‘Y’ Lumio is the Killer App: Low Latency Liquidswap AMM Kills Bad CEX

Pontem is on a mission to change the crypto industry... Not just with clickbait and low effort forks. Today is the dawn of crypto 2.0. With the advent of sub 100ms latency for DeFi, the industry will be revolutionized. If you are not on the right side of history, you might get ftXd.

With Lumio vertically integrated into Liquidswap and Pontem Wallet, we will deliver the best user experience in crypto across the entire tech stack: interface, smart contract, execution, and settlement. If Amazon did it, so can we... If we are right then soon Uniswap (+ uniswap wallet) and Backpack (+ backpack exchange) will also launch L2s as second movers. For now we have a competitive advantage.

To usher the world into crypto 2.0, Pontem is building Lumio to be vertically integrated with Liquidswap and Pontem Wallet. The killer app for crypto is a vertically integrated tech stack that can compete with centralized applications. When you can offer parity on latency, costs, Users only care about security. P2P self custodial DeFi is the killer app that has been staring us in the face. Lumio finally makes it a reality. Not your keys, not your crypto...

We've already shipped Liquidswap, the top DEX on the Aptos network. Now, we're working on Lumio, a modular altVM L2. Lumio on testnet today is built to handle a lot of transactions really quickly and supports the Move VM language. Our aim? To give people a DeFi experience that feels as good and fast as what they'd get on big-name exchanges like Coinbase or Binance, but all happening on the blockchain.

But, to get to why this is a big deal, we need to look back at how trading on the blockchain usually works and what we need to do to make it better. Most of the time, trading directly on the blockchain can be slow, expensive and clunky. We want to fix that.

With Lumio, we're building something that was never before possible in crypto: profitable on-chain market making. We're making sure users and institutions can

trade quickly and safely, without giving up control of their assets or compromising on security.

Why does speed matter? Just ask A16z and Columbia researchers: Faster chains = less profits for arbitragers... effectively killing toxic order flow AKA sniping AKA Loss Versus Rebalancing (LVR).

“

We define loss-versus-rebalancing, or LVR, as the gap between the rebalancing strategy’s performance, and the AMM LP’s performance. The intuition for this underperformance is related to the phenomenon of “sniping” in high-frequency trading settings. In the model of Budish et al. [2015], a market maker quotes prices to trade a risky asset. Whenever public information arrives causing the fair price of the risky asset to move, there is a “speed race” between the quoting market maker to cancel her order, and other traders to “snipe” the market maker’s stale quotes.

“

https://arxiv.org/pdf/2208.06046.pdf#:~:text=We define loss-versus-rebalancing,model of Budish et al.

“...arbitrageurs arrive to trade on the AMM at discrete times according to the arrivals of a Poisson process with rate $\lambda > 0$... the probability that an arbitrageur arrives and can make a profitable trade, i.e., the fraction of time that the mispricing process is outside the no-trade region in steady state, is given by

$$\mathsf{P}_{\mathsf{trade}} \triangleq \frac{1}{1+\underbrace{\sqrt{2\lambda}\gamma/\sigma}_{\triangleq \eta}}.$$

“

The crucial point is in 1/(1+ √(2λγ/σ)) in the equation for cumulative arbitrage profits, this term can be seen as the probability of a profitable trade, or Ptrade. As λ increases (which means interblock time decreases, or the chain is faster), the denominator of this term gets larger, and thus Ptrade gets smaller. This directly scales down the expected arbitrage profits, as the arbitrage profits in this context are a result of trades that occur during these blocks.

"

Our results also have the potential to better inform AMM design, and in particular, provide guidance around how to set trading fees in a competitive LP market, in order to balance LP fee income and LP loss due to arbitrageurs. Finally, the asymptotic regime analysis λ → ∞ above points to a significant potential mitigator of arbitrage profits: running a chain with lower mean interblock time Δt ≜ λ−1 (essentially, a faster chain), since we show that this effectively reduces arbitrage profit without negatively impacting LP fee income derived from noise trading.

"

https://arxiv.org/pdf/2305.14604.pdf

History of Irrational Markets: Information & Speed

In the realm of trading, the essence of success lies in possessing timely information and the agility to act swiftly in the market.

