Aptos

Executive Summary

The following report goes over economics, transaction fees and transaction fee mechanism on Aptos. It covers only the fees collected/burned as Aptos is a layer 1 blockchain and there are no proofs nor data submitted to Ethereum.

Aptos transactions include standard information such as the signature, sender address, and payload, along with two additional fields: the sequence number and expiration time. The sequence number is an unsigned integer that must match the sender's account sequence number at the time of execution, functioning as the account nonce. The expiration time is a timestamp indicating when the transaction will no longer be valid.

Higher gas fees on Aptos prioritize transaction selection for the next block, but within a block, the order of transaction execution is determined by the system. This order is based on transaction shuffling, which enhances parallel execution efficiency by considering conflict patterns. Transaction shuffling spaces transactions from the same sender apart within a block as much as possible, reducing conflicts and the need for re-execution. Aptos claims that this sender-

aware shuffling can improve transactions per second (TPS) by 25%. Total burn = 250,000.00 APT

Impact on our design: Implementing transaction shuffling in our sharded blockchain can significantly improve parallel processing capabilities. By reducing conflicts and optimizing transaction order, we can increase throughput and achieve higher TPS, improving overall network performance. This needs to be researched further.

Introduction

Aptos is a Layer 1 Blockchain using Move language. Mainnet was launched on October 12, 2022. It uses APT token as its utility token for gas fees. The gas price is denominated in Octas. 1 APT = 10^8 Octas.

Goals & Methodology

The goal of this research is to get familiar with:

1. Economic specifics
2. Pricing model
3. Tokenomics

Results & Discussion

Economy specifics:

The initial total supply of Aptos tokens (APT) at mainnet was 1 billion tokens. The supply is divided as follows:

Category	% of Initial Token Distribution	Initial Tokens
Community	51.02%	510,217,359.767
Core Contributors	19.00%	190,000,000.000
Foundation	16.50%	165,000,000.000

Investors	13.48%	134,782,640.233

Source: https://aptosfoundation.org/currents/aptos-tokenomics-overview

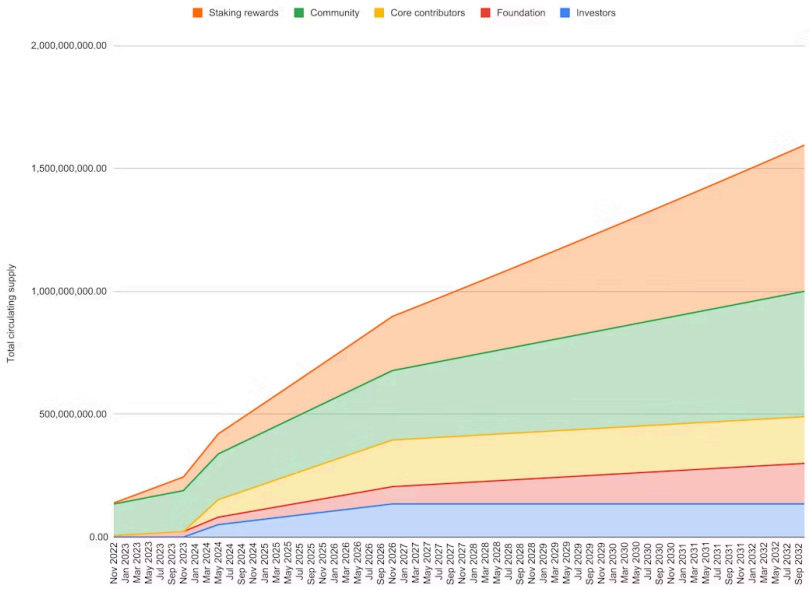
Community portion:

- Tokens are spent for grants, incentives and other growth initiatives. A majority of these tokens (410,217,359.767) are held by the Aptos Foundation, and a smaller portion (100,000,000) are held by Aptos Labs. **We could research the current holdings of both the Foundation and Labs.**
- 25,000,000 APT available initially to support ecosystem projects, grants, and other community growth initiatives
- 5,000,000 APT available initially to support the Aptos Foundation initiatives for the Foundation category
- 1/120 of the remaining tokens unlocked monthly over 10 years

Core contributors and investors portion:

- No APT available for the first twelve months
- 3/48 of such tokens unlock on the 13th month after mainnet launch and each month thereafter up to and including the 18th month
- 1/48 of the tokens unlock each month thereafter beginning on the 19th month after mainnet launch so that all such tokens are unlocked on the four-year anniversary of mainnet launch

APT supply schedule:



APT is used as the utility token for transaction fees. All fees are burned**.** Token holders can stake their APT as part of the network's proof-of-stake consensus mechanism, contributing to the security of the blockchain. In return, stakers are rewarded with additional APT tokens. APT is also used as a governance token where it used to vote on changes to economic parameters, and other proposals (all rewards and reward mechanisms are modifiable).

