, for each trading pair, we would have a single priority rule covering all order requests (post order, take orders, and cancel order). We would in no circumstances process orders deviated from the priority rule. If any unexpected issues are happening, they will be mitigated by our protocol governance procedures.

4.3.2. Single blocking thread processing same trading pairs

When handling different orders from the same trading pairs, the backend would have a single blocking thread processing orders. It could avoid unexpected overrides and retain the priority of orders strictly in compliance with the rule set out.

5. Decentralizing DeltaDeFi

DeltaDeFi comes with a philosophy of “product first”. At start we will focus on bringing the Cardano community the product that we lack but try not to be stuck by the decentralization problem. Right now, we have control over:

1. Hydra node operators
2. Immediate right of DApp state changes

In the future, when DeltaDeFi possibly grows in popularity and importance in the Cardano ecosystem and if we have sufficient resources, we will gradually release control over the 2 items.

5.1. Decentralizing Hydra Node Operators

Decentralizing Hydra node operators come with 2 steps.

First will be inviting reputable Cardano entities / organizations like founding entities to host some of the Hydra nodes. Given “at least one honest participants” principle, this alone will greatly increase the decentralization of the DeltaDeFi protocol.

Second will be converting or transforming the way we host Hydra to be more like a side chain, where we develop a custom consensus mechanism to decentralize the Hydra node set.

5.2. Decentralizing DApp Operation Right

Since the immediate right of changing DApp state is of paramount importance for DeltaDeFi operation and shipping the features that we promised. We expect the release of this right should come after the consensus mechanism being built for the DeltaDeFi Hydra node set, and then we will follow the same consensus from there for the DApp operation right. Any problematic

behaviour from the nodes will eventually cause some form of slashing and making sure the selected nodes are operating for the good of the DeltaDeFi community.

6. Conclusion

With the protocol design from DeltaDeFi, we can unleash the potential of the UTxO blockchain to handle high-frequency trading efficiently without sacrificing the custodian of traders' digital assets. The design also makes the on-chain validation loosely coupled with a trading engine, which leaves the potential for future migration to layer-2 blockchain to further unleash the scalability.