> The tale of Nathan Mayer Rothschild and his alleged financial maneuvers following the Battle of Waterloo serves as a classic example. He reportedly leveraged a rapid courier network to gain early intelligence on Napoleon's defeat, a move that, according to legend, allowed him to manipulate the bond market to his significant benefit. By selling his bonds to incite panic and then repurchasing them at a lower price before the broader public became aware of the victory, Rothschild is said to have amassed a considerable fortune. While this narrative is intriguing and widely cited, its factual accuracy remains a topic of historical debate.

This principle of leveraging information and speed has not only been relevant in the past but continues to hold true today. In the digital age, the need for physical couriers has been replaced by the instantaneous transmission of data via the internet. The key to success now hinges on accessing timely information and employing the fastest means to act upon it in the marketplace.

A modern illustration of this concept is the phenomenon of high-frequency trading (HFT). This was epitomized by an event in the early history of the internet, where a group of businessmen established a direct, dedicated internet connection between the Chicago Stock Exchange and the New York Stock Exchange. This setup was designed to arbitrage trade orders at unprecedented speeds, a strategy highlighted in the notable case of FlashBoys. This initiative underscored the critical advantage of speed in executing trades, proving once again that in the competitive arena of financial markets, information and the speed of response are indispensable elements for achieving success.

Modern Day and Blockchain Protocols

The emergence of blockchain technology and decentralized exchange (DEX) protocols has transformed the trading landscape, though the quest for optimal latency remains a constant across both centralized (CEX) and decentralized exchanges. Traders utilizing CEXs strive for the lowest possible latency, often relocating their trading infrastructure—such as bots and servers—closer to the CEX data centers. This strategic positioning is done to minimize ping and response times, thereby maximizing trading efficiency and profits.

The situation becomes more nuanced when dealing with decentralized protocols on blockchains like Ethereum and Solana. Here, we've seen the rise of Miner Extractable Value (MEV) bots. Initially, on Proof of Work (PoW) chains like Ethereum, traders could prioritize their transactions by paying higher fees, enticing miners to select their transactions for quicker processing. With the transition to Proof of Stake (PoS) systems, similar dynamics persist, but we also see innovations like Flashbots. These developments have essentially created a marketplace for validators and MEV bots, facilitating agreements to expedite transaction execution.

However, Ethereum's inherent characteristics, such as its 12-second block confirmation time and susceptibility to forks, introduce additional complexities. During the time it takes for a transaction to be confirmed on Ethereum, significant price movements can occur on CEXs, affecting the potential for on-chain actions and strategies. This discrepancy underscores the challenges and considerations unique to trading in decentralized environments, where the blockchain's technical limitations can impact the speed and efficiency of trade execution compared to traditional centralized platforms.

Ethereum's inherent traits, such as its 12-second block confirmation time and the potential for forks, present unique challenges. The delay in transaction confirmation can lead to missed opportunities due to significant price movements on centralized exchanges (CEXs). This gap highlights the unique difficulties of trading in decentralized settings, where blockchain's inherent constraints can affect the pace and efficiency of executing trades, in stark contrast to the more immediate transactions seen on traditional CEXs. Additionally, the possibility of block reorganizations (reorgs) on Ethereum adds another layer of complexity, necessitating a waiting period for transaction confirmations to ensure finality.

The Role of Layer 2 Solutions (L2s)

The introduction of Layer 2 solutions (L2s) marks a pivotal shift. Many popular L2s operate with a degree of centralization; for instance, their sequencers—responsible for block production—determine the order of transactions. In environments like Arbitrum or Optimism, the transaction pool (mempool) is not publicly accessible, effectively making it a private mempool. This setup can resemble trading on a CEX, but with notably higher latency, averaging around 2 seconds for Optimism and approximately 0.25 seconds for Arbitrum. Despite the seemingly centralized aspect of these L2s, they offer mechanisms for verification: Optimistic Rollups use a dispute resolution system, while ZK Rollups rely on zero-knowledge validity proofs for validation.

The evolution of L2 solutions is towards decentralized sequencing, akin to Proof of Stake (PoS) chains, where sequencer nodes would organize transactions. This would open up mempools to broader access, potentially through various methodologies—prioritizing higher fee transactions or implementing time-based solutions. However, it's important to recognize that such advancements in decentralization may not necessarily reduce latency. In fact, as decentralization typically introduces more nodes into the consensus process, latency might increase due to the added complexity and coordination required. Additionally, while L2s like Optimism are designed to minimize the likelihood of reorgs, they are not entirely immune, particularly if their underlying L1 (in this case, Ethereum) undergoes a reorg, although the probability remains low.