More information on staking can be found here: https://figment.io/insights/aptos-staking/

Transaction Pricing Mechanism Specifics

All Aptos transactions have a gas unit price (specified in Aptos tokens) that allows validators to
prioritize the highest value transactions in the network. The Aptos governance sets the minimum gas unit price. However, the market determines how quickly a transaction with a particular gas unit price is processed.

Transaction contains the standard information such as signature, sender adress,payload etc. with two additional information:

- **Sequence number**: This is an unsigned integer that must be equal to the sender's account sequence number at the time of execution. This is basically the account nonce.

- **Expiration time**: A timestamp after which the transaction ceases to be valid (i.e., expires).

Any transaction execution on the Aptos blockchain requires a processing fee. It comprises of two components:

1. Execution & IO costs:
    a. This covers usage of transient computation resources, such as processing transactions and propagating the validated record throughout the distributed network of the mainnet.
    b. It is measured in Gas Units whose price may fluctuate according to the load of the network.
    c. This portion of gas is burned permanently upon the execution of a transaction.
2. Storage fees:
    a. This covers the cost to persistently store validated record in the distributed blockchain storage.
    b. It is measured in fixed APT prices, so the permanent storage cost stays stable even as the gas unit price fluctuates with the network's transient load.
    c. The storage fee can be refunded when the allocated storage slot is deleted.
    d. This portion of Tokens is burned and minted again upon refund.

The fee model is standard :

$$fee = gas\_used * gas\_unit\_price$$

Introducing storage fees it looks as follows:

$$fee = gas\_used * gas\_unit\_price - storage\_fee\_refund\_octas$$

If it stores in the storage the storage_fee_refund_octas is a negative value. This is an interesting addition to the fee model that can stimulate freeing the storage slot.

Storage fee:

It’s charged according to the number of new slots allocated in the global state and the size increase in the existing slots.

It's priced in terms of absolute APT value. In most instances, the transaction fee is the predominant component of the overall transaction cost.

For backward compatibility reasons, the total storage fee of a transaction is currently presented to the client as **part of the total gas_used**. This means, this amount varies based on the gas unit price even for the same transaction.

Suppose we have a transaction that costs 100 gas units in execution & IO, and **5000** Octa in storage fees. The network will show:

- 100 + **5000** / 100 = 150 gas units if the gas unit price is 100
- 100 + **5000** / 200 = 125 gas units if the unit price is 200

Gas parameters set by governance:

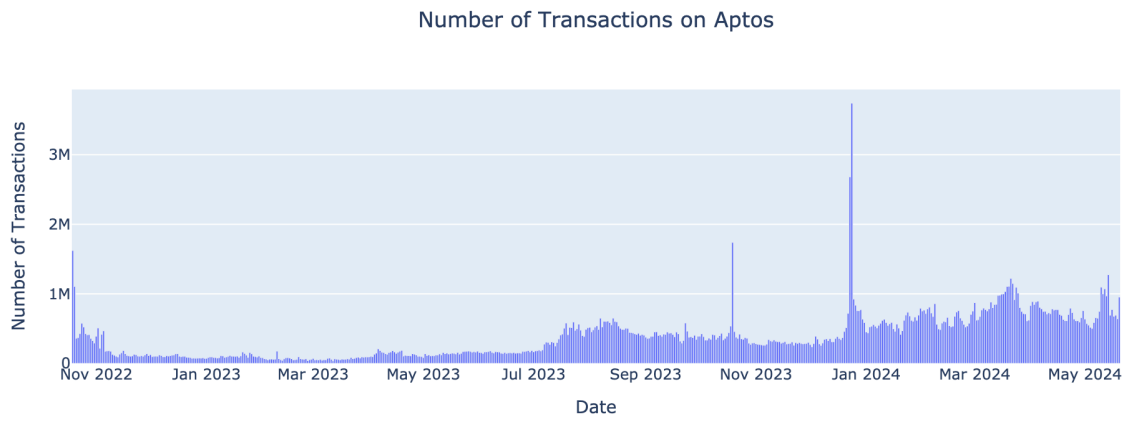
- txn.maximum_number_of_gas_units: Maximum number of gas units that can be spent.
- txn.min_transaction_gas_units`: Minimum number of gas units that can be spent.
- txn.max_execution_gas: The maximum number of gas units a transaction can spend on execution.
- txn.max_io_gas: The maximum number of gas units a transaction can spend on IO.
- txn.max_storage_fee: The maximum amount of APT a transaction can spend on persistent storage.