Towards a Solution

In navigating these technological landscapes, the goal is to enhance the efficiency and reliability of decentralized trading platforms. The ongoing developments in L2 technology and the exploration of decentralized sequencing represent significant steps forward. However, the challenge remains to balance the benefits of decentralization with the need for low latency and high throughput, essential for competitive trading environments. As the ecosystem evolves, the focus will be on creating solutions that not only mitigate the limitations of current blockchain infrastructures but also unlock new possibilities for traders and investors alike.

Project ‘Y’ Lumio (Y the F are we doing this?)

In the quest to replicate the seamless experience of centralized exchanges (CEXs) on the blockchain, Super Ultra Important Top Secret (SUITS) Project ‘Y’ Lumio identifies several critical attributes that are essential for success:

1. **High Transaction Throughput (TPS) and Frequency**: To match the real-time processing capabilities of CEXs, it's crucial to handle a large number of operations swiftly. This not only increases the volume of transactions but also enhances the overall efficiency of the blockchain, moving it closer to a "chain-less" or ‘Web2’ experience.
2. **Minimal Confirmation Time with No Forks**: The system should aim for rapid transaction confirmations. Ideally, the architecture should be designed in such a way that forks are either extremely unlikely or impossible, ensuring consistency and reliability.
3. **Time-Protected or Hidden Mempool**: Similar to the operational model of current CEXs, where the order book's mempool is not visible to all, starting with a temporarily hidden mempool could be a strategic move. This approach mimics the opaque nature of order handling in CEXs, providing a familiar trading environment for users.
4. **Low Transaction Costs**: Given the nature of high-frequency trading, where the number of transactions can be vast, keeping transaction fees low is imperative. Affordable costs will enable a higher volume of trades, facilitating more dynamic and accessible trading activities.
5. **Robust DeFi Platforms with Liquidity**: The ecosystem must include decentralized exchanges (DEXes), order books, credit markets, and other DeFi platforms that are rich in liquidity. This ensures that traders have a conducive environment for executing trades efficiently.

In light of these considerations, our research began with a focus on the Move VM, particularly its implementation in Aptos, given our expertise and ongoing development of Lumio. While we're open to exploring similar innovations on other platforms like Sui Move, Solana VM and WASM, our preliminary efforts have

concentrated on Aptos due to our familiarity and the existing integration of Move VM with Reth in Lumio, despite the slower speeds due to EVM dependencies.

Aptos has demonstrated impressive performance, notably achieving a throughput of 30-50k TPS, with current latency around 500-600ms. However, ongoing research indicates potential for even lower latency.

▼ **It's important to acknowledge the inherent trade-off between decentralization and latency; as a system becomes more decentralized, latency tends to increase due to factors like network delay and computational overhead.**

In our recent advancements with the Aptos node, we've laid crucial groundwork for ongoing modifications aimed at enhancing our Layer 2 (L2) framework. Notably, we've transitioned away from the conventional consensus mechanism to adopt a sequencer approach, which systematically processes transactions from the mempool, executes them, and documents the results. Concurrently, we've suspended Peer-to-Peer (P2P) connections between nodes and removed the logic for transaction distribution from the mempool, streamlining operations. Additionally, we've instituted a new protocol for generating and storing payloads—comprising transactions within a block and their execution hashes—in a dedicated database for future dispatch to L2, marking a significant step forward in our implementation strategy. Also, we separated the move executor from the state commit and mempool process. This significantly reduces the time the executor spends idle waiting for transactions and commits.

Latency measurements focus on intra-machine performance, setting aside network latency for later optimization. For minimized network delays, trading bots should ideally be located close to the sequencer with a strong connection.

Machine Configuration for Testing

Important notice: in the initial version of this article, drafted approximately a month ago, we commenced testing and enhancing the Aptos L2 node. After several iterations, we have observed significant improvements in performance for both the Aptos L2 and the Aptos L1 node, which operates in performance mode. These iterative developments have substantially enhanced the results for both implementations.