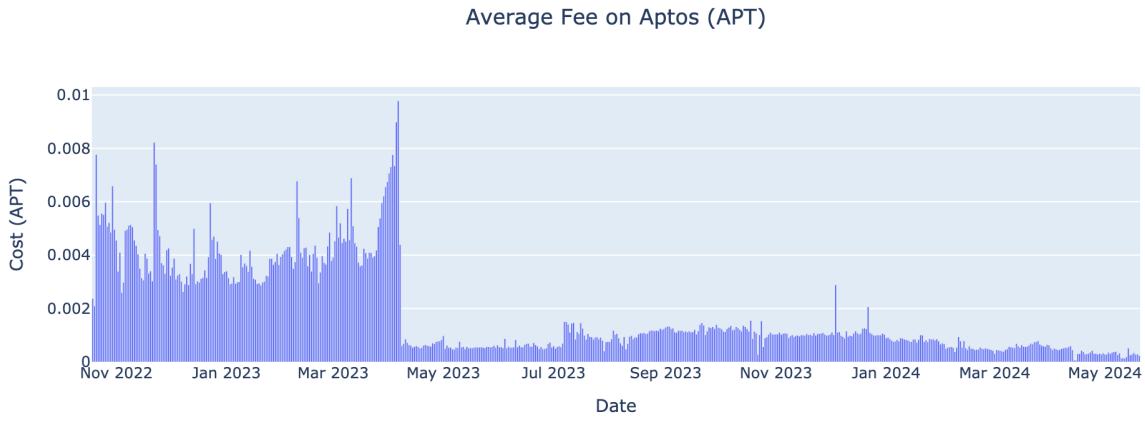
Transaction ordering:

Higher gas fees only prioritize transaction selection for the next block. Within a block, the order of transaction execution is determined by the system. This order is based on transaction shuffling, which makes parallel execution more efficient by considering conflict patterns.

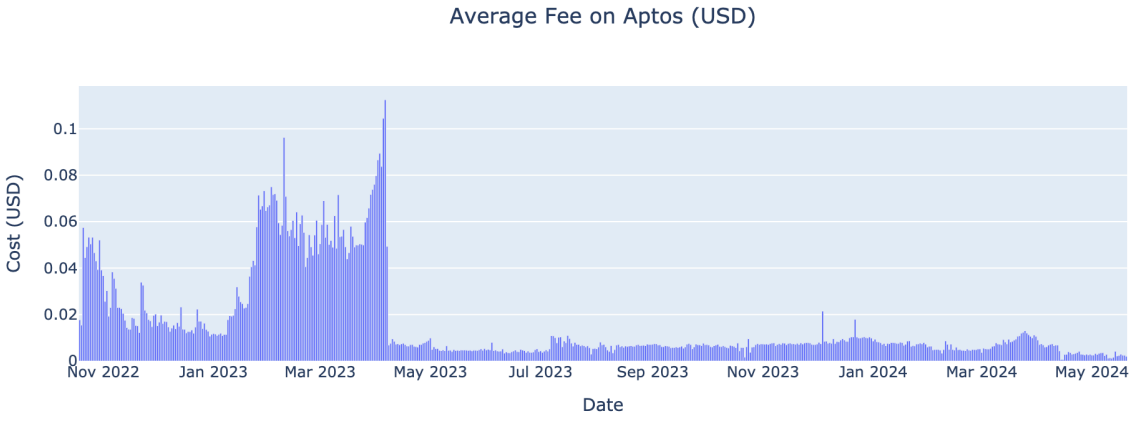
Transaction shuffling enables the shuffling of transactions within a single block so that transactions from the same sender are apart from each other as much as possible. This is done to reduce the number of conflicts and re-execution during parallel execution. **Aptos claims that the sender aware shuffling can improve the TPS by 25%.**