The configuration of the machine used for our testing, hosted on Alibaba, is detailed below. It is important to note that the resources were not fully utilized, potentially giving the impression of an overestimation of performance. This setup is suitable given that the node benefits from a dedicated virtual CPU (vCPU), which is crucial for the types of tests conducted. Additionally, full utilization of all CPUs and threads was unnecessary due to the nature of these tests and the workings of BlockSTM.

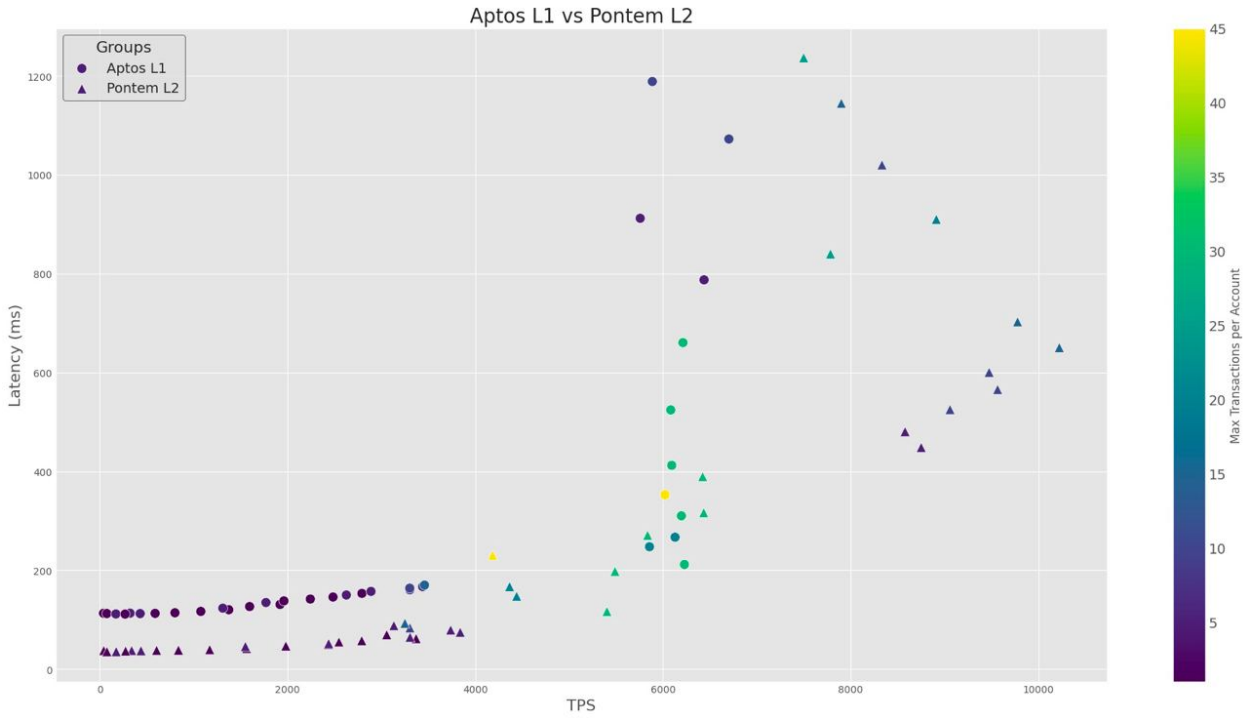
```
CPU model: Intel Xeon Platinum 8269CY
vCPU: 104 vCPUs
Memory: 192 GiB
Network speed: 25 Gbit/s
```

Tests

Understanding the testing of the Move VM and nodes is crucial, and it can be approached in two distinct ways. The first method involves testing the executor's performance in isolation—specifically, the Move VM without the node's framework. In this setup, the Move VM demonstrates impressive performance, achieving transaction processing speeds of 30-60k TPS.

However, in our case, the L2 node operates as a standalone unit, which requires the node to be functionally independent and configured for standalone operation. End-to-end (E2E) testing is paramount as it acts as the definitive verifier of our system’s integrity. Although similar to the operational dynamics of an L2 node, the standalone node primarily processes incoming transactions through Remote Procedure Call (RPC). This setup positions it as a reliable indicator of our system's functional validity.