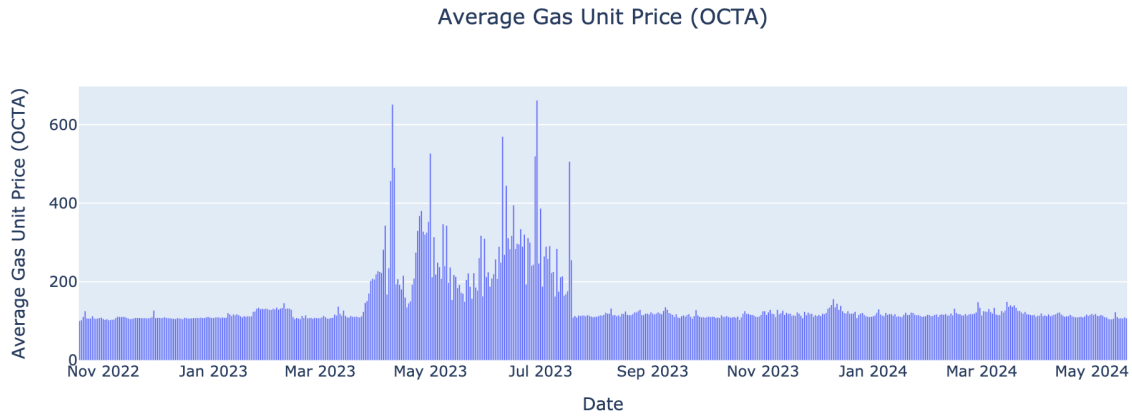
Data Analysis:

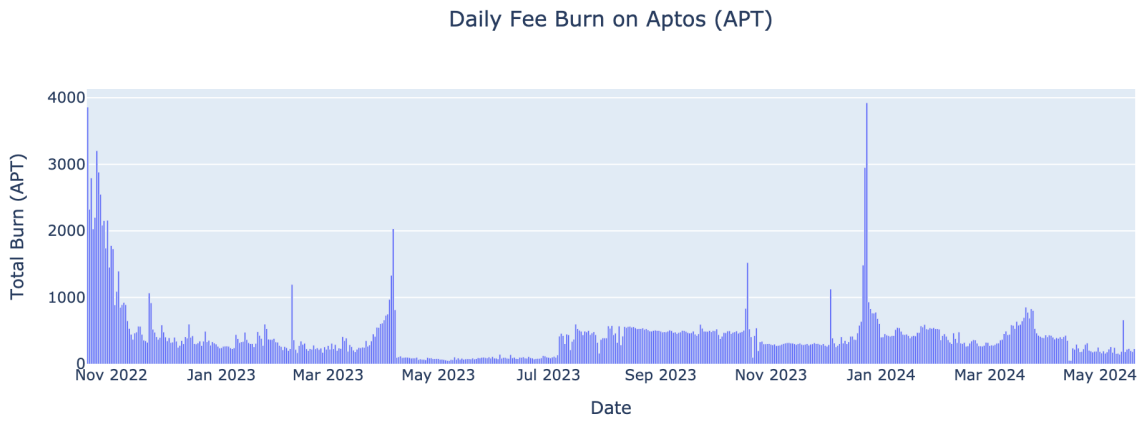


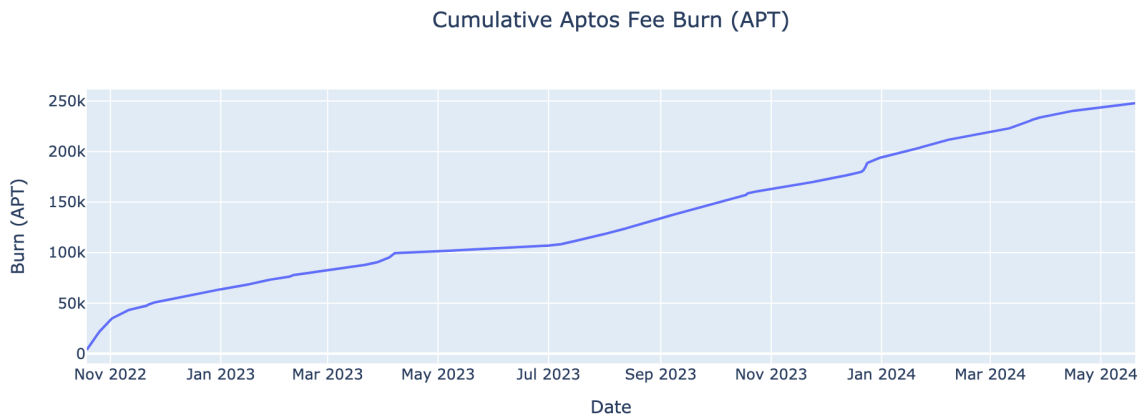


We can see the sudden drop of average transaction fee in May 2023. **Currently waiting on team response since there is no public information.**









We can see that the fee mechanism burned 250,000.00 APT so far. We could not obtain the profit since the profit differs compared to L2 solutions and is not a result of acquired fees.

Conclusion

Aptos in general is not really applicable to our use case, however its approach to Storage fees inclusion in the fee model and transaction shuffling seems like an interesting topic for discussion at least.