Here are the outcomes we've achieved to date, comparing End-to-End (E2E) latency and Transactions Per Second (TPS) between Pontem L2 and Aptos L1:



Aptos L1 (avgTPS)	Aptos L1 (avg latency)	Y L2 TPS	Y L2 Latency
324.73	112.7822474	340.49	38.0151308
322.24	113.4225855	339.72	37.78911943
40.21	113.2068336	42.5	37.73170549
75	112.6462211	75.94	35.51855511
171.37	112.1151647	175.18	35.38937183
171.47	111.7859558	174.96	35.70135184
269.48	111.5564119	274.52	36.83501128
427.29	111.8990052	439.26	37.00017721
430.17	112.5849819	439.13	37.41426208
589.55	112.99136	605.56	38.06076824
800.74	114.1166969	‘Y’ TPS	‘Y’ Latency
1076.375	116.8895694	1171.05	39.3831024
1374.45	120.2639742	1564	42.00358201
1310.83	123.4447895	1551.18	46.05379337
1596.33	126.7394199	1982.39	46.95425457

1920.04	130.9295017	2433.19	50.21155589
1770.22	134.8432647	2438.78	51.77871886
1961.34	138.1130073	2548.19	55.03525867
2243.3	141.9378253	2791.03	57.6975725
2484.82	145.9951236	3370.43	61.94308402
2627.14	150.1177245	3305.8	64.69394276
2792.91	153.348241	3056.97	69.76650406
2889.65	157.3962666	3840.26	74.79112341
3303.28	160.6872816	3738.66	79.30452186
3302.41	163.8837588	3304.55	83.59468866
3440.05	167.1316598	3133.5	88.12825873
3459.98	170.1994858	3251.14	92.98498219
6228.92	211.8359028	5403.64	116.6583734
5856.68	247.7772146	4441.65	147.6826802
6129.43	267.068877	4366.27	167.2504724
6196.46	310.353633	5489.43	197.8644596
6021.55	352.908376	4186.14	230.7482736
6094.36	412.435392	5836.76	270.9519378
6083.55	524.4983587	6435.06	316.7386428
6213.45	660.6965393	6423.71	389.6177502
6438.22	787.6666227	8751.82	448.4752703
5757.89	912.2047152	8579.8	480.6939892
6703.04	1072.620142	9058.84	525.3115679
5887.35	1188.979575	9565.83	565.8513579
6785.5	3273.913484	9475.58	600.5738202
6821.16	3794.410318	10223.24	650.6854173
6291.44	3918.520945	9778.76	702.8101244
6024	4019.446784	7785.16	840.0816995
6284.15	4031.779174	8913.12	910.0888977

6750.15	4076.551359	8334.17	1020.218334
5840.08	4204.584271	7900.79	1144.900197
6337.6	4422.913965	7499.84	1236.887062

A more comprehensive table is available at this link.

The tests reveal an increase in the Transactions Per Second (TPS) of the Pontem L2 node, achieving three to four times reduced latency compared to the Aptos node: 3,000 TPS with latency under 100 milliseconds, which ranks among the best results in the market. Maximum in TPS achieved for Pontem L2 node is 11001 TPS. vs 7784 max TPS for Aptos L1. Moreover, these results offer the potential to significantly boost TPS to 20,000-30,000, as indicated by some of our tests, particularly with further optimizations and enhanced testing tools.

The tests were conducted using a transaction emitter for Aptos L1. For Pontem L2, we forked the transaction emitter to maximize its performance at this stage. We also benchmarked it using metrics obtained directly from the Pontem L2 node.

Next steps

The figures we've reported represent average metrics, rather than peak values, suggesting the possibility of achieving even lower latencies and higher Transaction Per Second (TPS) rates under optimal conditions.

This indicates significant potential for growth through targeted optimizations, aiming to elevate TPS rates and diminish latency.

Following changes can still be made:

- Implement specific native caches for the most used applications, such as Liquidswap. Many layers have only one or two frequently used apps. Keep them always in memory and asynchronously commit to the database.
- We have an idea to move the epilogue and prologue of transaction processing into Rust and make it native.
- We are looking into the possibility of improving the process of state transition between sessions of the executor.
- Our testing environment can be enhanced so we can conduct more varied tests with more interesting results.

- And more.

For those interested in conducting benchmarks on L2, given its closed-source nature, we invite you to contact us for further discussion.

Besides this, the next step is Pontem Lumio v2, stay tuned for ultra perfomance AltVM layer.